SIAM J. SCI. COMPUT.
Vol. 33, No. 5, pp. 2199–2216

DUALITY BASED A POSTERIORI ERROR ESTIMATION FOR QUASI-PERIODIC SOLUTIONS USING TIME AVERAGES*

M. BRAACK†, E. BURMAN‡, AND N. TASCHENBERGER†

Abstract. We propose an a posteriori error estimation technique for the computation of average functionals of solutions for nonlinear time dependent problems based on duality techniques. The exact solution is assumed to have a periodic or quasi-periodic behavior favoring a fixed mesh strategy in time. We show how to circumvent the need of solving time dependent dual problems. The estimator consists of an averaged residual weighted by sensitivity factors coming from a stationary dual problem and an additional averaging error term coming from nonlinearities of the operator considered. In order to illustrate this technique the resulting adaptive algorithm is applied to several model problems: a linear scalar parabolic problem with known exact solution, the nonsteady Navier–Stokes equations with known exact solution, and finally to the well-known benchmark problem for Navier–Stokes (flow behind a cylinder) in order to verify the modeling assumptions.

Key words. error estimation, finite elements, adaptivity, fluid dynamics, Galerkin methods

AMS subject classifications. 35Q30, 65N12, 65N30, 76D05, 76M10, 76F65

DOI. 10.1137/100809519

1. Introduction.

In this work we present an a posteriori error estimator and an adaptive algorithm for transient problems with the focus on goal-oriented output functionals based on time averages. Such functionals appear in many applications with parabolic partial differential equations of quasi-periodic character. A typical application is the determination of time averaged drag or lift coefficients of immersed bodies in a fluid: Even under stable inflow conditions, the von Karman vortex street behind an obstacle arises at higher Reynolds number and shows a quasi-periodic behavior. In such situations, drag and lift coefficients are time dependent. However, from practical point of view the usual interest is often based on time averages. Due to nonlinear effects in the underlying equation (e.g., for Navier–Stokes), the determination of those quantities require the computation of the full transient behavior for a time horizon in dependence of the typical period. Hence, those simulations are usually numerically expensive. The use of an adaptive algorithm starting with a relatively coarse grid and followed by subsequent local grid refinement may reduce the numerical effort substantially. However, such an adaptive algorithm needs reliable and efficient a posteriori error estimates on the basis of computable quantities. Moreover, the error estimation should take into account the quantity of interest.

The dual-weighted-residual method (DWR) of Becker and Rannacher [3] is the basis of goal-oriented adaptivity in many published articles. This concept has entered into various fields of continuum mechanics, as, for instance, fluid dynamics [6], solid mechanics [5], reactive flows [7], and optimization (see, e.g., [2, 4]). For stationary problems this method yields reliable a posteriori error bounds for output functionals.

*Submitted to the journal's Computational Methods in Science and Engineering section September 23, 2010; accepted for publication (in revised form) June 13, 2011; published electronically September 6, 2011. This work is partially supported by the DFG Priority Program SPP 1276 (MetStröm). This support is gratefully acknowledged.
http://www.siam.org/journals/sisc/33-5/80951.html

†Mathematisches Seminar, Christian-Albrechts-Universität zu Kiel, Ludewig-Meyn-Str. 4, 24098 Kiel, Germany (braack@math.uni-kiel.de, taschenberger@math.uni-kiel.de).

‡Department of Mathematics, University of Sussex, Falmer, Brighton, BN1 9RF, United Kingdom (erik.burman@sussex.ac.uk).

However, the transfer of this approximation theory to nonstationary problems is not easy from the practical point of view. This is due to the fact that an additional nonstationary adjoint problem has to be solved which includes the primal solution as coefficient. This implies storing of the primal solution at each time step (or recomputation or interpolation), the storage of locally refined meshes, and a severe restriction of the adaptive algorithm: Each adaptation needs to solve the primal and the adjoint problem for the whole time period. Performing adaptation steps while progressing in time is not possible. In an earlier work [8], this methodology was presented for the case of transient compressible flows. In particular, the information of the adjoint solution was embedded into a global stabilization constant which was approximated numerically. The use of local adjoint information is described and tested in [12]. For a posteriori error control of the error at end-time, there is no way around the computation of such a transient dual problem. This approach is also used for accurate computations of the drag in immersed bodies in three-dimensional Navier–Stokes flows [9, 10]. However, due to the time-dependent adjoint problem, this method is less attractive for output functionals of averaged quantities.

Therefore, we propose in this work an a posteriori error estimation technique for the computation of functionals depending on averages of the solution for nonlinear time dependent problems based on cheaper duality techniques. The exact solution is assumed to have some periodic or quasi-periodic behavior favoring a fixed mesh strategy in time. We show how to circumvent the need of solving time-dependent dual problems. The estimator consists of an averaged residual weighted by sensitivity factors coming from a stationary dual problem and an additional averaging error term coming from nonlinearities of the considered operator. The resulting adaptive algorithm is applied to a linear test problem with known exact solution and to the nonsteady Navier–Stokes equations.

This work is structured as follows. We start in the following section with an abstract formulation of the general setting by help of a variational formulation of a parabolic partial differential equation. In section 3, we introduce the new concept of a posteriori error control for time averaged quantities. In particular, we use an error representation in order to derive a numerical strategy for a posterori error control and mesh adaptivity. Numerical examples are presented in section 4.

2. Setting of the nonlinear problem.

2.1. The continuous and the semidiscrete variational problem. Let $\Omega \in \mathbb{R}^d$ be a Lipschitz domain and let $I := [0, T]$ with $T > 0$ the time interval. We consider the abstract variational problem in the Bochner space $W := H^1(I, V)$ with a Hilbert space V. Hence, functions in W are weakly differentiable with image in V. For given data $f \in L^2(I, W')$ we look for solutions,

$$u \in W : \quad (\partial_t u, \psi)_Q + \int_I A(u, \psi)\, dt = (f, \psi)_Q \quad \forall \psi \in W\,, \tag{2.1}$$

where $(\cdot, \cdot)_Q$ stands for the L^2-scalar product in the space-time slab $Q := I \times \Omega$, and $A : V \times V \to \mathbb{R}$ is a semilinear form (linear in the second argument). This semilinear form is supposed to be Frechét differentiable with respect to the first argument.

The discrete solution u_h is sought in a (semi-)discrete subspace, $u_h \in H^1(I, V_h)$, with a conforming finite element space $V_h \subset V$. The corresponding semidiscretized form (still continuous in time) of (2.1) reads

$$u_h \in W_h : \quad (\partial_t u_h, \psi)_Q + \int_I A(u_h, \psi)\, dt = (f, \psi)_Q \quad \forall \psi \in W_h\,, \tag{2.2}$$

with $W_h := H^1(I, V_h)$. By an overline we denote time averages, i.e.,

$$\overline{u} := \frac{1}{T}\int_0^T u(t)\,dt \ \in \ V\,.$$

We are interested in accurate determination of linear functional output

$$J : V \to \mathbb{R}$$

for the time average, i.e., for $J(\overline{u})$. The aim is the development of an error estimator η for the error in terms of this functional,

$$J(\overline{u} - \overline{u}_h) \approx \eta\,.$$

2.2. Time averaged equation. Integration of (2.1) in time (by taking a test function constant in time), dividing by T and integration by parts yields the time-averaged equation

$$u \in H^1(I, V): \quad \sigma_T(u, \varphi) + \overline{A}(u, \varphi) = (\overline{f}, \varphi) \quad \forall \varphi \in V\,, \tag{2.3}$$

where we use the form $\overline{A} : W \times V \to \mathbb{R}$ and the linear form $\sigma : W \times V \to \mathbb{R}$ defined by the semilinear form

$$\overline{A}(u, \varphi) := \frac{1}{T}\int_I A(u, \varphi)dt\,,$$
$$\sigma_T(u, \varphi) := \frac{1}{T}(u(T) - u(0), \varphi)\,.$$

