# Computability and adaptivity in CFD

J. Hoffman[1] and C. Johnson[2]

[1] *Courant Institute, 251 Mercer Street, New York, NY-10012, USA*
[2] *Department of Computational Mathematics, Chalmers University, SE-41296 Göteborg, Sweden*

Abstract

We present an approach to Computational Fluid Dynamics CFD based on adaptive stabilized Galerkin finite element methods with duality based a posteriori error control for chosen output quantities of interest. We address the basic question of computability in CFD: For a given flow, what quantity is computable to what tolerance to what cost? We focus on incompressible Newtonian flow with medium to large Reynolds numbers involving both laminar and turbulent flow features. Our basic tool is a representation formula for the error in the quantity of interest in terms of a space-time integral of the residual of a computed solution multiplied by weights related to derivatives of the solution of an associated dual linearized problem with data connected to the output. We use the error representation formula to derive an a posterori error estimate combining residuals with computed dual weights, which is used for mesh adaptivity in space-time with the objective of satisfying a given error tolerance with minimal computational effort. We show in conrete examples that outputs such as mean values in time of drag and lift of turbulent flow around a bluff body are computable on a PC with a tolerance of a few percent using a few hundred thousand mesh points in space. We refer to our methodology as *Adaptive DNS/LES*, where automatically by adaptivity certain features of the flow are resolved in a *Direct Numerical Simulation DNS*, while certain other small scale turbulent features are left unresolved in a *Large Eddy Simulation LES*. The stabilization of the Galerkin method giving a weighted least square control of the residual acts as the subgrid model in the LES. The a posteriori error estimate takes into account both the error from discretization and the error from the subgrid model.

KEY WORDS: adaptivity, computability, adaptive finite element method, a posteriori error estimate, turbulence, incompressible flow, DNS, LES.

## 1. Introduction

The *Navier-Stokes equations* form the basic mathematical model in *fluid mechanics* and describe a large variety of phenomena of *fluid flow* occurring in hydro- and aero-dynamics, processing industry, biology, oceanography, meteorology, geophysics and astrophysics. Fluid flow may contain features of *incompressible* and *compressible* flow, *Newtonian* and *non-Newtonian* flow, and *turbulent* and *laminar* flow, with turbulent flow being irregular with rapid fluctuations in space and time and laminar flow being more organized. *Computational Fluid Dynamics* CFD concerns the digital/computational simulation of fluid flow by solving the Navier-Stokes equations numerically. In these notes we focus on CFD for laminar and turbulent incompressible Newtonian flow.

The basic issues of CFD is *computability* relating to errors from numerical computation, and *predictability* relating to errors from imprecision in given data. The basic question of computability/predictability for a given flow may be formulated as follows: what quantity can be

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

computed/predicted to what tolerance/norm to what cost? We emphasize the *quantitative* aspects concerning both the choice of *quantity of interest*, or *output*, the error tolerance/norm and the cost. For computability the cost reflects the precision of the computation with direct connection to the computational work (number of arithmetical operations and memory requirements), and for predictability the cost reflects the required precision of data. We expect a turbulent flow to be more costly than a laminar flow, and a pointwise quantity (e.g the viscous stresses at specific points) to be more costly than an average quantity (e.g. the drag or lift), and of course we expect the cost to increase with decreasing tolerance.

The *Reynolds number* $Re = \frac{UL}{\nu}$, where $U$ is a characteristic flow velocity, $L$ a characteristic length scale, and $\nu$ the viscosity of the fluid, is used to characterize different flow regimes. If $Re$ is relatively small ($Re \leq 10 - 100$), then the flow is viscous and the flow field is ordered and smooth or *laminar*, while for larger $Re$, the flow will at least partly be *turbulent* with time-dependent non-ordered features on a range of length scales down to a smallest scale of size to $Re^{-3/4}$, assuming $L = 1$. In many applications of scientific and industrial importance $Re$ is very large, of the order $10^6$ or larger, and the flow shows a combination of laminar and turbulent features. To accurately resolve all the features of a flow at $Re = 10^6$ in a *Direct Numerical Simulation DNS* would require of the order $Re^3 = 10^{18}$ mesh points in space-time, and thus would be impossible on any forseeable computer.

To overcome the impossibility of DNS at higher Reynolds numbers various techniques of *Large Eddy Simulation LES* have been attempted. In a LES the coarser scales of the flow are resolved by the mesh and one seeks to model the influence of the unresolved small scales on the resolved larger scales in a *subgrid model*. Various subgrid models have been proposed, but no clear answer to the question of the feasibility of LES in simulation of turbulence has been given. A main open problem of CFD today is the simulation of laminar/turbulent flow at high Reynolds numbers by some form of LES.

We present in these notes an approach to solve this problem based on adaptive stabilized Galerkin finite element methods with duality based a posteriori error control for chosen output quantities of interest. We thus address the basic question of computability for incompressible Newtonian flow at medium to large Reynolds numbers involving both laminar and turbulent flow features. Our basic tool is the stabilized Galerkin method together with an a posteriori error estimate derived from a representation of the error in output in terms of a space-time integral of the residual of a computed solution multiplied by weights related to derivatives of the solution of an associated dual linearized problem with data connected to the output. We use an adaptive procedure where we compute on a sequence of successively refined meshes with the objective of reaching a stopping criterion based on the a posteriori error estimate with minimal computational effort (minimal number of mesh points in space-time). We show in concrete examples that outputs such as a mean value in time of the drag of a bluff body in a laminar/turbulent flow is computable on a PC (with tolerances on the level of a few percent using a few hundred thousand mesh points in space).

The stabilized Galerkin method is the Galerkin/least squares space-time finite element method developed over the years together with Hughes, Tezduyar and coworkers, here referred to as the General Galerkin $\mathrm{G}^2$-method. This method includes the *streamline diffusion method* on Eulerian space-time meshes, the *characteristic Galerkin method* on Lagrangian space-time meshes with orientation along particle trajectories, and *Arbitrary Lagrangian-Eulerian ALE methods* with different mesh orientation. $\mathrm{G}^2$ offers a general flexible methodology for the discretization of the incompressible and compressible Navier-Stokes equations applicable to a great variety of flow problems from creeping low Reynolds number flow through medium to large Reynolds number turbulent flow, including free or moving boundaries. With continuous piecewise polynomials in space of order $p$ and discontinuous or continuous piecewise polynomials in time of order $q$, we refer to this method as cG(p)dG(q) or

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

cG(p)cG(q). Below we present computational results with cG(1)cG(1).

We describe the methodology for CFD presented in these notes as *Adaptive DNS/LES $G^2$*, where automatically by adaptivity certain features of the flow are pointwise resolved in a DNS, while certain other small scale features are left unresolved in a LES. The stabilization in $\mathrm{G}^2$ (adding a weighted least square control of the residual) acts as a subgrid model in LES introducing viscous damping of high frequencies. The $\mathrm{G}^2$ subgrid model is similar to a standard Smagorinsky model (adding a viscous term with viscosity proportional to $h^2|\epsilon(U_h)|$, where $h = h(x)$ is the local mesh size in space, $U_h$ the discrete computed solution, and $\epsilon(U_h)$ the strain tensor), but with less damping of low frequencies.

The a posteriori error estimate underlying the stopping criterion has (i) a contribution from the residual of the Galerkin solution inserted in the Navier-Stokes equations (which estimates the output error from the Galerkin discretization) and (ii) a contribution from the stabilization (which estimates the output error from the subgrid model). If we reach the stopping criterion, this means that the sum of (i) and (ii) are below the tolerance, and thus in particular that the contribution from the subgrid model to the output error is below the tolerance.

The $\mathrm{G}^2$ subgrid model may be viewed as adding a viscosity roughly of size $h(x)$ in areas of turbulence, and thus $\mathrm{G}^2$ will act as a LES if $h(x) > \nu$ in a turbulent area. We will show below that the mean drag of a bluff body in two benchmark problems at Reynolds numbers $22.000 - 40.000$, with $\nu \approx 10^{-5}$, is computable up to a tolerance of a few percent with $h(x) \approx 10^{-2}$ (thus $h(x) >> \nu$) in the turbulent wake, thus definitely using LES in part of the domain.

We thus show that certain outputs of a partly turbulent flow are computable without resolving all the small scale features of the turbulent flow by using a relatively simple subgrid model. The reason this is possible is that the output does not depend on all the exact details of the turbulent flow, which we observe by computing the contribution to the a posteriori error estimate from the subgrid model and noting that this contribution indeed is small. This is analogous to the observed fact that in a shock problem for compressible flow, certain outputs are computable without resolving the shocks to their actual width.

The fact that certain outputs do not critically depend on the exact nature of the subgrid model being used, does not mean that we can do without a subgrid model. The local intensity of dissipation in a turbulent flow is a characteristic feature of the flow which has to be captured in the subgrid model to its correct level, but certain outputs may be insensitive to the exact nature of the dissipation. More precisely, the dissipation in LES may occur at coarser scales than in the real flow, while the total dissipation is correct. We notice this phenomenon in the bluff body problems, where the intensity of the dissipation in the turbulent wake is nearly constant during the refinement process (captured in a LES), while the volume of the turbulent wake expands (captured in a DNS in the boundary layer surrounding the wake).

The key test of computability in Adaptive DNS/LES $\mathrm{G}^2$ of a certain output is thus the a posteriori error estimate combining residuals with dual weights. If indeed this combination is small enough, which we test computationally, then we reach the stopping criterion and thus we have computed the desired output to the given tolerance. Evidently, the size of the dual weights are here crucial: if the weights are too large, then we may not be able to reach the stopping criterion with available computing power. We observe that large mean value outputs such as drag and lift typically have smaller dual weights than more pointwise outputs. Altogether, we may say that the computability of a certain output directly couples to the size of the dual weights and thus by computing these weights the computability of a certain output can be assessed.

The dual problem is a linear convection-diffusion-reaction problem with the gradient $\nabla U_h$ acting as the coefficient in the reaction term. In turbulent flow $\nabla U_h$ will be large, and thus potentially generating exponential growth of the dual solution and very large dual weights. Nevertheless, the dual solution and the dual weights turn out to be of moderate size, which we observe by computing the dual solution, and which we intuitively may explain by the fact that $\nabla U_h$ is fluctuating with a combination of production and consumption in the reaction, with apparently only a moderate net production. This is the crucial fact behind the computability of certain output, which we may view as a fortunate "miracle" of CFD, and which may be very difficult to "understand" or rationalize by mathematical analysis, although we may successively get used to it by computing dual solutions and eventually possibly grasping it intuitively.

The key new ingredient in our work as presented in these notes is the use of duality to assess the basic problem of computability of a specific output for a specific flow. In particular, we show that the crucial dual solution may be computed at an affordable cost for complex time-dependent laminar/turbulent flows in 3d.

We may view the stabilized Galerkin method as producing an approximate solution of the Navier Stokes equations, with a residual which in a weak sense is small (reflecting the Galerkin orthogonality) and which is also controlled in a strong sense (reflecting the least squares stabilization). The existence and uniqueness of exact solutions to the Navier-Stokes equations is one of the major open problems of mathematical analysis today, as formulated one of the ten \$1 million Clay Prize problems. It is conceivable that new input to this problem may come from computed solutions with a posteriori error control. In particular, the dual weights carry sensitivity information which directly relates to the question of uniqueness.

In these notes we do not explicitely use any filtered or averaged Navier-Stokes equations for mean-values of the exact solution, as is often attempted in CFD. Such equations include so-called Reynolds stresses arising from the averaging of the nonlinear convection term, the modeling of which in a subgrid model represents an open problem. As indicated, we use in Adaptive DNS/LES $\mathrm{G}^2$ a subgrid model arising from the stabilization and this model may be viewed as a model of the Reynolds stresses, which is similar to a Smagorinsky model with dissipation mainly on the smallest computational scales. We estimate this term a posteriori and find that its effect on an output such as drag is small. We propose to take this as an indication that also the effect of the true Reynolds stresses is small, and thus conclude that a detailed Reynolds stress model is not required. We thus circumvent the basic open problem of CFD of a detailed modeling of the Reynolds stresses by giving concrete computational evidence that a crude such model is enough, if the output is a mean value such as drag. More precisely, in the adaptive process we estimate the output error contribution from the stabilization subgrid model and refine the mesh until this contribution is below the tolerance, and we take this as evidence that also the effect of the Reynolds stresses is below the tolerance. In fact, the Reynolds stresses with their dependence on the exact solution are inaccessible, and their detailed modeling may simply be impossible, but it may still be possible in an adaptive process to bound their influence on certain outputs.