Corresponding to (2.3) the time-averaged residual is denoted by

$$\overline{\varrho}(u, \varphi) := (\overline{f}, \varphi) - \overline{A}(u, \varphi) - \sigma_T(u, \varphi)\,.$$

The basic result of this work is based on fluctuations in time due to possible nonlinearities of $A(u, \cdot)$, defined by

$$K(u, \varphi) := \overline{A}(u, \varphi) - A(\overline{u}, \varphi)\,.$$

Note that K vanishes if $A(\cdot, \cdot)$ is bilinear and all coefficients are constant in time. An important equation is the Galerkin orthogonality for the averaged residual:

$$\overline{\varrho}(u, \varphi) - \overline{\varrho}(u_h, \varphi) = 0 \quad \forall \varphi \in V_h\,. \tag{2.4}$$

3. A posteriori error representation. By $A'(\xi) : V \times V \to \mathbb{R}$ we denote the Frechét derivative of $A(\cdot, \cdot)$ at $\xi \in V$:

$$A'(\xi)(v, \varphi) := \lim_{\epsilon \to 0} \frac{1}{\epsilon}\left(A(\xi + \epsilon v, \varphi) - A(\xi, \varphi)\right)\,.$$

The a posteriori error representation we are going to present is based on a stationary adjoint problem of the form

$$z \in V: \quad A'(\xi)(\varphi, z) = J(\varphi) \quad \forall \varphi \in V\,. \tag{3.1}$$

If $A(\cdot, \cdot)$ is nonlinear in the first argument, $z = z(\xi)$ depends on the choice of $\xi \in V$. By $i_h : V \to V_h$ we denote an arbitrary interpolation operator.

PROPOSITION 3.1. *If A is continuously Frechét differentiable, the discretization error with respect to the linear functional J can be represented by*

$$J(\overline{u}-\overline{u}_h) = \overline{\varrho}(u_h, z - i_h z) + K(u_h)(z) - K(u)(z) + \sigma_T(u_h - u, z)\,,$$

where $z = z(\overline{\xi}) \in V$ is the dual solution of (3.1) *to the linearization at $\overline{\xi} = \lambda\overline{u} + (1-\lambda)\overline{u}_h$ with an appropriate $\lambda \in [0,1]$.*

Proof. We performed an averaging in time of the continuous problem (2.1) in order to get (2.3). The same averaging can be carried out to the discrete equation (2.2). Subtracting both results leads to the following perturbed Galerkin orthogonality property for $e := u - u_h$:

$$\overline{A}(u,\varphi) - \overline{A}(u_h,\varphi) = -\sigma_T(e,\varphi) \qquad \forall \varphi \in V_h\,. \tag{3.2}$$

The mean value theorem ensures the existence of at least one $\lambda \in [0,1]$, so that for $\overline{\xi} := \lambda\overline{u} - (1-\lambda)\overline{u}_h \in V$ it holds that

$$A'(\overline{\xi})(\overline{e},\varphi) = A(\overline{u},\varphi) - A(\overline{u}_h,\varphi) \qquad \forall \varphi \in V\,.$$

Let $z = z(\overline{\xi}) \in V$ be the associated dual solution of (3.1). Hence, for the discretization error in terms of J it holds that

$$J(\overline{u}) - J(\overline{u}_h) = A'(\xi)(\overline{e}, z) \;=\; A(\overline{u}, z) - A(\overline{u}_h, z)\,.$$

With the Galerkin orthogonality (3.2) for an arbitrary interpolation $i_h z \in V_h$ of z and together with (2.3) we obtain

$$\begin{aligned} J(\overline{u}) - J(\overline{u}_h) &= \overline{A}(u,z) - \overline{A}(u_h,z) - K(u,z) + K(u_h,z) \\ &= \overline{A}(u, z - i_h z) - \overline{A}(u_h, z - i_h z) - K(u,z) + K(u_h,z) - \sigma_T(e, i_h z) \\ &= (\overline{f}, z - i_h z) - \overline{A}(u_h, z - i_h z) - K(u,z) + K(u_h,z) \\ &\quad - \sigma_T(u,z) + \sigma_T(u_h, i_h z)\,. \end{aligned}$$

Expressing this result in terms of the time-averaged residual we arrive at

$$\begin{aligned} J(\overline{u}) - J(\overline{u}_h) &= \overline{\varrho}(u_h, z - i_h z) + \sigma_T(u_h, z - i_h z) - K(u,z) + K(u_h,z) \\ &\quad - \sigma_T(u,z) + \sigma_T(u_h, i_h z)\,. \end{aligned}$$

This gives us the desired error representation. ∎

In this proposition it was shown that the error can be represented in the form

$$J(\overline{u} - \overline{u}_h) = e_1 + e_2 + e_3\,,$$

with the contributions

$$e_1 := \overline{\varrho}(u_h)(z - i_h z)\,, \quad e_2 \;:=\; K(u_h, z) - K(u, z)\,, \quad e_3 \;:=\; \sigma_T(u_h - u, z)\,. \tag{3.3}$$

In the next section, we discuss how these terms can be numerically approximated.

3.1. Finite dimensional dual problem. We propose to solve a discrete approximation of the dual problem (3.1) for $\xi = u_h$ (i.e., $\lambda = 0$):

$$z_h \in V_h: \quad A'(\overline{u}_h)(\varphi, z_h) = J(\varphi) \quad \forall \varphi \in V_h\,. \tag{3.4}$$

Building the Frechét derivative in (3.4) at $\overline{u}_h$ instead of ξ is reasonable when $\overline{u}_h$ is close to $\overline{u}$ or when the nonlinearity of A is moderate. When $A(u,\varphi)$ is linear in both arguments the dual solution z_h is the solution of

$$z_h \in V_h: \qquad A(\varphi, z_h) \;=\; J(\varphi) \qquad \forall \varphi \in V_h\,.$$

3.2. Finite element approximation. In this section, we briefly describe the finite elements we used for the numerical test cases presented later in this work. Furthermore, the specific finite elements are used for a possible numerical approximation of the error representation. However, for other finite element approximations, the error representation in Proposition 3.1 is still valid but different strategies for the numerical approximation have to be designed.

Let $\mathcal{T}_h$ be a shape-regular, admissible decomposition of Ω into quadrilaterals for $d = 2$ or hexahedrals for $d = 3$. The diameter of a cell $K \in \mathcal{T}_h$ will be denoted by h_K. Let $\widehat{K} := (-1, 1)^d$ denote the reference element and $\mathbb{Q}_r(\widehat{K})$ the space of all polynomials on $\widehat{K}$ with maximal degree $r \geq 0$ in each coordinate direction. We will use the continuous H^1-conforming finite element spaces

$$Q_r(\mathcal{T}_h) := \{v_h \in H^1(\Omega) : \ v_h|_K \circ F_K \in \mathbb{Q}_r(\widehat{K}) \quad \forall K \in \mathcal{T}_h\}\,.$$

Hence, our finite element approximation of u is in the scalar case $u_h \in V_h := Q_r(\mathcal{T}_h)$. For vector-valued problems with s components it is $V_h := Q_r(\mathcal{T}_h)^s$. The Lagrange nodal basis of V_h will be denoted by $\{\phi_1, \ldots, \phi_n\}$. Later we will make use of the nodal basis of $Q_{2r}(\mathcal{T}_{2h})^s$ denoted by $\{\phi_1^{(2r)}, \ldots, \phi_n^{(2r)}\}$.

3.3. Numerical evaluation of the estimator. Now, we discuss how the terms e_k, $k \in \{1, 2, 3\}$, in (3.3) can be approximated numerically by computable quantities $\eta_k \approx e_k$. The stopping criterion of the adaptive algorithm is

$$\eta := \eta_1 + \eta_2 + \eta_3 \ < \ TOL\,, \tag{3.5}$$

for a given tolerance TOL. Furthermore, the estimator η must be localized in order to use the information for mesh adaptation. For this, we will express the indicator parts η_k by sums

$$\eta_k = \sum_{i=1}^{n} \eta_{k,i}\,,$$

where i is the index of the ith basis function. This allows the use of the nonnegative quantities

$$\zeta_i := |\eta_{1,i} + \eta_{2,i} + \eta_{3,i}| \geq 0\,, \tag{3.6}$$

as criterion for mesh refinement. A measure for the quality of such a localization is that the quotient

$$\sum_{i=1}^{n} \zeta_i / |\eta|$$

should be close to one.

For the computation of the η_k and ζ_i it turns out to be very helpful if $\mathcal{T}_h$ results from a globally coarser mesh $\mathcal{T}_{2h}$. The reason is that we can perform a patchwise recovery process of higher order. By $i_{2h}^{(2r)} : Q_r(\mathcal{T}_h) \to Q_{2r}(\mathcal{T}_{2h})$ we denote the nodal interpolation to elements of order $2r$ on the mesh $\mathcal{T}_{2h}$.

Approximation of e_1. The underlying difficulty for approximating e_1 is comparable with the situation for stationary problems: We need a suitable approximation of the interpolation error $z - i_h z$. This can be done, for instance, by the patchwise

higher-order recovery mentioned above. Now the quadratic recovery of z_h gives the following approximation of η_1:

$$e_1 \approx \eta_1 \;:=\; \overline{\varrho}(u_h, i_{2h}^{(2r)} z_h - z_h)\,. \tag{3.7}$$

We introduce the nodal interpolation operator $i_{2h} : V_h \to V_{2h}$ and the filtering operator $\pi_h : V_h \to V_h$, defined by

$$\pi_h \phi := \phi - i_{2h}\phi\,,$$

giving the small-scale linear fluctuations. We denote the nodal vector of the filtered dual solution $\pi_h z_h$ by $Z^\pi \in \mathbb{R}^n$, i.e.,

$$\pi_h z_h = \sum_{i=1}^{n} \phi_i Z_i^\pi\,. \tag{3.8}$$

The error estimator can be localized on the basis of the following proposition.

PROPOSITION 3.2. *The estimator η_1 of* (3.7) *can be represented as the l_2 scalar product of the vector of the time-averaged residuals evaluated at the nodal basis of $Q_{2r}(\mathcal{T}_{2h})$*

$$\Psi_i = \overline{\varrho}(u_h, \phi_i^{(2r)})$$

and the nodal vector of the filtered discrete dual solution: $\eta_1 = \Psi^T Z^\pi$.

Proof. The higher order interpolation operator $i_{2h}^{(2r)}$ is obviously the identity on V_{2h}. This implies that

$$i_{2h}^{(2r)} i_{2h}\phi = i_{2h}\phi \quad \forall \phi \in V_h\,.$$

Therefore, it holds, for all $\phi \in V_h$, that

$$i_{2h}^{(2r)}\pi_h\phi - \pi_h\phi = i_{2h}^{(2r)}\phi - i_{2h}^{(2r)} i_{2h}\phi - (\phi - i_{2h}\phi) \;=\; i_{2h}^{(2r)}\phi - \phi\,.$$

This implies for the functions of the nodal basis $\{\phi_i\}$ of V_h that

$$\phi_i^{(2r)} - \phi_i = i_{2h}^{(2r)}\pi_h\phi_i - \pi_h\phi_i\,.$$

Now, we use the following identities:

$$\begin{aligned}
\overline{\varrho}(u_h, i_{2h}^{(2r)} z_h - z_h) &= \sum_{i=1}^{n} \overline{\varrho}(u_h, \phi_i^{(2r)} - \phi_i)\cdot Z_i \\
&= \sum_{i=1}^{n} \overline{\varrho}(u_h, i_{2h}^{(2r)}\pi_h\phi_i - \pi_h\phi_i)\cdot Z_i \\
&= \sum_{i=1}^{n} \overline{\varrho}(u_h, i_{2h}^{(2r)}\phi_i - \phi_i)\cdot Z_i^\pi\,.
\end{aligned}$$

By using the identity $i_{2h}^{(2r)}\phi_i = \phi_i^{(2r)}$ and Galerkin orthogonality $\overline{\varrho}(u_h, i_{2h}^{(2r)}\phi_i) = 0$, we arrive at the assertion

$$\overline{\varrho}(u_h, i_{2h}^{(2r)}\phi_i - \phi_i)\cdot Z_i^\pi = \overline{\varrho}(u_h, \phi_i^{(2r)}) \;=\; \Psi_i\,. \qquad \square$$

Local error indicators $\eta_{1,i}$ for each node $\mathcal{N}_i$ of the mesh can be obtained by

$$\eta_{1,i} := \Psi_i Z_i^\pi \,, \tag{3.9}$$

since the sum over all $\eta_{1,i}$ is obviously equal to η_1.

We aim to reduce the necessary numerical costs as far as possible. Therefore, storing the solution u_h at each time step is not very attractive. The presentation of η_1 in the previous proposition leads us directly to a cheap rule to evaluate η_1 where only one additional vector, namely Ψ, has to be stored for the whole computation. This vector of averaged residuals can be successively updated after each time step. If Ψ^j is the vector of these residuals up to time $t = t_j$, i.e.,

$$\Psi_i^j = \frac{1}{t_j}\left((u_h(0),\phi_i^{(2r)}) + \int_0^{t_j}[(f,\phi_i^{(2r)}) - A(u_h(t),\phi_i^{(2r)})]\,dt\right) \,,$$

we obtain the update by

$$\Psi_i^{n+1} = \frac{1}{t_{n+1}}\left(t_n\Psi_i^n + \int_{t_n}^{t_{n+1}}[(f,\phi_i^{(2r)}) - A(u_h(t),\phi_i^{(2r)})]\,dt\right) \,.$$

Here, $\phi_i^{(2r)} := \varphi_i^{(2r)} - \varphi_i$ is the difference of the Q_{2r} and Q_r Lagrange finite element basis functions for the node N_i in the triangulation.

Approximation of e_2. The term e_2 vanishes for linear problems, i.e., when $A(\cdot,\cdot)$ is bilinear. For the more interesting case of nonlinear problems it holds

$$\begin{aligned} e_2 &= K(u_h,z) - K(u,z) \\ &= [\overline{A}(u_h,z) - \overline{A}(u,z)] + [A(\overline{u}_h,z) - A(\overline{u},z)] \,. \end{aligned}$$

For e_1 it was argued that $\overline{i_{2h}^{(2r)}u_h}$ is a better approximation to $\overline{u}$ than it is $\overline{u}_h$. The same argument here leads to

$$e_2 \approx \eta_2 \;:=\; K(u_h,z_h) - K(i_{2h}^{(2r)}u_h,z_h) \,. \tag{3.10}$$

The interpolant $i_{2h}^{(2r)}u_h$ is a better approximation than u_h if h is small enough, the solution u is sufficiently regular and the mesh has certain symmetry properties. However, we do not really need $i_{2h}^{(2r)}u_h$ to be closer to u than u_h, but only that the averages improve. This seems reasonable if there is some scale similarity in the discrete solution.