## 2. Outline

We first recall the incompressible Navier-Stokes equations, and then we present $\mathrm{G}^2$. We then pass directly to a key benchmark problem of computing the mean value in time of the drag (or $c_D$ coefficient) of a square cylinder at Reynolds number 22.000, for which there are experimental reference values avaliable with $c_D = 1.9 - 2.1$. With Adaptive DNS/LES $\mathrm{G}^2$ we obtain a value $c_D \approx 2.0$ using about $10^5$ mesh points in space, with an estimated tolerance of about $10\%$.

Next we consider a surface mounted cube at Reynolds number 40.000. Although this may be viewed

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

as a generic bluff body problem, experimental values for $c_D$ do not seem to be avaliable and the only computational result prior to ours seems to be that of Krajnović and Davidson, 2002, presenting results in the range $c_D = 1.12 - 1.24$. We obtain with Adaptive DNS/LES $\mathrm{G}^2$ a value $c_D \approx 1.45$ using a few hundred thousand mesh points in space, with again an estimated tolerance of about $10\%$. Our value thus appears to definitely be larger than that of Krajnović and Davidson, 2002. The present lack of knowledge of the exact value makes this problem into an objective test problem. We propose this problem as a basic benchmark of CFD, and invite to submittance of computational results to the *CDE-Forum* (`http://www.phi.chalmers.se/cdeforum/`).

We then pass on to a computational study of computability of various space-time averages of the fluid velocity in transition to turbulence in shear flow, and we conclude with results for some benchmark problems for stationary laminar flow.

All computations are performed with the open-source software *DOLFIN*, developed by Hoffman and Logg using $\mathrm{G}^2$. For further information and latest updates, see the DOLFIN homepage (`http://www.phi.chalmers.se/dolfin/`).

## 3. References

For an overview of adaptive finite element methods including references, we refer to the survey articles Eriksson *et al.*, 1995, Becker and Rannacher, 2001, and the books Eriksson *et al.*, 2003, and Barth *et al.*, 2004, containing many details on various aspects of adaptive finite element methods omitted in these notes. For an overview of finite element methods for the incompressible Navier-Stokes equations including references, we refer to Rannacher, 1999, and for an overview of computational methods for turbulence we refer to Sagaut, 2001, and the references therein.

For incompressible flow, applications of adaptive finite element methods based on this framework have been increasingly advanced with computation of quantities of interest such as the drag force for 2d stationary benchmark problems in Becker and Rannacher, 1996; Giles *et al.*, 1997, and drag and lift forces and pressure differences for 3d stationary benchmark problems in Hoffman, 2003. In Hoffman and Johnson, 2002a, time dependent problems in 3d are considered, and the extension of this framework to LES is investigated in Hoffman, 2002a; Hoffman, 2002b. This extension is crucial and opens for a large wealth of real world applications. The generalization to *Adaptive DNS/LES* is presented in Hoffman and Johnson, 2003, and the computational examples in this paper using Adaptive DNS/LES are taken from Hoffman, 2003a; Hoffman, 2003b.

## 4. The incompressible Navier-Stokes equations

The incompressible Navier-Stokes equations expressing conservation of momentum and incompressibility of a unit density constant temperature Newtonian fluid with constant *kinematic viscosity* $\nu > 0$ enclosed in a volume $\Omega$ in $\mathbb{R}^3$, take the form: find $(u, p)$ such that

$$\begin{aligned} D_{u,t}u - \nu\Delta u + \nabla p &= f && \text{in } \Omega \times I, \\ \operatorname{div} u &= 0 && \text{in } \Omega \times I, \\ u &= w && \text{on } \partial\Omega \times I, \\ u(\cdot, 0) &= u^0 && \text{in } \Omega, \end{aligned} \tag{1}$$

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

where $u(x,t) = (u_i(x,t))$ is the *velocity* vector and $p(x,t)$ the *pressure* of the fluid at $(x,t)$, and $f$, $w$, $u^0$, $I = (0,T)$, is a given driving force, Dirichlet boundary data, initial data and time interval, respectively. Further,

$$D_{u,t}v \equiv \dot{v} + (u \cdot \nabla)v \tag{2}$$

is the *particle derivative* of $v(x,t)$ measuring the rate of change $\frac{d}{dt}v(x(t),t)$ of $v(x(t),t)$ along the trajectory $x(t)$ of a fluid particle with velocity $u$, satisfying $\dot{x}(t) = u(x(t),t)$, where as usual $\dot{v} = \partial v/\partial t$. The quantity $\nu\Delta u - \nabla p$ represents the total fluid force, and may alternatively be expressed as

$$\nu\Delta u - \nabla p = \operatorname{div}\sigma(u,p), \tag{3}$$

where $\sigma(u,p) = (\sigma_{ij}(u,p))$ is the *stress tensor*, with components $\sigma_{ij}(u,p) = 2\nu\epsilon_{ij}(u) - p\delta_{ij}$, composed of the *stress deviatoric* $2\nu\epsilon_{ij}(u)$ with zero trace and an isotropic pressure. Here $\epsilon_{ij}(u) = (u_{i,j} + u_{j,i})/2$ is the *strain tensor*, with $u_{i,j} = \partial u_i/\partial x_j$, and $\delta_{ij}$ is the usual Kronecker delta, the indices $i$ and $j$ ranging from 1 to 3. A Neumann type boundary condition, corresponding to the boundary stress being prescribed, takes the form $\sigma \cdot n = g$, where $(\sigma \cdot n)_i = \sum_j \sigma_{ij} n_j$ and $g = (g_i)$ is a given boundary stress with $g_i$ the force component in the $x_i$-direction.

In the model (1) we assume that the *temperature* $T$ is constant. In the general case with variable density $\rho$ and temperature $T$, (1) is modified by replacing $D_{u,t}$ by $\rho D_{u,t}$, and adding the following equations expressing conservation of mass and energy:

$$\begin{aligned} D_{u,t}\rho &= 0 \qquad && \text{in } \Omega \times I, \\ D_{u,t}T - \nabla \cdot (\mu \nabla T) &= F \qquad && \text{in } \Omega \times I, \end{aligned} \tag{4}$$

together with boundary and initial conditions, where $\mu$ is a heat conduction coefficient and $F$ a heat source, assuming the heat capacity is equal to one. We note that since $\nabla \cdot u = 0$, we have $D_{u,t}\rho = \dot{\rho} + \nabla \cdot (\rho u) = 0$, which is the usual equation expressing mass conservation.

We assume that (1) is normalized so that the reference velocity and typical length scale are both equal to one. The Reynolds number $Re$ is then equal to $\nu^{-1}$. Of course, the specification of the length scale may not be very obvious and thus the Reynolds number may not have a very precise quantitative meaning.

## 5. Discretization: General Galerkin $G^2$

In this section we present the general space-time Galerkin least squares stabilized finite element method, referred to as the General Galerkin $G^2$-method, for the incompressible Navier-Stokes equations (1). This method includes the *streamline diffusion method* on Eulerian space-time meshes, the *characteristic Galerkin method* on Lagrangian space-time meshes with orientation along particle trajectories, and *Arbitrary Lagrangian-Eulerian ALE methods* with different mesh orientation. Further, the least-squares stabilizations present in the $G^2$-method, does take care of the two difficulties traditionally met in the discretization of the incompressible Navier-Stokes equations, namely

- instabilities from Eulerian discretization of convection terms,
- pressure instabilities in equal order interpolation of velocity and pressure.

Altogether, $G^2$ offers a general flexible methodology for the discretization of the incompressible Navier-Stokes equations applicable to a great variety of flow problems from creeping viscous flow to slightly viscous flow, including free or moving boundaries.

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

Let $0 = t_0 < t_1 < ... < t_N = T$ be a sequence of discrete time steps with associated time intervals $I_n = (t_{n-1}, t_n]$ of length $k_n = t_n - t_{n-1}$ and space-time slabs $S_n = \Omega \times I_n$, and let $W_n \subset H^1(\Omega)$ be a finite element space consisting of continuous piecewise polynomials of degree $p$ on a mesh $\mathcal{T}_n = \{K\}$ of mesh size $h_n(x)$ with $W_{0n}$ the functions in $W_n$ vanishing on $\Gamma$, the boundary of $\Omega$. To define the $G^2$-method for (1) with homogeneuos Dirichlet boundary conditions for the velocity ($w = 0$), let for a given velocity field $\beta$ on $S_n = \Omega \times I_n$ vanishing on $\Gamma \times I_n$, the particle paths $x(\bar{x}, \bar{t})$ be defined by

$$\begin{aligned} \frac{dx}{d\bar{t}} &= \beta(x, \bar{t}) \quad \bar{t} \in I_n, \\ x(\bar{x}, t_n) &= \bar{x}, \quad \bar{x} \in \Omega, \end{aligned} \tag{5}$$

and introduce the corresponding mapping $F_n^\beta : S_n \to S_n$ defined by $(x, t) = F_n^\beta(\bar{x}, \bar{t}) = (x(\bar{x}, \bar{t}), \bar{t})$, where $x = x(\bar{x}, \bar{t})$ satisfies (5). Define for a given $q \geq 0$, the spaces

$$\bar{V}_n^\beta = \{\bar{v} \in H^1(S_n)^3 : \bar{v}(\bar{x}, \bar{t}) = \sum_{j=0}^{q} (\bar{t} - t_n)^j U_{h,j}(\bar{x}),\, U_{h,j} \in [W_{0n}]^3\},$$

$$\bar{Q}_n^\beta = \{\bar{q} \in H^1(S_n) : \bar{q}(\bar{x}, \bar{t}) = \sum_{j=0}^{q} (\bar{t}_n - t_n)^j q_j(\bar{x}),\, q_j \in W_n\},$$

together with their analogs in $(x, t)$-coordinates:

$$V_n^\beta = \{v : \bar{v} \in \bar{V}_n^\beta\}, \qquad Q_n^\beta = \{q : \bar{q} \in \bar{Q}_n^\beta\}, \tag{6}$$

where $v(x, t) = \bar{v}(\bar{x}, \bar{t})$ and $q(x, t) = \bar{q}(\bar{x}, \bar{t})$. Defining finally $V^\beta \times Q^\beta = \prod_n V_n^\beta \times Q_n^\beta$, we can now formulate the $G^2$-method as follows: Find $(U_h, P_h) \in V^\beta \times Q^\beta$, such that for $n = 1, 2, ..., N$,

$$\begin{aligned} &(\dot{U}_h + (U_h \cdot \nabla)U_h, v)_n - (P_h, \operatorname{div} v)_n + (q, \operatorname{div} U_h)_n + (2\nu\epsilon(U_h), \epsilon(v))_n \\ &+ (\delta_1 a(U_h; U_h, P_h), a(U_h; v, q))_n + (\delta_2 \operatorname{div} U_h, \operatorname{div} v)_n + ([U_h^{n-1}], v_+^{n-1}) \\ &= (f, v + \delta_1 a(U_h; v, q))_n \quad \forall (v, q) \in V_n^\beta \times Q_n^\beta, \end{aligned} \tag{7}$$

where $a(w; v, q) = D_{w,t} v + \nabla q - \nu \Delta v$ with the Laplacian defined elementwise, $\delta_1 = \frac{1}{2}(k_n^{-2} + |U_h|^2 h_n^{-2})^{-1/2}$ in the convection-dominated case $\nu < U_h h_n$ and $\delta_1 = \kappa_1 h^2$ otherwise, $\delta_2 = \kappa_2 h$ if $\nu < U_h h_n$ and $\delta_2 = \kappa_2 h^2$ otherwise, with $\kappa_1$ and $\kappa_2$ positive constants of unit size, and

$$(v, w)_n = \int_{I_n} (v, w) dt, \;\; (v, w) = \sum_{K \in \mathcal{T}_n} \int_K v \cdot w \, dx,$$

$$(\epsilon(v), \epsilon(w))_n = \sum_{i,j=1}^{3} (\epsilon_{ij}(v), \epsilon_{ij}(w))_n.$$

Further, $[v^n] = v_+^n - v_-^n$ is the jump across the time level $t_n$ with $v_\pm^n$ the limit from $t > t_n / t < t_n$. In the Eulerian *streamline diffusion method* we choose $\beta = 0$, which means that the mesh does not move in time. The *characteristic Galerkin method* is obtained by choosing $\beta = U_h$ (and then $\delta_1 = \kappa_1 h^2$), which means that the mesh moves with the fluid particles. We may also choose $\beta$ differently which gives various versions of ALE-methods, with the mesh and particle velocity being (partly) different; for example we may move the mesh with the particle velocity at a free boundary, while allowing the mesh to move differently inside the domain. Further, Neumann boundary conditions may be implemented in the usual variational form.