For (3.10) the localization is given by

$$\eta_{2,i} := (K(u_h)(\varphi_i) - K(i_{2h}^{(2)}u_h)(\varphi_i))Z_i \,. \tag{3.11}$$

As illustrated later by a numerical example, the summation of cellwise absolute values may somewhat overestimate the sum, i.e., the quotient

$$\eta_2^{-1}\sum_{i=1}^{N}|\eta_{2,i}|$$

may become considerably large. However, since the localization is used only for adaptivity and not for error control, a certain overestimation due to localization is maintainable as long as the quotient remains bounded.

TABLE 3.1
Adaptive algorithm.

	Adaptive algorithm
1.	Start with initial mesh $\mathcal{T}_0$, $k := 0$.
2.	Solve nonstationary primal problem (2.2) obtaining $u_k \in W_k$.
3.	Solve stationary dual problem (3.4) obtaining $z_k \in V_k$.
3.	Evaluate error estimator η according to (3.7),(3.10), and (3.13).
4.	If (3.5) is fulfilled, then STOP.
5.	Localize error indicators by determining $\eta_{k,i}$ and ζ_i according to (3.9),(3.11),(3.13), and (3.6).
6.	Order $K_1, K_2, \ldots, K_n$ such that $\zeta_{K_1} \geq \zeta_{K_2} \geq \ldots$.
7.	Determine cellwise error indicators $\{\zeta_{K_j}\}$, e.g., by (3.13).
8.	Refine cells K_i, with $1 \leq i \leq R \leq n_r$.
9.	Increase $k+1 \to k$ and go to 2.

Due to the patch structure of the mesh for the higher order recovery, it is also attractive for a validation to compute the Q_{2r}-solution $u_{2h}^{(2r)}$ of the problem. This allows the approximation

$$e_2 \approx \eta_2' := K(u_h, z_h) - K(u_{2h}^{(2r)}, z_h)\,. \tag{3.12}$$

The numerical results presented later will show that both strategies give relatively good efficiency indices, i.e., a ratio of the estimator and the error close to one. However, the latter choice is not useful in practice, because with the help of $u_{2h}^{(2r)}$ one may approximate the error $J(\overline{u} - \overline{u}_h)$ directly without dual problem by $J(\overline{u_{2h}^{(2r)}} - \overline{u}_h)$.

Approximation of e_3. If the error is bounded, $|u(T) - u_h(T)| \leq C$, with a constant C independent of T, the error part e_3 becomes arbitrarily small for $T \to \infty$. Hence, for periodic or quasi-periodic solutions, the term e_3 can often be neglected. However, for small T this part should be taken into account, for instance, by using a higher order interpolation as well:

$$\eta_3 := \frac{1}{T}(i_{2h}^{(2r)} u_h(T) - u_h(T) - i_{2h}^{(2r)} u_h(0) + u_h(0), z_h)\,,$$
$$\eta_{3,i} := \frac{1}{T}(i_{2h}^{(2r)} u_h(T) - u_h(T) - i_{2h}^{(2r)} u_h(0) + u_h(0), \varphi_i) Z_i\,.$$

Similarly as for η_2', the use of higher order computation instead of higher order interpolation lead to η_3'.

3.4. Adaptive algorithm. Now, we collect the adaptive algorithm for obtaining a hierarchy of meshes $\mathcal{T}_{h_k}$ with mesh size function h_k. For abbreviation, the finite element space W_{h_k} and the corresponding finite element solutions u_{h_k}, z_{h_k} are simply denoted by W_k, u_k, and z_k, respectively. In each cycle of the adaptive loop we have to decide which cell $K \in \mathcal{T}_{h_k}$ should be refined. This is done by shifting the information of the ζ_i to cellwise quantities ζ_K. This is usually done by building the sum of those contributions η_i, so that the support of the corresponding basis function, $supp(\varphi_i)$ intersects with K:

$$\zeta_K := \sum_{supp(\varphi_i) \cap K \neq \emptyset} \zeta_i\,. \tag{3.13}$$

The adaptive step consists of refining those n_r cells K with the largest values of ζ_K. The overall algorithm looks as shown in Table 3.1.

4. Numerical tests. In this section we present the results of the adaptive algorithm to several test problems. As a first proof of concept, we consider a linear problem with known exact solution. This test case is chosen in order to validate the case where the fluctuation term e_2 due to nonlinearities vanishes. The second test case is nonlinear and consists of the Navier–Stokes flow with exact solution. Finally, we apply the adaptive algorithm to the well-known benchmark problem "Flow around a cylinder"; see [11]. All simulations are done by bilinear elements, i.e., $r = 1$.

4.1. Heat equation with known exact solution. As a first example, we take the linear heat equation in the unit square $\Omega = (0,1)^2$ with known exact solution:

$$\begin{aligned} \partial_t u - \Delta u &= f \quad \text{in } \Omega\,, \\ u &= u_{\partial\Omega} \quad \text{on } \partial\Omega\,, \end{aligned}$$

and initial condition $u(x,0) = u_0(x)$. The right-hand side is chosen in such a way that the solution becomes the rotating cone

$$u(x,t) = \frac{1}{1 + a\|x - x^*(t)\|^2}\,,$$

with $\psi(x) := \|x - x^*(t)\|^2$, $x^* = (\frac{1}{2} + \frac{1}{4}\cos(2\pi t), \frac{1}{2} + \frac{1}{4}\sin(2\pi t))$ and $a = 50$. The right-hand side is determined by $f = \partial_t u - \Delta u$. The corresponding bilinear form reads

$$A(u,\varphi) = (\nabla u, \nabla\varphi)\,.$$

We used the second order Crank–Nicholson scheme with a timestep of size $\Delta t = 0.00125$. A complete turn of the rotating cone takes, therefore, 800 timesteps. The functional considered is the mean flux along $\Gamma = (0,1) \times \{0\}$ defined by

$$J(\overline{u}) := \frac{1}{c}\int_0^1 \int_\Gamma \frac{\partial u}{\partial x_2}\,dx\,dt\,.$$

The constant c is chosen in order to normalize the flux to 1:

$$c := \int_0^1 \int_\Gamma \frac{\partial u}{\partial x_2}\,dx\,dt \;=\; 2a\int_0^1 \left(\int_0^1 u(x_1,0,t)^2\,dx_1\right) x_2^*(t)\,dt \;=\; 0.249702115724673\,.$$

This value is computed by numerical integration using the exact solution. The value c is accurate up to at least eight digits. However, in the numerical computation of u_h we have two sources of errors, one due to time discretization and one due to discretization in space. The error in time is of order 10^{-4}. Since we compute in this example only an estimator for the error with respect to space, we limit our computations to meshes where the spatial error dominates the error in time.

In Table 4.1 we list the results. We compare the computed estimator $\eta = \eta_1 + \eta_3$ and the error $J(\overline{u} - \overline{u}_h)$. Its ratio, the efficiency index

$$I_{eff} = \frac{\eta}{J(\overline{u} - \overline{u}_h)}\,,$$

is listed in the last column. The error is underpredicted by a nearly constant factor which indicates that the asymptotic behavior of η is correct. This behavior is also very nicely visible in the error plot in Figure 4.1. The solution is shown in Figure 4.2 and the obtained meshes are shown in Figure 4.3.

TABLE 4.1
Comparison of the error in the flux $J(\overline{u}-\overline{u}_h)$ and the estimator η for subsection 4.1. The last column shows the efficiency index I_{eff} which is the quotient of both quantities.

# nodes	$J(\bar{u}_h)$	$J(\bar{u}-\bar{u}_h)$	$\eta=\eta_1+\eta_3$	I_{eff}
25	1.37871	3.7871e-01	1.8643e-01	0.492
81	1.07471	7.4712e-02	5.8384e-02	0.781
189	1.02588	2.5875e-02	2.0858e-02	0.806
401	1.01212	1.2127e-02	8.8753e-03	0.732
1055	1.00578	5.7755e-03	4.0934e-03	0.709
2003	1.00298	2.9796e-03	2.0060e-03	0.673
3685	1.00142	1.4195e-03	9.7644e-04	0.688
6029	1.00073	7.2975e-04	4.8437e-04	0.664
12609	1.00035	3.4561e-04	2.4156e-04	0.699
19361	1.00017	1.6748e-04	1.1934e-04	0.713
43465	1.00009	8.9616e-05	6.0744e-05	0.678
62643	1.00004	3.9643e-05	2.9259e-05	0.738
146257	1.00002	2.1785e-05	1.5417e-05	0.708

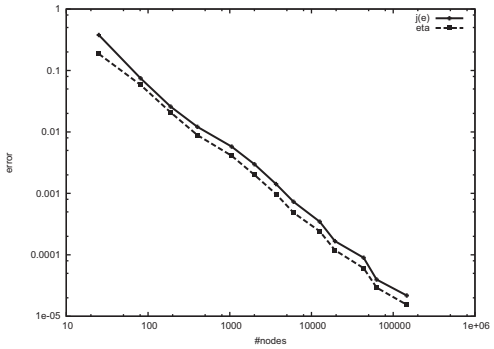

FIG. 4.1. *Comparison of the error in the flux $J(\overline{u}-\overline{u}_h)$ (continuous line) and the estimator η (dashed line) for the rotating cone problem (subsection 4.1).*

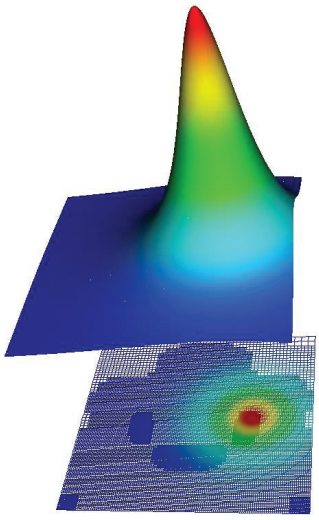

FIG. 4.2. *Solution with mesh for roting cone problem (subsection 4.1).*

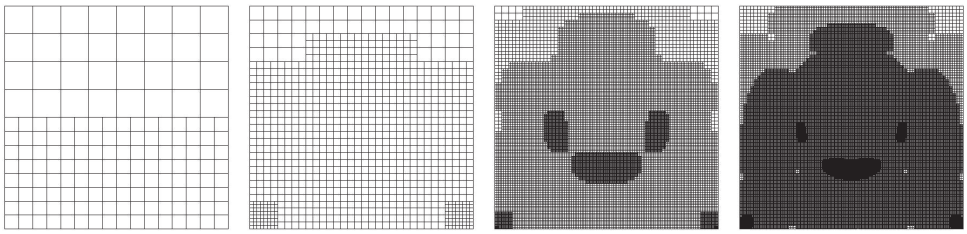

FIG. 4.3. *Obtained meshes for rotating cone (subsection* 4.1*).*

4.2. Navier–Stokes with known exact solution. We solve the nonsteady incompressible Navier–Stokes equations on the unit square $\Omega := (0,1)^2$,

$$\begin{aligned} \partial_t v + (v \cdot \nabla)v - \nu \Delta v + \nabla p &= f\,, \\ \operatorname{div} v &= 0\,, \end{aligned}$$

with Dirichlet conditions on $\partial\Omega$. The right-hand side f is defined in such a way that the exact solution is given by

$$v_1(x,y,t) = g(t)\psi(y)\,, \quad v_2(x,y,t) \;=\; g(t)\psi(x)\,, \quad p(x,y,t) \;\equiv\; 0\,,$$

with time dependent function $g(t) = 1 + \sin(-\frac{\pi}{2} + 2\pi t)$, $t \in [0,1]$, and

$$\psi(x) := x - \frac{e^{\frac{x}{\nu}} - 1}{e^{\frac{1}{\nu}} - 1}\,.$$

The viscosity is set to $\nu = 0.01$ and results in a relatively strong boundary layer. As before, the discrete solutions u_h and z_h consist of Q_1 finite elements. The considered functional is the boundary integral along the upper boundary Γ:

$$J(\bar{u}) := \int_0^T \int_\Gamma \left(\frac{\partial v}{\partial x_1} - pn \right) ds\, dt.$$

In the case of Navier–Stokes, the part e_2 of the error takes the form of the Reynolds stresses

$$\begin{aligned} e_2 &= \left(\overline{(v_h \cdot \nabla)v_h} - (\overline{v}_h \cdot \nabla)\overline{v}_h - \overline{(v \cdot \nabla)v} + (\overline{v} \cdot \nabla)\overline{v}, z \right) \\ &= \left(\overline{(v_h \cdot \nabla)v_h - (v \cdot \nabla)v}, z \right) + \left((\overline{v} \cdot \nabla)\overline{v} - (\overline{v}_h \cdot \nabla)\overline{v}_h, z \right) \\ &= \left((\overline{v'} \cdot \nabla)\overline{v_h} + (\overline{v_h} \cdot \nabla)\overline{v'} + (\overline{v'} \cdot \nabla)\overline{v'} - \overline{(v' \cdot \nabla)v_h + (v_h \cdot \nabla)v' + (v' \cdot \nabla)v'}, z \right), \end{aligned}$$

with the notation $v' := v - v_h$. We know from turbulence modeling that at high Reynolds number an accurate approximation of such terms is anything else than trivial. However, the use of this approach for solely mesh adaptation (instead of error estimation) does not require an accurate estimator but rather qualitatively information on the error in terms of error localization. Moreover, even the (original) DWR method with nonsteady dual problem provides only reliable estimation of the linear part of the equation. To the best of the author's knowledge, up to now there is no published work beyond the treatment of the linear parts of the underlying operator.

For this specific example, we list the corresponding error parts and the efficiency index for global mesh refinement in Table 4.2 for $T = 1$ and for $T = 10$. In the

TABLE 4.2
Navier–Stokes (subsection 4.2) with global mesh refinement for $T = 1$ and $T = 10$.

# nodes	$J(\bar{u} - \bar{u}_h)$s	η_1	η_2	η_3	η	I_{eff}
$T = 1$						
289	4.73e-02	2.08e-03	−1.08e-03	−5.82e-04	4.19e-04	0.01
1089	8.96e-03	2.30e-03	3.44e-04	−3.49e-05	2.61e-03	0.29
4225	1.77e-03	6.64e-04	1.35e-04	−1.93e-06	7.98e-04	0.45
16641	3.67e-04	1.12e-04	3.56e-05	−1.17e-07	1.48e-04	0.40
66049	8.52e-05	2.09e-05	9.08e-06	−7.22e-09	3.00e-05	0.35
263169	2.09e-05	4.83e-06	2.32e-06	−4.51e-10	7.15e-06	0.34
$T = 10$						
25	2.09e-01	−1.16e-02	−1.53e-02	−5.93e-03	−3.28e-02	−0.16
81	1.32e-01	−1.44e-02	−8.40e-03	−7.30e-04	−2.36e-02	−0.18
289	3.41e-02	1.94e-03	−1.58e-03	−5.86e-05	3.02e-04	0.01
1089	5.42e-03	2.30e-03	3.13e-04	−3.46e-06	2.61e-03	0.48
4225	8.08e-04	6.62e-04	1.33e-04	−1.93e-07	7.95e-04	0.98
16641	1.22e-04	1.12e-04	3.54e-05	−1.17e-08	1.48e-04	1.21
66049	2.38e-05	2.12e-05	9.08e-06	−7.25e-10	3.03e-05	1.27

TABLE 4.3
Relation of the local error indicators and the parts of the estimator for subsection 4.2.