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

The variational formulation (7) with $\delta_1 = \delta_2 = 0$ is obtained by multiplying the momentum equation by $v$, integrating over $S_n$ including integration by parts, and adding the incompressibility equation multiplied by $q$ and integrating over $S_n$. Choosing $\delta_1$ and $\delta_2$ positive as indicated introduces stabilizing least-squares terms. Note that the viscous term $(2\nu\epsilon(U_h), \epsilon(v))_n$ may alternatively occur in the form $(\nu\nabla U_h, \nabla v)_n = \sum_{i=1}^{3}(\nu\nabla U_{h,i}, \nabla v_i)_n$. In the case of Dirichlet boundary conditions the corresponding variational formulations will be equivalent, but will generate slightly different Neumann boundary conditions. Note also that we may write the term $-(P_h, \mathrm{div}\, v)$ alternatively in the form $(\nabla P_h, v)$ if $v$ vanishes on the boundary.

**Remark 1.** *In extreme situations with very large velocity gradients, we may add residual dependent shock-capturing artificial viscosity, replacing $\nu$ by $\hat{\nu} = \max(\nu, \kappa_3|R(U_h, P_h)|h^2)$, where $R(U_h, P_h) = \sum_{i=1}^{4} R_i(U_h, P_h)$ with*

$$\begin{array}{rcl} R_1(U_h, P_h) & = & |\dot{U}_h + U_h \cdot \nabla U_h + \nabla P_h - f - \nu\Delta U_h|, \\ R_2(U_h, P_h) & = & \nu D_2(U_h), \\ R_3(U_h, P_h) & = & |[U_h^{n-1}]|/k_n \quad \text{on } S_n, \\ R_4(U_h, P_h) & = & |\mathit{div}\, U_h|, \end{array} \tag{8}$$

*and where*

$$D_2(U_h)(x,t) = \max_{y \in \partial K}(h_n(x))^{-1}|[\frac{\partial U_h}{\partial n}(y,t)]| \tag{9}$$

*for $x \in K$, with $[\cdot]$ the jump across the element edge $\partial K$, and $\kappa_3$ is a positive constant of unit size. Note that $R_1(U_h, P_h)$ is defined elementwise and that with piecewise linears in space, the Laplacian $\Delta U_h$ is zero. In the computations presented below, we chose $\kappa_3 = 0$ corresponding to shutting off the artificial viscosity. Note that $R_1(U_h, P_h) + R_2(U_h, P_h)$ bounds the residual of the momentum equation, with the Laplacian term bounded by the second order difference quotient $D_2(U_h)$ arising from the jumps of normal derivatives across element boundaries.*

**Remark 2.** *The special case of the Stokes equations is of course obtained omitting the nonlinear terms $(U_h \cdot \nabla)U_h$ and $(U_h \cdot \nabla)v$, and setting $\delta_1 = \kappa_1 h^2$, $\delta_2 = \kappa_2 h^2$. This method contains the pressure stabilizing term $(\delta_1 \nabla P_h, \nabla q)$, which corresponds to a weighted Laplacian equation for the pressure in terms of the velocity.*

**Remark 3.** *Since in the local Lagrangean coordinates $(\bar{x}, \bar{t})$ on each slab $S_n$ with $\beta = U_h$,*

$$\frac{\partial \bar{U}_h}{\partial \bar{t}} \equiv \frac{\partial}{\partial \bar{t}} U_h(x(\bar{x}, \bar{t}), \bar{t}) = \dot{U}_h + U_h \cdot \nabla U_h,$$

*the convection term $U_h \cdot \nabla U_h$ effectively dissappears in the characteristic Galerkin method, when expressed in the characteristic coordinates $(\bar{x}, \bar{t})$, and thus the discrete equations on each time step effectively correspond to a Stokes problem.*

**Remark 4.** *The order of the $G^2$-method with polynomials of degree $p$ in space/time is generally $p + 1/2$, see Eriksson et al., 1996. The time stepping method in (7) is dG(q), the discontinuous Galerkin method with piecewise polynomials of order q, which is of order $2q + 1$ seen as an ODE-solver.*

### *5.1. The Eulerian cG(1)dG(0) method*

We now consider the the $G^2$-method (7) with $p = 1$, $q = 0$ and $\beta = 0$ for (1), which is the Eulerian cG(1)dG(0) method with continuous piecewise linears in space (cG(1)) and piecewise constants in time (dG(0)) corresponding to the backward Euler method. We then seek an approximate velocity $U_h(x,t)$ such that $U_h(x,t)$ is continuous and piecewise linear in $x$ for each $t$, and $U_h(x,t)$ is piecewise constant in $t$ for each $x$. Similarly, we seek an approximate pressure $P_h(x,t)$ which is continuous piecewise linear in $x$ and piecewise constant in $t$. More precisely, we seek $U_h^n \in V_n^0 = W_{0n}^3$ and $P_h^n \in Q_n^0 = W_n$ for $n = 1, ..., N$, and we define

$$\begin{aligned} U_h(x,t) &= U_h^n(x) \quad x \in \Omega, \quad t \in (t_{n-1}, t_n], \\ P_h(x,t) &= P_h^n(x) \quad x \in \Omega, \quad t \in (t_{n-1}, t_n]. \end{aligned} \tag{10}$$

We can now write the cG(1)dG(0) method without stabilization as follows: For $n = 1, ..., N$, find $(U_h^n, P_h^n) \in V_n^0 \times Q_n^0$ such that

$$\begin{aligned} &(\frac{U_h^n - U_h^{n-1}}{k_n}, v) + (U_h^n \cdot \nabla U_h^n + \nabla P_h^n, v) + (\nabla \cdot U_h^n, q) \\ &+ (\nu \nabla U_h^n, \nabla v) = (f^n, v) \quad \forall (v,q) \in V_n^0 \times Q_n^0, \end{aligned} \tag{11}$$

The cG(1)dG(0) method with $\delta_1$-stabilization takes the form: For $n = 1, ..., N$, find $(U_h^n, P_h^n) \in V_n^0 \times Q_n^0$ such that

$$\begin{aligned} &(\frac{U_h^n - U_h^{n-1}}{k_n}, v) + (U_h^n \cdot \nabla U_h^n + \nabla P_h^n, v + \delta_1(U_h^n \cdot \nabla v + \nabla q)) + (\nabla \cdot U_h^n, q) \\ &+ (\nu \nabla U_h^n, \nabla v) = (f^n, v + \delta_1(U_h^n \cdot \nabla v + \nabla q)) \quad \forall (v,q) \in V_n^0 \times Q_n^0, \end{aligned} \tag{12}$$

where $\delta_1 = \frac{1}{2}(k_n^{-2} + |U_h|^2 h_n^{-2})^{-1/2}$ in the convection-dominated case $\nu < U_h h_n$. Note that if $k \approx \frac{h_n}{U_h}$, which is a natural choice of time step respecting a CFL-condition, then $\delta_1 \approx \frac{h_n}{U_h}$. The stabilized form of the cG(1)dG(0) method is obtained by replacing $v$ by $v + \delta_1(U_h^n \cdot \nabla v + \nabla q)$ in the terms $(U_h^n \cdot \nabla U_h^n + \nabla P_h^n, v)$ and $(f^n, v)$. In principle, we should make the replacement throughtout, but in the present case of the cG(1)dG(0) only the indicated terms get involved because of the low order of the approximations. The perturbation in the stabilized method is of size $\delta_1$, and thus the stabilized method has the same order as the original method (first order in $h$ if $k \sim h$).

Letting $v$ vary in (12) while choosing $q = 0$, we get the following equation (the discrete momentum equation):

$$\begin{aligned} &(\frac{U_h^n - U_h^{n-1}}{k_n}, v) + (U_h^n \cdot \nabla U_h^n + \nabla P_h^n, v + \delta_1 U_h^n \cdot \nabla v) \\ &+ (\nu \nabla U_h^n, \nabla v) = (f^n, v + \delta_1 U_h^n \cdot \nabla v) \quad \forall v \in V_n^0, \end{aligned} \tag{13}$$

and letting $q$ vary while setting $v = 0$, we get the following discrete "pressure equation"

$$(\delta_1 \nabla P_h^n, \nabla q)) = -(\delta_1 U_h^n \cdot \nabla U_h^n, \nabla q) - (\nabla \cdot U_h^n, q) + (\delta_1 f^n, \nabla q) \quad \forall q \in Q_n^0. \tag{14}$$

The cG(1)dG(0) has a backward Euler first order accurate time stepping, and thus in general is too dissipative.

### 5.2. *The Eulerian cG(1)cG(1) method*

The cG(1)cG(1) method is a variant of $G^2$ using the continuous Galerkin method cG(1) in time instead of a discontinuous Galerkin method. With cG(1) in time the trial functions are continuous piecewise linear and the test functions piecewise constant. cG(1) in space corresponds to both test functions and

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

trial functions being continuous piecewise linear. Let $0 = t_0 < t_1 < ... < t_N = T$ be a sequence of discrete time steps with associated time intervals $I_n = (t_{n-1}, t_n]$ of length $k_n = t_n - t_{n-1}$ and space-time slabs $S_n = \Omega \times I_n$, and let $W^n \subset H^1(\Omega)$ be a finite element space consisting of continuous piecewise linear functions on a mesh $\mathcal{T}_n = \{K\}$ of mesh size $h_n(x)$ with $W^n_w$ the functions $v \in W^n$ satisfying the Dirichlet boundary condition $v|_{\Gamma_D} = w$.

We now seek functions $(U_h, P_h)$, continuous piecewise linear in space and time, and the cG(1)cG(1) method for the Navier-Stokes equations (1), with homogeneous Dirichlet boundary conditions reads: For $n = 1, ..., N$, find $(U_h^n, P_h^n) \equiv (U_h(t_n), P_h(t_n))$ with $U_h^n \in V_0^n \equiv [W_0^n]^3$ and $P_h^n \in W^n$, such that

$$\begin{aligned}((U_h^n - U_h^{n-1})k_n^{-1} + \hat{U}_h^n \cdot \nabla \hat{U}_h^n, v) + (2\nu\epsilon(\hat{U}_h^n), \epsilon(v)) - (P_h^n, \nabla \cdot v)\\ + (\nabla \cdot \hat{U}_h^n, q) + SD(\delta, \hat{U}_h^n, P_h^n, v, q) = (f, v) \quad \forall (v, q) \in V_0^n \times W^n,\end{aligned} \tag{15}$$

where $\hat{U}_h^n = \frac{1}{2}(U_h^n + U_h^{n-1})$, with the stabilizing term

$$SD(\delta, \hat{U}_h^n, P_h^n, v, q) \equiv (\delta_1(\hat{U}_h^n \cdot \nabla \hat{U}_h^n + \nabla P_h^n - f), \hat{U}_h^n \cdot \nabla v + \nabla q) + (\delta_2 \nabla \cdot \hat{U}_h^n, \nabla \cdot v),$$

where $\delta_1 = \frac{1}{2}(k_n^{-2} + |U|^2 h_n^{-2})^{-1/2}$ in the convection-dominated case $\nu < \hat{U}_h^n h_n$ and $\delta_1 = \kappa_1 h^2$ otherwise, $\delta_2 = \kappa_2 h$ if $\nu < \hat{U}_h^n h_n$ and $\delta_2 = \kappa_2 h^2$ otherwise, with $\kappa_1$ and $\kappa_2$ positive constants of unit size.