#nodes	$\zeta_1/\lvert\eta_1\rvert$	$\zeta_2/\lvert\eta_2\rvert$	$\zeta_3/\lvert\eta_3\rvert$
289	28.2	81.4	2.3
557	11.3	1659.7	23.1
1085	15.2	257.4	3.8
2069	334.1	138.1	58.2
4709	17.1	111.4	71.2
12465	8.6	60.2	13.0
37987	5.1	50.8	2.2

simulation of only one period ($T = 1$), the estimator underestimates the exact error by a factor between 0.3 and 0.45, except on the coarsest mesh. These values are acceptable because it is a nonlinear problem. For ten periods of simulation ($T = 10$), the estimator is even better leading to an efficiency index close to one on finer meshes.

Obviously, the contribution η_3 is estimated by nearly zero. The efficiency index remains bounded but the error is systematically underestimated.

Now, we use the estimator for local mesh refinement. In Table 4.3 we list the relation of the local error indicators ζ_k and the indicators η_k. The relation does not increase under mesh refinement so that the localization works quite well. However, for the nonlinear error part η_2, the localization leads to a quite large factor between ζ_2 and $|\eta_2|$. In order to address the question if ζ_2 can be used as an efficient refinement criterion anyhow, we apply two types of local refinement strategies: (i) based on η_1 and η_3, (ii) based on η_1, η_2, and η_3. The obtained error curves are depicted in Figure 4.4. Both strategies are much more efficient than global refinement. The use of all error parts η_1, η_2, and η_3 (dotted curve in Figure 4.4) is even better than the use of η_1 and η_3 only (dashed curve). The errors and indicators are also listed in Table 4.4. For obtaining an error of about $1.5 \cdot 10^{-5}$ only 12,465 nodes are needed in comparison to 263,169 nodes on a regular mesh with an error of about $2 \cdot 10^{-5}$.

The efficiency index of the estimator behaves on locally refined meshes much worse compared to structured meshes. On finer meshes, not even the sign of the error is correct. However, as discussed previously, the estimator works fine for detecting appropriate cells for local mesh refinement.

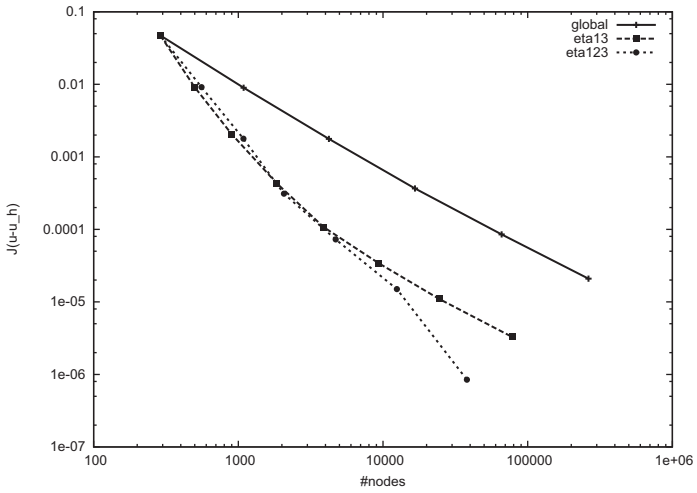

FIG. 4.4. *Comparison of the discretization error in dependence of the number of nodes for global (continuous curve) and local refinement (dashed curve) on the basis of η_1, η_3, and on the basis of η_1, η_2, η_3 (dotted curve) for the Navier–Stokes problem in subsection* 4.2 *with* $T = 1$.

TABLE 4.4
Navier–Stokes (subsection 4.2*) with local mesh refinement and* $T = 1$.

# nodes	$J(\bar{u} - \bar{u}_h)$	η_1	η_2	η_3	η	I_{eff}
289	4.73e-02	2.08e-03	−1.08e-03	−5.82e-04	4.19e-04	0.01
557	9.14e-03	2.09e-03	−3.61e-05	−7.04e-06	2.04e-03	0.22
1085	1.78e-03	4.78e-04	−1.25e-04	9.90e-06	3.63e-04	0.20
2069	3.11e-04	5.83e-06	−9.55e-05	−2.63e-07	−9.00e-05	−0.29
4709	7.28e-05	−3.11e-05	−4.17e-05	5.68e-08	−7.28e-05	−1.00
12465	1.50e-05	−1.87e-05	−2.64e-05	8.40e-08	−4.51e-05	−3.01
37987	8.47e-07	−9.78e-06	−1.08e-05	1.26e-07	−2.04e-05	−24.14

In order to understand the unsatisfactory efficiency index for this example, we use another way of approximation of e_k, $k = 1, 2, 3$. Instead of patchwise quadratic recovery $i_{2h}^{(2)} u_h$ for approximating e_2 and e_3, we compute a higher order solution according to (3.12) for e_2. The obtained results are listed in Table 4.5. For quadratic approximation in the estimator and globally refined meshes the efficiency index converges to one even for small numbers of periods ($T = 1$). This means that the estimator seems to be (in this nonlinear example) asymptotically exact. On locally refined meshes the efficiency index I_{eff} is slightly worse but still close to one and therefore acceptable. Hence, the accuracy of the estimator benefits from higher order computation but the adaptivity itself does not. The reason for the poor efficiency index in Table 4.4 is the underestimated value for e_3 which becomes large due to the short period $T = 1$. However, for local mesh refinement the higher order interpolation works well even for a single period.

4.3. Flow around a cylinder. As the last numerical example, we choose a configuration of a more realistic flow problem but without an analytical exact solution. We consider a flow around an obstacles which was defined by Schäfer and Turek [11] as benchmark problems within the DFG high-priority research program Flow simulation with high-performance computers. The configuration of the problem is presented in

TABLE 4.5

Navier–Stokes (subsection 4.2) with $T = 1$ and global and local mesh refinement. The error estimation is based on higher order computations for η_2', η_3'.

# nodes	$J(\bar{u} - \bar{u}_h)$	η_1	η_2'	η_3'	η	I_{eff}
	Global mesh refinement					
289	4.73e-02	2.08e-03	−3.25e-03	4.17e-03	3.00e-03	0.06
1089	8.96e-03	2.30e-03	−8.00e-04	2.80e-03	4.30e-03	0.48
4225	1.77e-03	6.64e-04	−1.56e-04	8.67e-04	1.38e-03	0.78
16641	3.67e-04	1.12e-04	−1.89e-05	2.49e-04	3.42e-04	0.93
66049	8.52e-05	2.09e-05	−3.56e-06	6.57e-05	8.33e-05	0.98
	Local mesh refinement					
289	4.73e-02	2.08e-03	−3.25e-03	4.17e-03	3.00e-03	0.06
759	8.21e-03	3.20e-04	1.40e-04	3.92e-03	4.38e-03	0.53
1675	1.74e-03	5.33e-05	6.84e-05	1.37e-03	1.49e-03	0.86
3085	2.40e-04	1.91e-04	−2.45e-05	1.92e-04	3.59e-04	1.50
7147	2.66e-05	−5.07e-05	−6.74e-05	1.75e-04	5.65e-05	2.12
14583	8.03e-06	1.77e-05	1.71e-06	−4.07e-06	1.53e-05	1.91
30231	−4.75e-06	−1.14e-04	−4.91e-05	1.58e-04	−4.79e-06	1.01

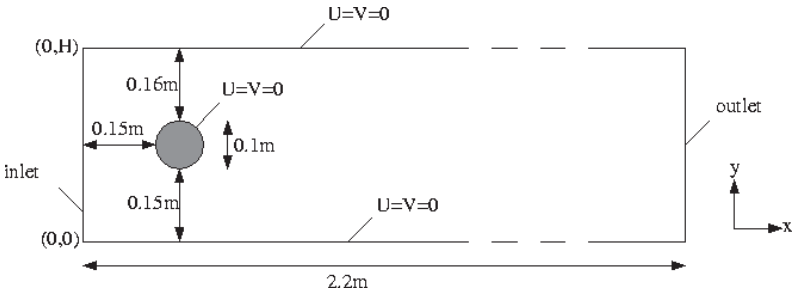

FIG. 4.5. *Configuration of the benchmark problem "Flow around a cylinder".*

Figure 4.5. The viscosity is set to $\nu = 10^{-3}$, the characteristic inflow velocity to $U_m = 1.5m/s$ as defined in [11] . We measure the drag coefficient c_{drag} in the time interval $I = [5\,s, 15\,s]$, where the solution has a periodic behavior. This time span corresponds to approximately 60 periods.

We computed a reference value solving the problem on fine grids with 40,000 and 160,000 nodes using Q_2 elements and extrapolating these values, leading to the reference mean drag $c_{drag}(\bar{u}) \approx 3.19613266$. The following computations were made using the Fractional-Step-θ scheme with an adequate small timestep of $\Delta t = 4.e-3\,s$.

The velocity components of the stationary dual solution are shown in Figure 4.6. Some of the obtained locally refined meshes are shown in Figure 4.7. The refinement is basically nearby the cylinder. In Figure 4.8 we have a look on the several contributions of the localized error contributions of η_1, η_2, and η_3. The linear part η_1 of the estimator measures mostly the error close to the cylinder and the nonlinear part η_2 is large behind the cylinder where the nonlinearities of the solution is apparently comparatively large. The third part η_3 can be neglected for this example as a result of the large timespan I considered.

In Table 4.6 the results for global and local mesh refinements are listed and a comparison of the error evolution is shown in Figure 4.9. Even for local mesh refinement we get a quadratic convergence order. The error for local refined meshes has for 18,344 nodes the same magnitude as for more than 164,000 nodes with global refine-

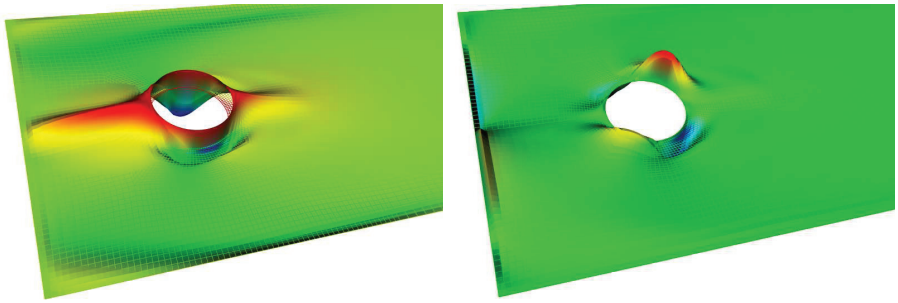

FIG. 4.6. *Horizontal (left) and vertical (right) velocity components of the stationary dual solution in the vicinity of the cylinder.*

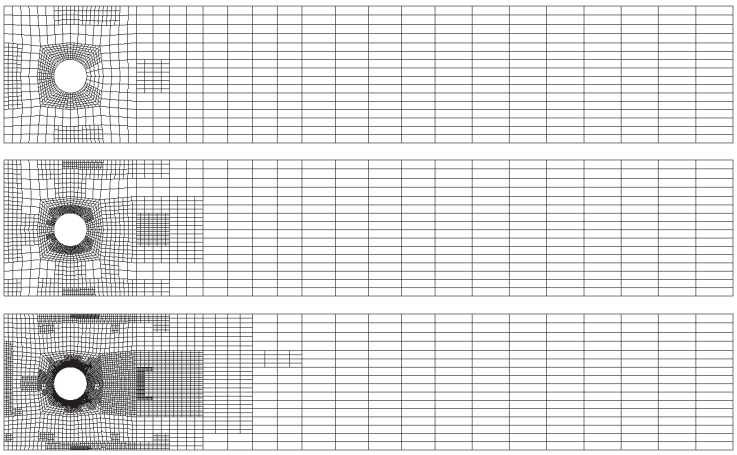

FIG. 4.7. *Local refined meshes after* 1, 2, *and* 3 *refinements for "Flow around a cylinder".*

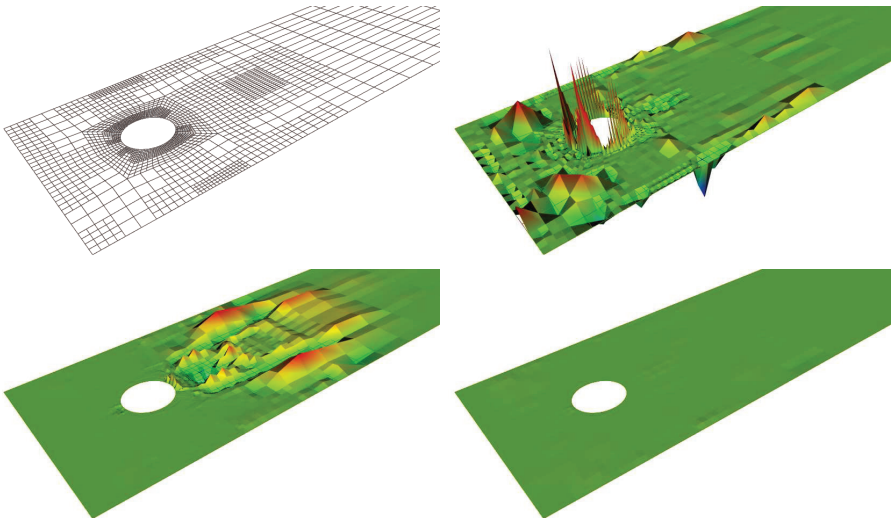

FIG. 4.8. *The mesh (upper left) and nodal error indicators* $\eta_{1,i}$ *(upper right),* $\eta_{2,i}$ *(lower left), and* $\eta_{3,i}$ *(lower right) after two refinements for "Flow around a cylinder". The scaling and colors scale identical for the three error contributions.*

TABLE 4.6
Results for global and local mesh refinement for "Flow around a cylinder".

# nodes	$J(\bar{u}-\bar{u}_h)$	η_1	η_2	η_3	η	I_{eff}
	Global mesh refinement					
708	2.99e-01	−3.62e-01	1.14e-02	1.75e-03	−3.49e-01	−1.16
2696	1.55e-01	−2.53e-02	4.04e-03	3.19e-04	−2.09e-02	−0.13
10512	3.21e-02	1.50e-02	9.95e-04	4.53e-05	1.60e-02	0.50
41504	7.58e-03	4.50e-03	3.20e-04	1.17e-05	4.83e-03	0.64
164928	1.86e-03	1.14e-03	9.90e-05	2.90e-06	1.24e-03	0.67
	Local mesh refinement					
708	2.99e-01	−3.62e-01	1.14e-02	1.75e-03	−3.49e-01	−1.16
1282	1.46e-01	−3.66e-02	1.13e-02	6.72e-04	−2.46e-02	−0.17
2610	3.27e-02	1.08e-02	5.36e-03	2.34e-04	1.64e-02	0.50
6698	8.32e-03	4.19e-03	1.54e-03	9.01e-05	5.82e-03	0.70
18344	1.66e-03	1.10e-03	5.11e-04	3.23e-05	1.64e-03	0.99
60448	4.61e-04	2.87e-04	1.63e-04	1.10e-05	4.60e-04	1.00

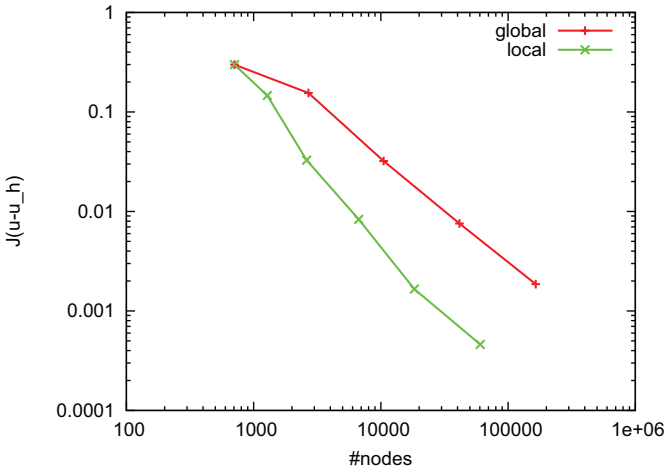

FIG. 4.9. *Comparison of global and local refinement for "Flow around a cylinder".*

ments, which is a huge reduction of computational costs. Furthermore, we get a pretty good efficiency index of approximately 0.7 for global mesh refinement. Moreover, the efficiency index for local mesh refinement is extremely close to 1. We do not claim that our error estimator is asymptotically exact. Rather, such a good estimation is by concurrence.