In the implementation of Adaptive DNS/LES used in the computations below, we use cG(1)cG(1) with the least squares stabilization acting as a dissipative subgrid model.

## 6. Adaptive computation of the drag of a bluff body

We want to compute a mean value in time of the drag of a bluff body in a channel subject to a time-dependent turbulent flow:

$$N(\sigma(u, p)) \equiv \frac{1}{|I|} \int_I \int_{\Gamma_0} \sum_{i,j=1}^{3} \sigma_{ij}(u, p) n_j \phi_i \, ds \, dt, \tag{16}$$

where $(u, p)$ solves (1) in the fluid volume $\Omega$ surrounding the body (using suitable boundary conditions as specified below), $\Gamma_0$ is the surface of the body in contact with the fluid, and $\phi$ is a unit vector along the channel in the direction of the flow. We first derive an alternative expression for the drag, which naturally fits with a Galerkin formulation, by extending $\phi$ to a function $\Phi$ defined in the fluid volume $\Omega$ and being zero on the remaining boundary $\Gamma_1$ of the fluid volume. Multiplying the momentum equation in (1) by $\Phi$ and integrating by parts, we get using the zero boundary condition on $\Gamma_1$

$$\begin{aligned}N(\sigma(u, p)) \quad = \quad & \frac{1}{|I|} \int_I (\dot{u} + u \cdot \nabla u - f, \Phi) - (p, \nabla \cdot \Phi)\\ & + (2\nu\epsilon(u), \epsilon(\Phi)) + (\nabla \cdot u, \Theta) \, dt,\end{aligned} \tag{17}$$

where we also added the integral of $\nabla \cdot u = 0$ multiplied with a function $\Theta$. Obviously, the representation does not depend on the particular extension $\Phi$ of $\phi$, or $\Theta$.

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

We are thus led to compute an approximation of the drag $N(\sigma(u,p))$ from a computed $(U_h, P_h)$ using the formula

$$N^h(\sigma(U_h,P_h)) = \frac{1}{|I|}\int_I (\dot U_h + U_h\cdot\nabla U_h - f,\Phi) - (P_h,\nabla\cdot\Phi) + (2\nu\epsilon(U_h),\epsilon(\Phi)) + SD(\delta,U_h,P_h,\Phi,\Theta) + (\nabla\cdot U_h,\Theta)\,dt, \tag{18}$$

where now $\Phi$ and $\Theta$ are finite element functions (with as before $\Phi = \phi$ on $\Gamma_0$ and $\Phi = 0$ on $\Gamma_1$), and where $\dot U_h = (U_h^n - U_h^{n-1})/k_n$ on $I_n$. By the Galerkin orthogonality (15), it follows that $N^h(\sigma(U_h,P_h))$ is independent of the choice of $(\Phi,\Theta)$ (assuming for simplicity that we use Dirichlet boundary conditions).

### 6.1. *The dual problem*

We introduce the following linearized dual problem: Find $(\varphi,\theta)$ with $\varphi = \phi$ on $\Gamma_0$ and $\varphi = 0$ on $\Gamma_1$, such that

$$\begin{aligned} -\dot\varphi - (u\cdot\nabla)\varphi + \nabla U_h\cdot\varphi - \nu\Delta\varphi + \nabla\theta &= 0 && \text{in } \Omega\times I,\\ \operatorname{div}\varphi &= 0 && \text{in } \Omega\times I,\\ \varphi(\cdot,T) &= 0 && \text{in } \Omega, \end{aligned}\tag{19}$$

where $(\nabla U_h\cdot\varphi)_j = (U_h)_{,j}\cdot\varphi$. We notice that the dual problem is a linear convection-diffusion-reaction problem where the convection acts backward in time and in the opposite direction of the exact flow velocity $u$. We further note that the coefficient $\nabla U_h$ of the reaction term locally is large in turbulent regions, and thus potentially generating rapid exponential growth. However, $\nabla U_h$ is fluctuating and the net effect of the reaction term turns out to generate slower growth, as we learn from computing approximations of the dual solution.

We notice the presence of both the exact velocity $u$ and a computed velocity $U_h$ as coefficients in the dual problem. Below we will compute approximations of the dual solution where we replace $u$ by $U_h$ in the dual problem, an issue which we discuss below.

### 6.2. *An error representation*

To derive a representation of the error $N(\sigma(u,p)) - N^h(\sigma(U_h,P_h))$, we subtract (18) from and (17) with $(\Phi,\Theta)$ finite element functions, to get

$$\begin{aligned} &N(\sigma(u,p)) - N^h(\sigma(U_h,P_h)) = \frac{1}{|I|}\int_I (\dot u + u\cdot\nabla u,\Phi) - (p,\nabla\cdot\Phi)\\ &+(2\nu\epsilon(u),\epsilon(\Phi)) + (\nabla\cdot u,\Theta) - ((\dot U_h + U_h\cdot\nabla U_h,\Phi) - (P_h,\nabla\cdot\Phi)\\ &+(2\nu\epsilon(U_h),\epsilon(\Phi)) + (\nabla\cdot U_h,\Theta) + SD(\delta,U_h,P_h,\Phi,\Theta))\,dt. \end{aligned}\tag{20}$$

With $(\varphi,\theta)$ the solution to the dual problem (19), we also have that

$$\begin{aligned} &\frac{1}{|I|}\int_I (\dot u + u\cdot\nabla u,\varphi) - (p,\nabla\cdot\varphi) + (2\nu\epsilon(u),\epsilon(\varphi)) + (\nabla\cdot u,\theta)\\ &-((\dot U_h + U_h\cdot\nabla U_h,\varphi) - (P_h,\nabla\cdot\varphi) + (2\nu\epsilon(U_h),\epsilon(\varphi)) + (\nabla\cdot U_h,\theta))\,dt\\ &= \frac{1}{|I|}\int_I -(\dot\varphi,e) + (u\cdot\nabla e + e\cdot\nabla U_h,\varphi) - (p-P_h,\nabla\cdot\varphi)\\ &+(2\nu\epsilon(e),\epsilon(\varphi)) + (\nabla\cdot e,\theta)\,dt = 0, \end{aligned}\tag{21}$$

using partial integration with $\varphi(T) = e(0) = 0$, where $e = u - U_h$, and that $(u \cdot \nabla)u - (U_h \cdot \nabla)U_h = (u \cdot \nabla)e + (e \cdot \nabla)U_h$. By (20) and (21), we thus have that

$$
\begin{aligned}
&N(\sigma(u,p)) - N^h(\sigma(U_h,P_h)) = \frac{1}{|I|}\int_I (\dot{U}_h + U_h \cdot \nabla U_h, \varphi - \Phi) \\
&-(P_h, \nabla \cdot (\varphi - \Phi)) + (\nabla \cdot U_h, \theta - \Theta) + (2\nu\epsilon(U_h), \nabla(\varphi - \Phi)) + SD(\delta, U_h, P_h, \Phi, \Theta) \\
&-((\dot{u} + u \cdot \nabla u, \varphi - \Phi) - (p, \nabla \cdot (\varphi - \Phi)) + (\nabla \cdot u, \theta - \Theta) + (2\nu\epsilon(u), \nabla(\varphi - \Phi)))\, dt \\
&= \frac{1}{|I|}\int_I (\dot{U}_h + U_h \cdot \nabla U_h - f, \varphi - \Phi) - (P_h, \nabla \cdot (\varphi - \Phi)) \\
&+(\nabla \cdot U_h, \theta - \Theta) + (2\nu\epsilon(U_h), \nabla(\varphi - \Phi)) + SD(\delta, U_h, P_h, \Phi, \Theta)\, dt,
\end{aligned}
$$

using partial integration with $\varphi = \Phi = \phi$ on $\Gamma_0$ and $\varphi = \Phi = 0$ on $\Gamma_1$. We have now proved the following error representation, where we express the total error as a sum of error contributions from the different elements $K$ in space (assuming here for simplicity that the space mesh is constant in time), and we use the subindex $K$ to denote integration over element $K$ so that $(\cdot, \cdot)_K$ denotes the appropriate $L_2(K)$ inner product:

**Theorem 5.** *If $(u,p)$ is the exact Navier-Stokes solution, $(U_h, P_h)$ is a cG(1)cG(1) solution, and $(\varphi, \theta)$ is the corresponding dual solution satisfying (19), then*

$$
|N(\sigma(u,p)) - N^h(\sigma(U_h,P_h))| = |\sum_{K \in \mathcal{T}_n} \mathcal{E}_K|,
$$

*where $\mathcal{E}_K = e_D^K + e_M^K$ with*

$$
\begin{aligned}
e_D^K =& \frac{1}{|I|}\int_I \big((\dot{U}_h + U_h \cdot \nabla U_h - f, \varphi - \Phi)_K - (P_h, \nabla \cdot (\varphi - \Phi))_K \\
&+ (\nabla \cdot U_h, \theta - \Theta)_K + (2\nu\epsilon(U_h), \epsilon(\varphi - \Phi))_K\big)\, dt, \\
e_M^K =& \frac{1}{|I|}\int_I SD(\delta, U_h, P_h, \Phi, \Theta)_K\, dt.
\end{aligned}
$$

We may view $e_D^K$ as the error contribution from the discretization on element $K$, and $e_M^K$ as the contribution from the subgrid model on element $K$.

**Remark 6.** *In the computational examples below we use non-Dirichlet boundary conditions, such as slip conditions at lateral boundaries and transparant outflow conditions, which introduce additional boundary terms in the error representation in Theorem 5. Since the dual solutions for these examples are small at such non-Dirichlet boundaries, we neglect the corresponing boundary terms in the computations.*

### *6.3. An a posteriori error estimate*

From the error representation in Theorem 5 there are various possibilities to construct error indicators and stopping criterions in an adaptive algorithm. Using standard interpolation estimates, with $(\Phi, \Theta)$ a finite element interpolant of $(\varphi, \theta)$, we may estimate the contribution $e_D^K$ from discretization as follows (cf. Hoffman, 2002a)

$$
\begin{aligned}
e_D^K \leq \frac{1}{|I|}\int_I \big((&|R_1(U_h,P_h)|_K + |R_2(U_h,P_h)|_K) \cdot (C_h h^2 |D^2\varphi|_K + C_k k |\dot{\varphi}|_K) \\
&+ \|R_4(U_h)\|_K (C_h h^2 \|D^2\theta\|_K + C_k k \|\dot{\theta}\|_K)\big)\, dt,
\end{aligned}
$$

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

where the residuals $R_i$ are defined in (8), $D^2$ denotes second order spatial derivatives, and we write $|w|_K \equiv (\|w_1\|_K, \|w_2\|_K, \|w_3\|_K)$, with $\|w\|_K = (w,w)_K^{1/2}$, and let the dot denote the scalar product in $\mathbb{R}^3$.

The next step involves replacing the exact dual solution $(\varphi, \theta)$ by a computed approximation $(\varphi_h, \theta_h)$ obtained using G$^2$ on (usually) the same mesh as we use for the primal problem. Doing so we are led to the following a posteriori error estimate:

$$|N(\sigma(u,p)) - N^h(\sigma(U_h, P_h))| \approx |\sum_{K \in \mathcal{T}_n} \mathcal{E}_{K,h}| \tag{22}$$

where $\mathcal{E}_{K,h} = e^K_{D,h} + e^K_{M,h}$ with

$$\begin{aligned} e^K_{D,h} &= \frac{1}{|I|}\int_I \big( (|R_1(U_h,P_h)|_K + |R_2(U_h,P_h)|_K) \cdot (C_h h^2 |D^2\varphi_h|_K + C_k k |\dot{\varphi}_h|_K) \\ &\quad + \|R_4(U_h)\|_K \cdot (C_h h^2 \|D^2\theta_h\|_K + C_k k \|\dot{\theta}_h\|_K) \big)\, dt, \\ e^K_{M,h} &= \frac{1}{|I|}\int_I SD(\delta, U_h, P_h, \varphi_h, \theta_h)_K \, dt, \end{aligned}$$

where we have replaced the interpolant $(\Phi, \Theta)$ by $(\varphi_h, \theta_h)$. Again we may view $e^K_{D,h}$ as the error contribution from the discretization on element $K$, and $e^K_{M,h}$ as the contribution from the subgrid model on element $K$.