4.4. Stationary dual solution at high Reynolds number. For higher Reynolds number it is not clear whether it is reasonable to solve stationary dual problems. The linear system probably become ill-conditioned or even singular. In order to address this aspect, we finally consider the 2D lit-driven cavity problem. The maximal Reynolds number is chosen as $Re = 40.000$. The configuration is the standard 2D lit driven cavity problem with a time-depending but smooth overflow for the horizontal velocity component v_1 at $y = 1$:

$$v_1(t,x) = (1+\sin(-0.5\pi+2\pi t))(1+\sin(-0.5\pi+2\pi x))\,.$$

The functional under consideration is the mean drag force on the lower boundary. It is well known from stationary adaptive processes [1] that such a functional leads to a

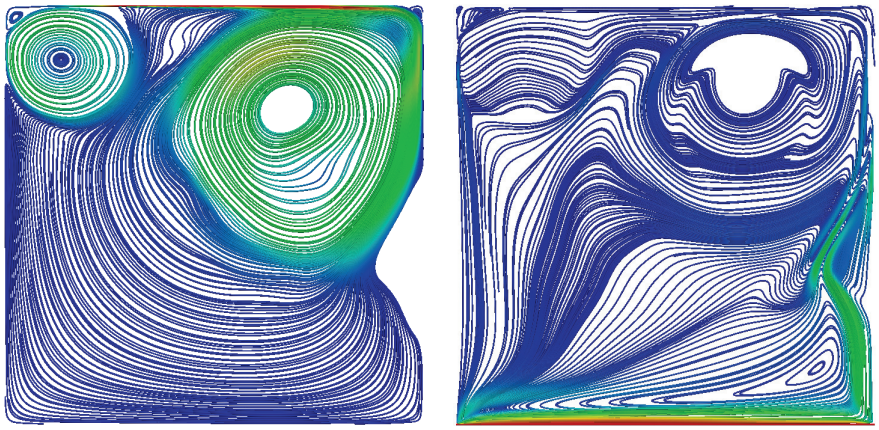

FIG. 4.10. *Left: Streamlines of the mean velocity field* $\overline{v}_h$. *Right: Streamlines of the stationary dual velocity field* z_h^v. *The colors represent the particular norm, i.e.,* $\|v_h(x)\|$ *and* $\|z_h^v(x)\|$, *respectively.*

TABLE 4.7
Number of GMRES iterations for the stationary dual problem for different levels of refinement and different Re numbers.

	Re		
# nodes	400	4.000	40.000
289	8	15	20
1089	10	20	28
4225	10	27	37
16641	11	26	40
66049	11	27	40

dual problem with zero right-hand side and nonhomogeneous boundary values (on the lower boundary). In Figure 4.10 we show streamlines of the averaged primal velocity field and of the stationary adjoint velocity field. Actually, $\overline{v}_h$ shows small vortices close to the upper corners.

In Table 4.7 we list the number of GMRES iterations, preconditioned with multigrid and incomplete LU (ILU) smoother, in dependency of the number of nodes and the Reynolds number. The relative tolerance is set to 10^{-6}. Due to the multigrid preconditioner, the number of linear iterations is independent of the mesh size. With respect to the Reynolds number, we observe a moderate increase of necessary iterations. However, even for the highest *Re* number (corresponding to a viscosity of $\mu = 5 \cdot 10^{-5}$), the stationary dual problem was easily solvable with 40 linear iterations. This corresponds to a notable convergence rate of 0.7. Taking in mind that in practice a much larger tolerance is reasonable, this test supports the feasibility of our approach. In the worst case, when the linear solver did not converge it would always be possible to perform a pseudotime stepping scheme (e.g., backward Euler) for the dual problem. Since we have averaged coefficients, this is straightforward and the time relaxed problem can always be solved.

5. Conclusion. We presented an error estimator and an adaptive algorithm for accurate determination of functional output of the mean value. The underlying equation is a general time-dependent (system of) partial differential equation. We avoid computations of time-dependent adjoint problems and storing of the primal

solution even for nonlinear equations. By several numerical examples we illustrate the accuracy of the estimator and the performance of the adaptive algorithm.

REFERENCES

[1] R. Becker and M. Braack, *Solution of a stationary benchmark problem for natural convection with high temperature difference*, Int. J. Therm. Sci., 41 (2002), pp. 428–439.

[2] R. Becker, M. Braack, D. Meidner, R. Rannacher, and B. Vexler, *Adaptive finite element methods for PDE-constrained optimal control problems*, in Reactive Flows, Diffusion and Transport, W. Jäger et al., eds., Springer, Berlin, 2006, pp. 177–205.

[3] R. Becker and R. Rannacher, *An optimal control approach to a posteriori error estimation in finite element methods*, Acta Numer., 10 (2001), pp. 1–102.

[4] O. Bendix and B. Vexler, *A posteriori error estimation and adaptivity for elliptic optimal control problems with state constraints*, Comput. Optim. Appl., 44 (2009), pp. 3–25.

[5] H. Blum and F.-T. Suttmeier, *An adaptive finite element discretisation for a simplified Signorini problem*, Calcolo, 37 (1999), pp. 65–77.

[6] M. Braack and T. Richter, *Solutions of 3D Navier-Stokes benchmark problems with adaptive finite elements*, Comput. & Fluids, 35 (2006), pp. 372–392.

[7] M. Braack and Th. Richter, *Solving multidimensional reactive flow problems with adaptive finite elements*, in Reactive Flows, Diffusion and Transport, W. Jäger et al., eds., Springer, Berlin, 2006, pp. 93–112.

[8] E. Burman, *Adaptive finite element methods for compressible flow*, Comput. Methods Appl. Mech. Engrg., 19 (2000), pp. 1137–1162.

[9] J. Hoffman, *Adaptive simulation of the subcritical flow past a sphere*, J. Fluid Mech., 568 (2006), pp. 77–88.

[10] J. Hoffman, *Efficient computation of mean drag for the subcritical flow past a circular cylinder using general Galerkin G2.*, Internat. J. Numer. Methods Fluids, 59 (2009), pp. 1241–1258.

[11] M. Schäfer and S. Turek, *Benchmark computations of laminar flow around a cylinder. (With support by F. Durst, E. Krause and R. Rannacher)*, in Flow Simulation with High-Performance Computers II. DFG priority research program results 1993–1995, E.H. Hirschel, ed., Notes Numer. Fluid Mech. 52, Vieweg, Wiesbaden, 1996, pp. 547–566.

[12] M. Schmich and B. Vexler, *Adaptivity with dynamic meshes for space-time finite element discretizations of parabolic equations*, SIAM J. Sci. Comput., 30 (2008), pp. 369–393.