### *6.4. An adaptive algorithm*

In the computations we use Adaptive DNS/LES cG(1)cG(1) with an algorithm for adaptive mesh refinement in space (with for simplicity the same space mesh for all time steps) based on the a posteriori error estimate (22), of the form: Given an initial coarse computational space mesh $\mathcal{T}^0$, start at $k = 0$, then do

*(1) Compute approximation to the primal problem using $\mathcal{T}^k$.*
*(2) Compute approximation to the dual problem using $\mathcal{T}^k$.*
*(3) If* $|\sum_{K \in \mathcal{T}_k} \mathcal{E}^k_{K,h}| < TOL$ *then STOP, else:*
*(4) Refine a fraction of the elements in $\mathcal{T}^k$ with largest $\mathcal{E}^k_{K,h} \to \mathcal{T}^{k+1}$.*
*(5) Set $k = k+1$, then goto (1).*

## 7. Drag of a square cylinder

We now present results for the benchmark problem of a flow over a square cylinder governed by the Navier-Stokes equations (1) with Reynolds number 22.000, based on the cylinder side $D = 0.1$, as described in Rodi *et al.*, 1997. The computational domain is a channel of size $21D \times 14D \times 4D$ in the $x_1$-direction with the cylinder directed in the $x_3$-direction and centered at $x_1 = 5D$ and $x_2 = 7D$. The inflow velocity is equal to unity, we use no slip boundary conditions on the cylinder, slip boundary conditions on the lateral boundaries, and a transparant outflow boundary condition.

We seek to compute the *mean drag coefficient* $\bar{c}_D$ over a time interval $I = [0, 100D]$ at fully developed flow, defined by

$$\bar{c}_D = \frac{1}{|I|}\int_I c_D, \qquad \bar{c}_D \equiv \frac{2N(\sigma(u,p))}{\bar{U}_h^2 A}, \tag{23}$$

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

where $c_D(t)$ is the drag coefficient at time $t$, we set $\bar{U}_h = 1$ as the inflow velocity, and the area $A = D \times 4D = 4D^2$.

In Rodi *et al.*, 1997, experimental reference values of $\bar{c}_D$ in the interval $[1.9, 2.1]$ are reported, and different groups report computational results in the (wide) interval $[1.6, 2.8]$, using RANS or LES with various subgrid models, wall-functions, and number of mesh points.

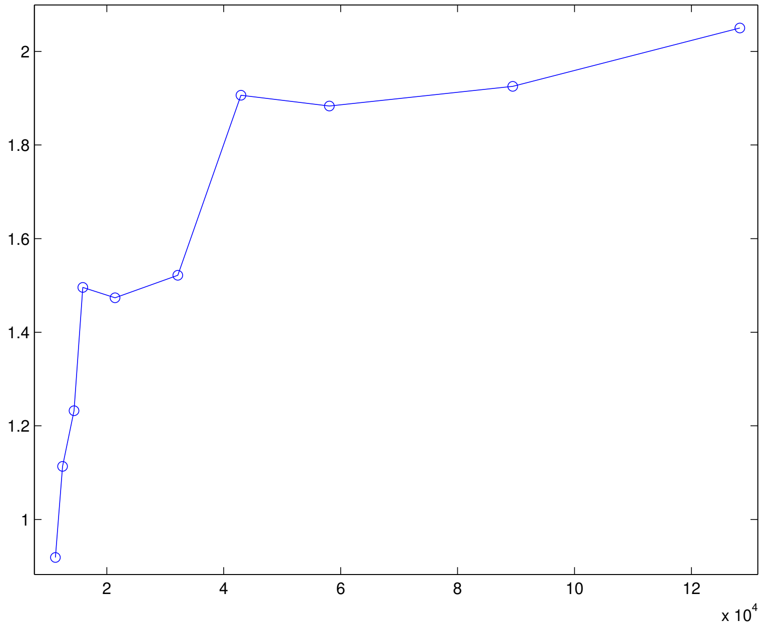


Figure 1. Square cylinder: mean drag coefficient $\bar{c}_D$ as a function of the number of mesh points.

We plot our computational approximation of $\bar{c}_D$ in Figure 1, as a function of the number of mesh points in space. Using about $10^5$ mesh points in space we have $\bar{c}_D \approx 2.0$, and already using about $5 \cdot 10^4$ mesh points we are within the tolerance of the reported experimental reference results in Rodi *et al.*, 1997.

In Figure 2 we plot the a posteriori error estimates $e_{D,h} = \sum_K e^K_{D,h}$ and $e_{M,h} = \sum_K e^K_{M,h}$ as functions of the number of mesh points, indicating an estimated tolerance of $10 - 20\%$ for the results on the finer meshes. $e_{M,h}$ consists of sums in space and time of integrals over the space-time elements, and we may want to use a more conservative estimate of this term by taking the absolute values inside any or both of these sums. In Figure 2 we plot $e_{M,h}$ with the absolute values inside the sum in space, but with the absolute values outside the sum in time. We have also plotted an approximation of the true error assuming 2.0, the mean of the experimental results, to be the true value.

A snapshot of the primal solution is plotted in Figure 3, and in Figure 4 we plot a snapshot of the dual solution. We note that the dual solution, with boundary data on the cylinder, is of moderate size, and in particular is not exploding as pessimistic worst case analytical estimates may suggest, but rather seems to behave as if the net effect of the crucial reaction term (with large oscillating coefficient $\nabla U_h$) is only a moderate growth. We also note that $(\varphi_h, \theta_h)$ is very concentrated in space, thus significantly influencing the adaptive mesh refinement. The resulting computational mesh after 9 adaptive mesh

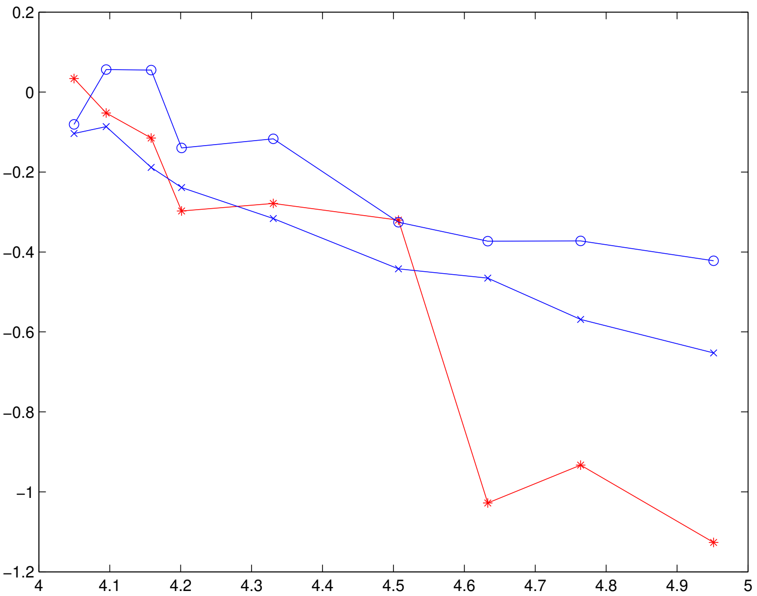


Figure 2. Square cylinder: a posteriori error estimates $e_{D,h}$ ('o') and $e_{M,h}$ ('x'), and the true error ('*') based on $\bar{c}_D = 2.0$, as functions of the number of mesh points in space.

refinements is plotted in Figure 5. Without the dual weights in the a posteriori error estimate the mesh would come out quite differently. We notice in particular that the adaptive method automatically captures the turbulent wake, which seems to be essential for accurately computing drag.

## 8. The drag of a surface mounted cube

We now proceed to present Adaptive DNS/LES cG(1)cG(1) for another basic benchmark problem of CFD of computing the drag of a surface mounted cube at Reynolds number 40.000. As in the previous problem the incoming flow is laminar time-independent with a laminar boundary layer on the front surface of the body, which separates and develops a turbulent time-dependent wake attaching to the rear of the body. The flow is thus again very complex with a combination of laminar and turbulent features including boundary layers and a large turbulent wake, see Figure 6.

The cube side length is $H = 0.1$, and the cube is centrally mounted on the floor of a rectangular channel of length $15H$, height $2H$, and width $7H$, at a distance of $3.5H$ from the inlet. The cube is subject to a Newtonian flow $(u, p)$ governed by the Navier-Stokes equations (1) with kinematic viscosity $\nu = 2.5 \cdot 10^{-6}$ and unit inlet bulk velocity corresponding to a Reynolds number of 40.000 using the dimension of the cube as characteristic dimension. The inlet velocity profile is interpolated from experiments, as given in Krajnović and Davidson, 2002, and we use no slip boundary conditions on the cube and the vertical channel boundaries, slip boundary conditions on the lateral channel boundaries, and a transparent outflow boundary condition. We compute $\bar{c}_D$ over a time interval $[0, 40H]$ at fully developed flow with Adaptive DNS/LES $\mathrm{cG}(1)\mathrm{cG}(1)$, with the space mesh and time steps being constant in time.

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

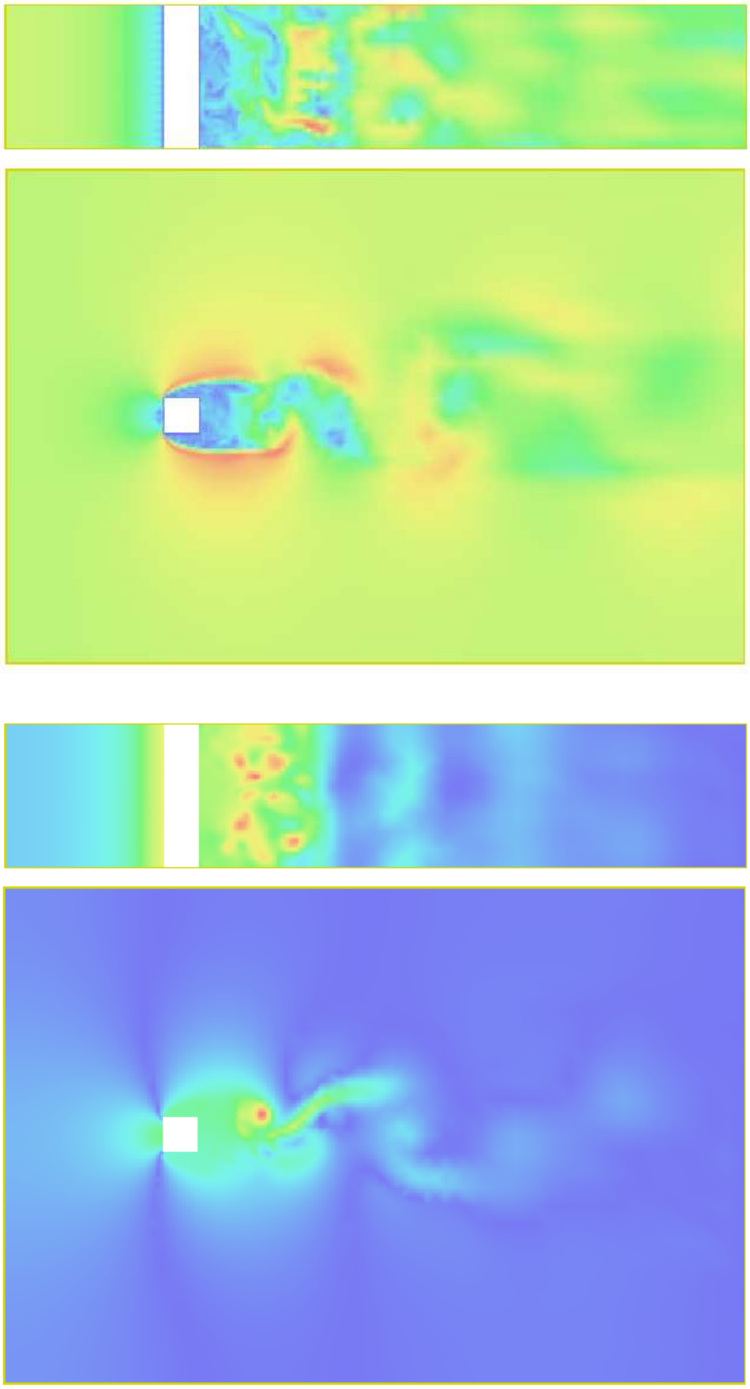

Figure 3. Square cylinder: velocity $|u|$ (upper) and pressure $|p|$ (lower), after 9 adaptive mesh refinements, in the $x_1x_3$-plane at $x_2 = 7D$ (upper) and in the $x_1x_2$-plane at $x_3 = 2D$ (lower).

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

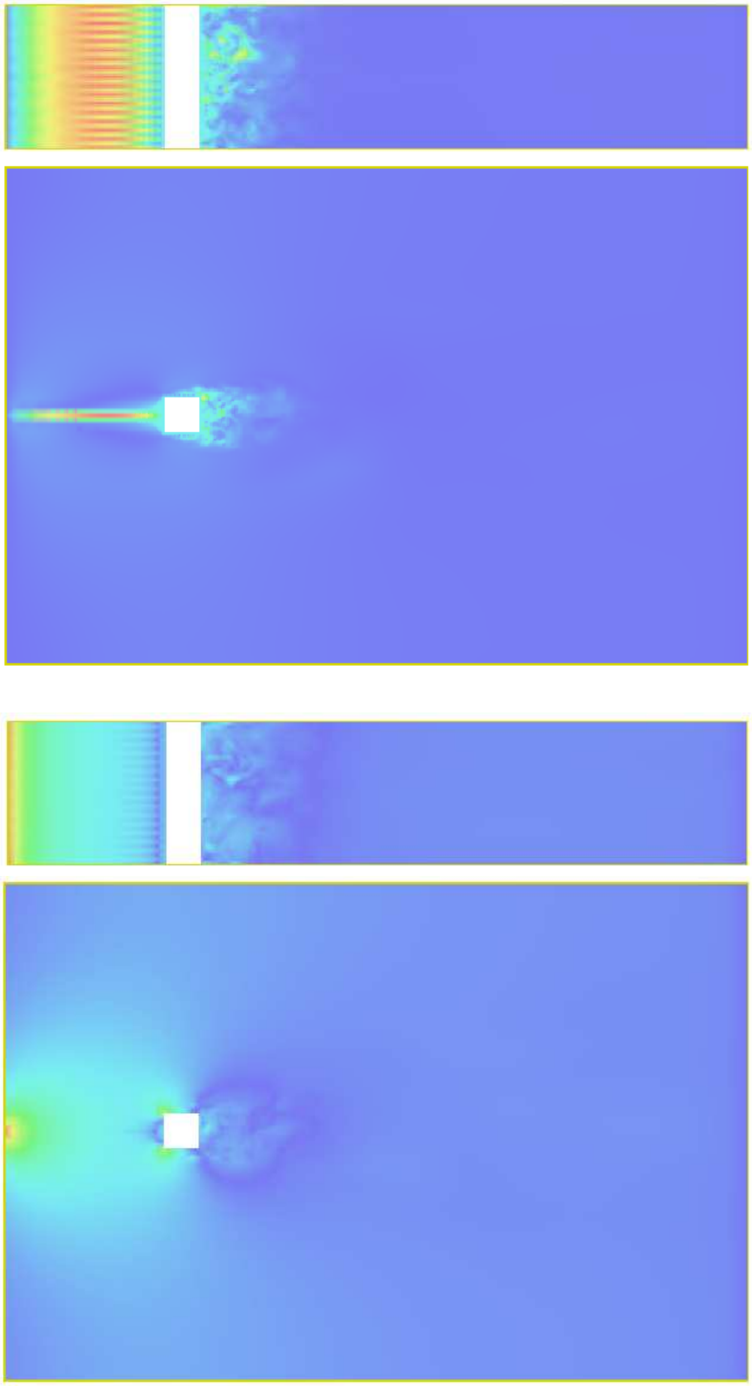

Figure 4. Square cylinder: dual velocity $|\varphi_h|$ (upper) and pressure $|\theta_h|$ (lower), after 9 adaptive mesh refinements in the $x_1x_3$-plane at $x_2 = 7D$ (upper) and in the $x_1x_2$-plane at $x_3 = 2D$ (lower).

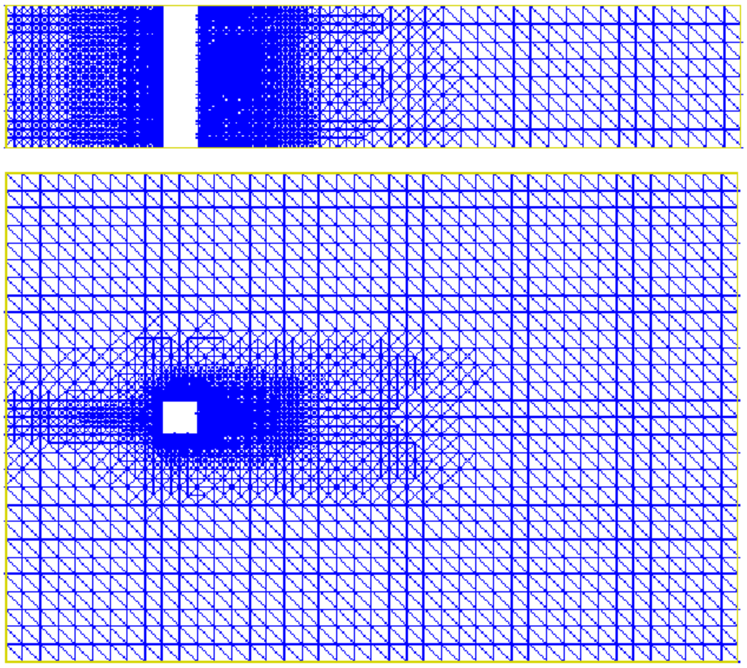

Figure 5. Square cylinder: computational mesh after 9 adaptive mesh refinements, in the $x_1x_3$-plane at $x_2 = 7D$ (upper) and in the $x_1x_2$-plane at $x_3 = 2D$ (lower).

In Figure 7 we show the computed values of $\bar{c}_D$ (for a time interval of length $40H$). The approximations of $\bar{c}_D$ approaches $\approx 1.45$, a value that is well captured already using less than $10^5$ mesh points. We notice that the computed drag increases during the early refinement process and then quickly stabilizes.

We know of no experimental reference values of $\bar{c}_D$, but in Krajnović and Davidson, 2002, $\bar{c}_D$ is approximated computationally. The computational setup is similar to the one in this paper except the numerical method, a different length of the time interval, and that we in this paper use a channel of length $15H$, compared to a channel of length $10H$ in Krajnović and Davidson, 2002. Using different meshes and subgrid models, approximations of $\bar{c}_D$ in the interval $[1.14, 1.24]$ are presented in Krajnović and Davidson, 2002.

In Figure 8 we plot the a posteriori error estimates $e_{D,h}$ and $e_{M,h}$, as well as the true error based on $\bar{c}_D = 1.44$, the computational approximation on the finest mesh. In the evaluation of $e_{M,h}$ we have set the absolute values inside the sums in space and time. We find that once the value for $\bar{c}_D$ has stabilized, the a posteriori error estimates indicate that it may be hard to further increase the precision in $\bar{c}_D$. This may be related to the fact that to further increase the precision in $\bar{c}_D$, a better pointwise approximation of the trajectory of the true $c_D$ is demanded, which may be very expensive. Such an increased precision may not even be desired, since the actual trajectory of $c_D$ may be sensitive even for very small perturbations, and thus it is typically very hard to replicate also experiments with identical $c_D$ trajectories. In Figure 9 we plot the trajectories of $c_D$ for the computations on the 5 finest computational meshes.

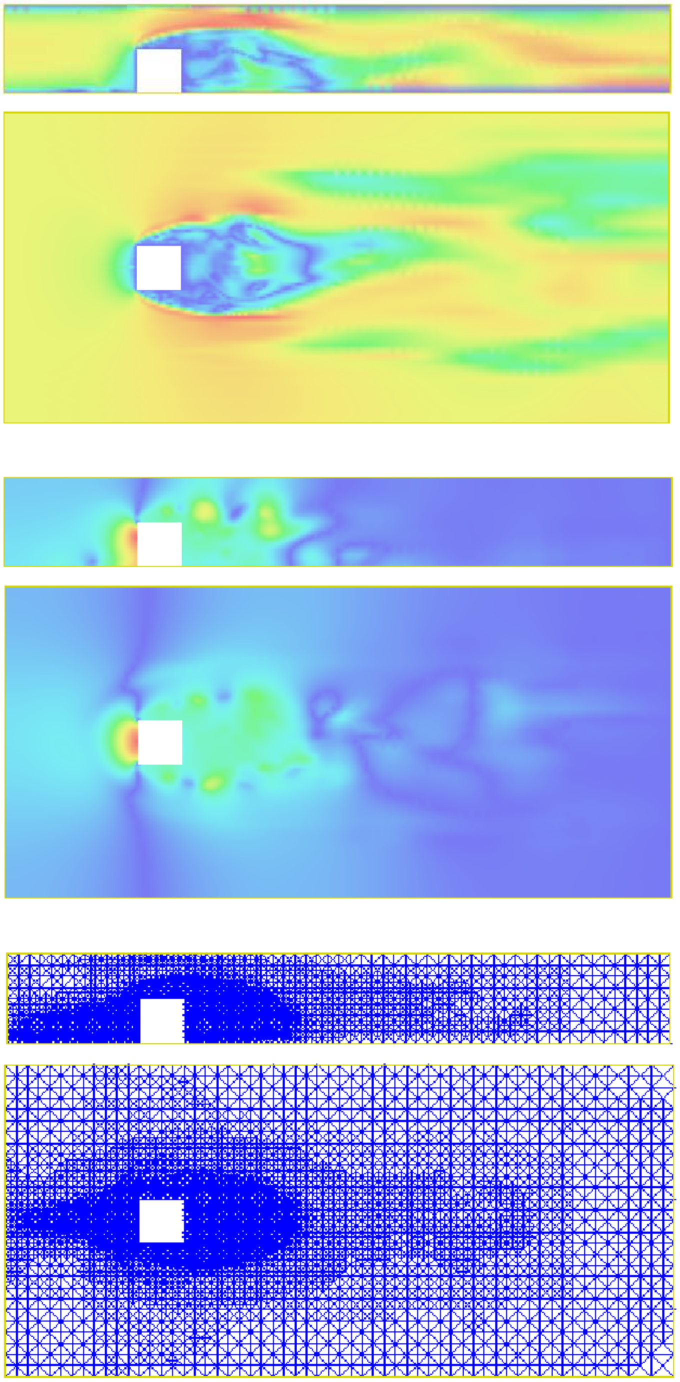

Figure 6. Surface mounted cube: velocity $|u|$ (upper), pressure $|p|$ (middle), and computational mesh (lower), after 13 adaptive mesh refinements, in the $x_1x_2$-plane at $x_3 = 3.5H$ and in the $x_1x_3$-plane at $x_2 = 0.5H$.

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

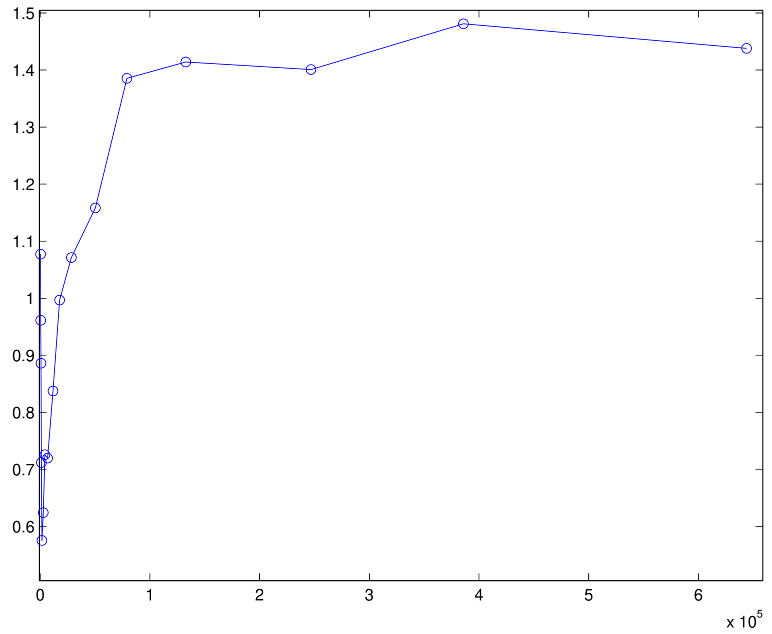


Figure 7. Surface mounted cube: mean drag coefficient $\bar{c}_D$ as a function of the number of mesh points.

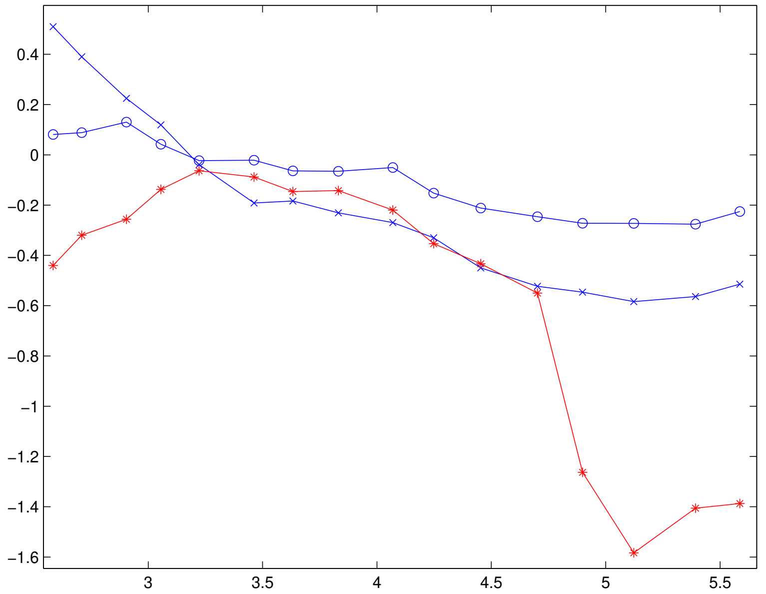


Figure 8. Surface mounted cube: a posteriori error estimates $e_{D,h}$ ('o') and $e_{M,h}$ ('x'), and the true error ('*') based on $\bar{c}_D = 1.44$, as functions of the number of mesh points in space.

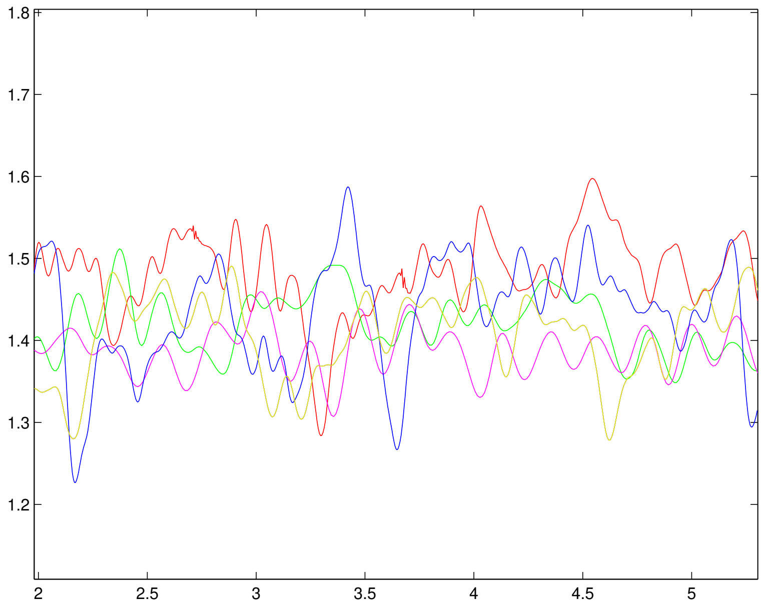


Figure 9. Surface mounted cube: drag coefficient $\bar{c}_D$ as a function time, for the 5 finest adaptively refined meshes.

## 9. The drag vs the total dissipation

For accurate approximation of the drag in the bluff body problems we need to capture the correct global dissipation, which follows from the basic global energy balances for the Navier-Stokes equations and G$^2$, obtained by multiplication by $(u, p)$ and choosing $(v, q) = (U_h, P_h)$ respectively, to get

$$\frac{d}{dt}\|u\|^2 \approx \int_{\Gamma_{in}} pu \cdot n \, ds - \int_{\Gamma_{out}} pu \cdot n \, ds - \nu\|\epsilon(u)\|^2,$$
$$\frac{d}{dt}\|U_h\|^2 \approx \int_{\Gamma_{in}} P_h U_h \cdot n \, ds - \int_{\Gamma_{out}} P_h U_h \cdot n \, ds - (\nu\|\epsilon(U_h)\|^2 + SD(\delta, U_h, P_h, U_h, P_h)),$$

using partial integration, dropping the small boundary terms containing $\nu$, and denoting by $\Gamma_{in}$ and $\Gamma_{out}$ the inflow and outflow boundaries, respectively. Here $\|\cdot\|$ denotes the $L_2(\Omega)$ norm, and $D_\nu(u) = \nu\|\epsilon(u)\|^2$ represents the exact global dissipation (rate), and $D(U_h, P_h) = D_\nu(U_h) + SD(\delta, U_h, P_h, U_h, P_h)$ the corresponding G$^2$ approximate global dissipation (rate). We notice that the difference of the two boundary integrals represents the pressure drop from inflow to outflow, which roughly should correspond to the drag. Since in the bluff body problems we have $\frac{d}{dt}\|u\|^2 \sim \frac{d}{dt}\|U_h\|^2 \sim 0$, we thus have that the pressure drop $\sim$ global dissipation, and thus we expect that the drag $\sim$ global dissipation.

In Figure 11 we show the development under the mesh refinement process of the volume of the turbulent wake, in the case of the surface mounted cube problem. We note that the volume increases as the early refinement proceeds. The initial large values for the volume on the coarsest meshes are related to large numerical dissipation from the stabilization on these under resolved meshes. We may view the refinement as increasing the effective Reynolds number in the computation and thus we may expect the

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

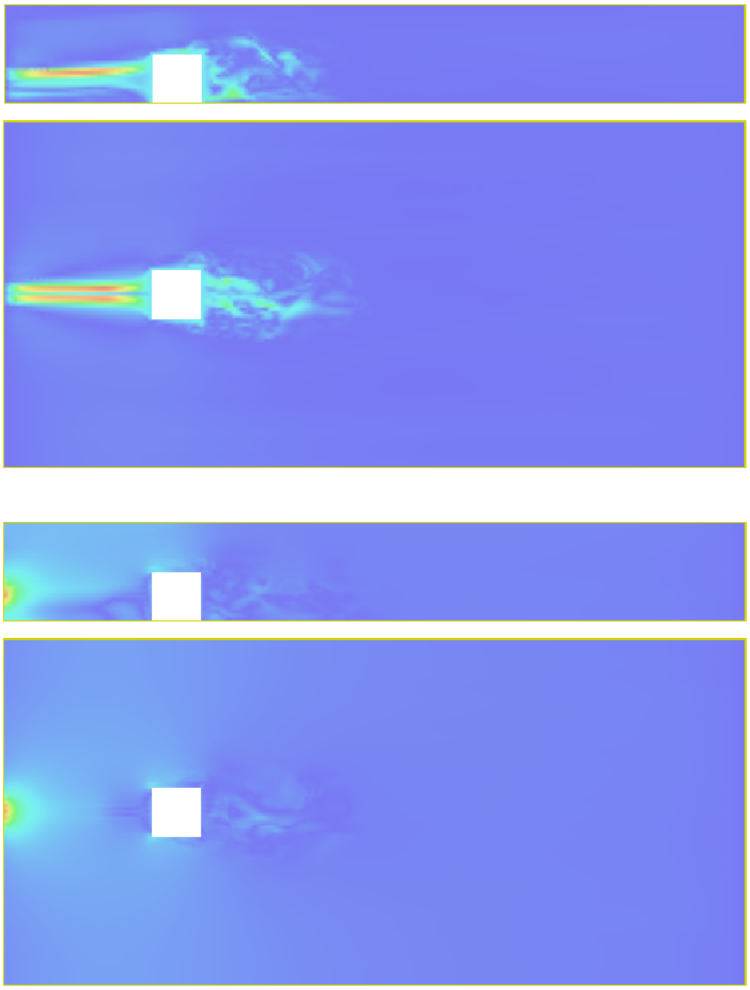

Figure 10. Surface mounted cube: dual velocity $|\varphi|$ (upper) and dual pressure $|\theta|$ (lower), after 13 adaptive mesh refinements, in the $x_1x_2$-plane at $x_3 = 3.5H$ and in the $x_1x_3$-plane at $x_2 = 0.5H$.

expansion of the turbulent wake to parallel an expansion of the wake as the Reynolds number increases.

In Figure 12 we show (the mean value of) the computed intensity of the global dissipation $D(U_h, P_h)$ in the turbulent wake, being the sum of the intensity of $D_\nu(U_h)$, which is small, and the intensity of $SD(\delta, U_h, P_h, U_h, P_h)$ corresponding to the stabilization. We hope (the mean value of) $D(U_h, P_h)$ to be an approximation of (the mean value of) the dissipation rate $D_\nu(u)$, which we expect to be most significant in the turbulent wake. We observe that (the mean value of) the intensity of $D(U_h, P_h)$ is very nearly constant during the refinement process, which we may take as evidence that indeed (the mean of the) the intensity of the mesh dissipation $D(U_h, P_h)$ may approximate (the mean value of) the true fine scale dissipation intensity $D_\nu(u)$.

We conclude that to get the correct drag, we have to compute the correct global dissipation, including

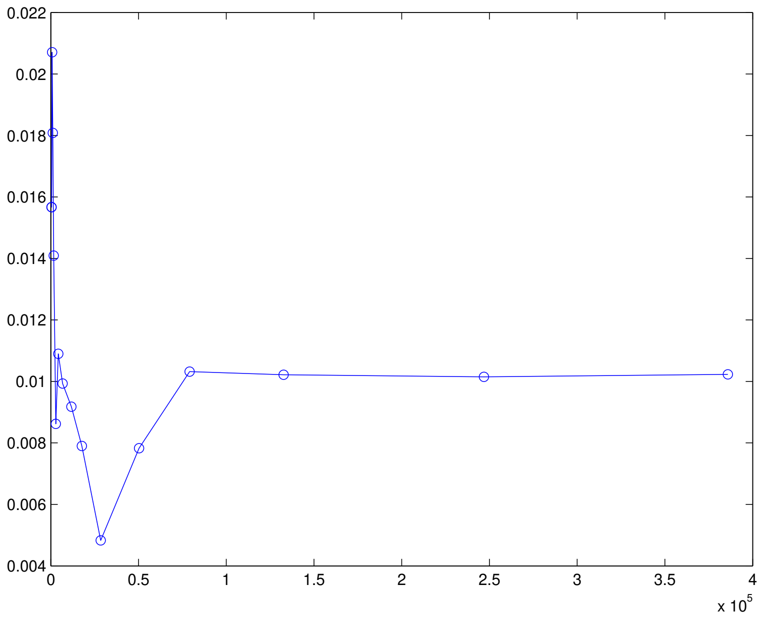


Figure 11. Surface mounted cube; volume of turbulent wake, defined as the part of the domain with mean dissipation intensity $D(U_h, P_h) > 0.1$, versus number of mesh points.

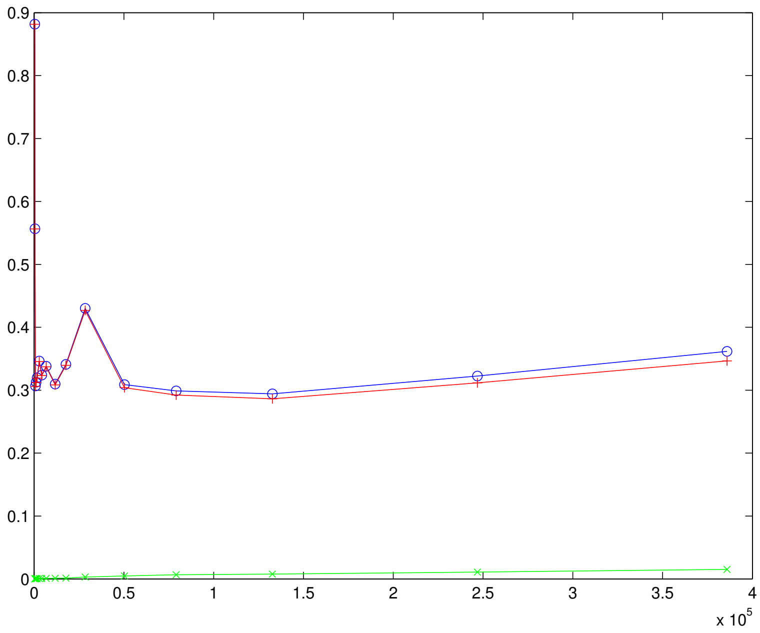


Figure 12. Surface mounted cube: dissipation intensity $D(U_h, P_h)$ ('o') as the sum of $D_\nu(U_h)$ ('x') and the intensity of the stabilization term $SD(\delta, U_h, P_h, U_h, P_h)$ ('+') in the turbulent wake, vs number of mesh points.

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

a correct volume of the turbulent wake and the correct turbulent dissipation intensity. With Adaptive DNS/LES we seem to be able to do so using effectively LES in the turbulent wake and some form of DNS to accurately trace the boundary of the turbulent wake. In particular, we do not have to resolve the large turbulent volume by DNS, and we seem to be able to compute the drag correctly with in fact less than $10^5$ mesh points in space.

If we combine the increase in volume of the turbulent wake (in the early refinement) with the near constancy of the intensity of the dissipation there, we find that the total dissipation rate (coming mostly from the turbulent wake) increases under the early refinement process, which we may view to correspond to the increase of the drag under the early refinement.

## 10. Reliability and efficiency of the adaptive method

We now forcus, in the context of the above drag problems, on two key points relating to the *reliability* and *efficiency* of the adaptive method based on the a posteriori error estimate (22), which directly couples to whether this estimate indeed gives a reasonably sharp bound of the true error, or not. The two key points are

1. Replacement of $u$ by a computed velocity $U_h$ in the dual problem.
2. Replacement of the dual solution $(\varphi, \theta)$ by a computed dual solution $(\varphi_h, \theta_h)$.

We may view both these points to relate to a *stability* of the dual solution under perturbations of 1. the convection coefficient and 2. numerical computation. To test such stability we compute dual solutions on the different meshes and we check the variation of certain key aspects of the dual solution as measured in a couple of different norms as functions of the number of mesh points in space.

We first focus on the discretization error term $e_{D,h}$. We may estimate this term using Cauchy's inequality in space and time as follows (taking only space discretization coupled to $\varphi$ into account and neglecting the small $\nu$-term):

$$e_{D,h} \leq C_h \|hR_1(U_h, P_h)\| \|hD^2\varphi_h\|$$

where $C_h \approx 0.1$ and by the least squares stabilization in G$^2$ we have that $\|\sqrt{h}R_1(U_h, P_h)\|$ is bounded (recalling that $\delta_1 \sim h$). Here $\|\cdot\| = \|\cdot\|_{L_2(I;L_2(\Omega))}$ denotes a $L_2$ norm in space-time. Thus, very roughly we would expect to have

$$e_{D,h} \leq C_h \sqrt{h} \|hD^2\varphi_h\|.$$

In Figure 13 we display the variation of $\|hD^2\varphi_h\|$ as a function of the number mesh points in space for the square cylinder, and we notice that it is nearly constant. We also show a similar behaviour for the surface-mounted cube.

The fact that $\|hD^2\varphi_h\|$ may stay nearly bounded during the refinement process may be seen to couple to the presence of a dissipative term of the form $-h\Delta\varphi_h$ in the dual problem, effectively coupling to the stabilization used in the computation of the dual solution with a numerical viscosity of size $h(x)$. By using "elliptic regularity" we may thus expect to be able to estimate $\|hD^2\varphi_h\|$ in terms of the lower order terms in the dual problem, and thus it may be possible that indeed $\|hD^2\varphi_h\|$ stays nearly bounded during the refinement process. Such a fact is similar to a property of shock problems for compressible flow explored in Johnson, 1997.

Next, the error contribution from subgrid modeling $e_{M,h} = SD(\delta, U_h, P_h, \varphi_h, \theta_h)$ may be estimated roughly as follows, using the basic energy estimate to bound $SD(\delta, U_h, P_h, U_h, P_h)$, Cauchys

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

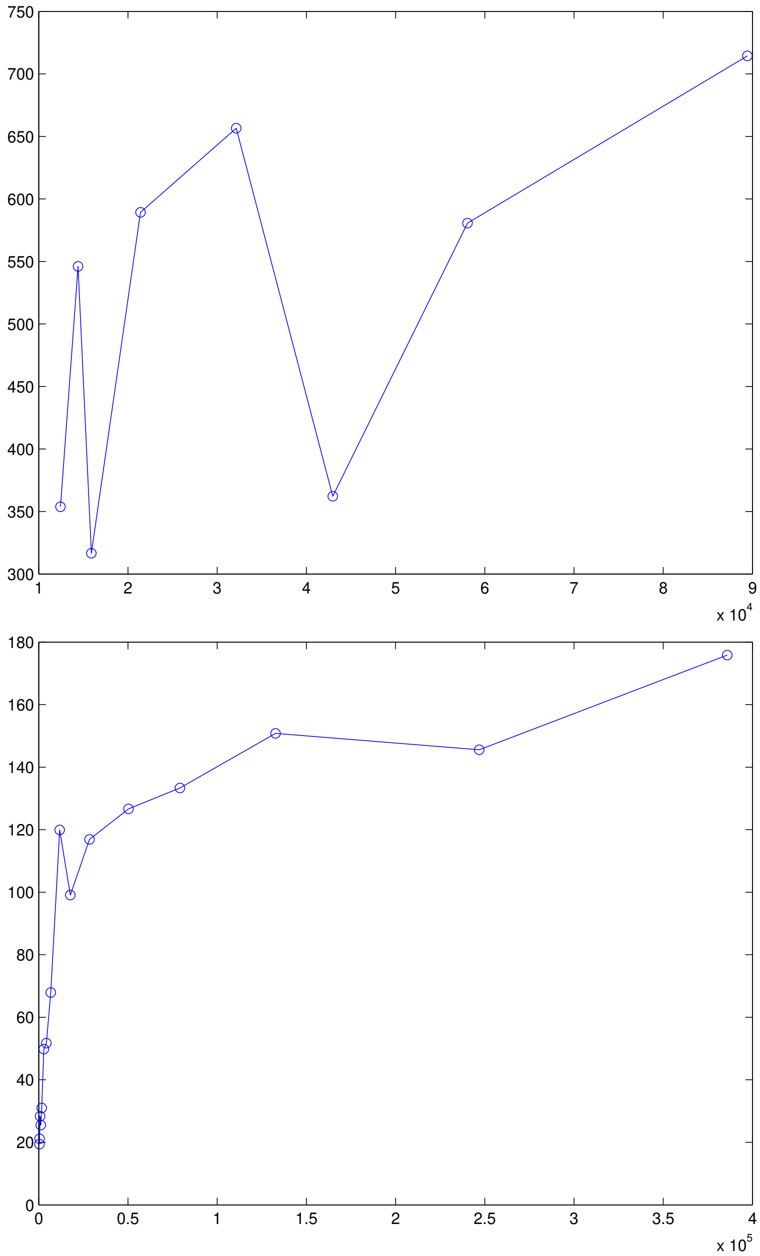


Figure 13. Stability factor $\|hD^2\varphi_h\|_{L_2(I;L_2(\Omega))}$ for the square cylinder (upper) and the surface mounted cube (lower), versus number of mesh points in space.

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

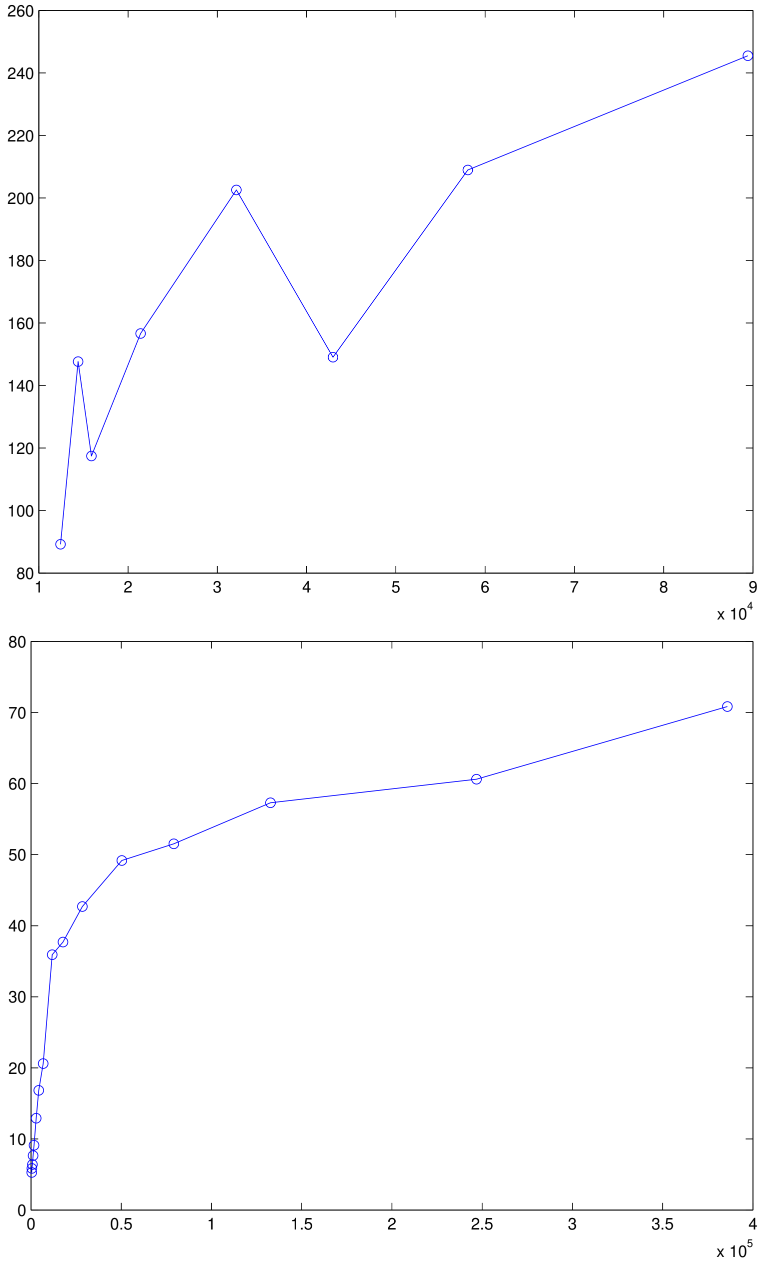


Figure 14. Stability factor $\|\nabla\varphi_h\|_{L_2(I;L_2(\Omega))}$ for the square cylinder (upper) and the surface mounted cube (lower), versus number of mesh points in space.

inequality, and recalling that $\delta_1 \sim h$, to get

$$e_{M,h} \leq \sqrt{h}\|\nabla\varphi_h\|$$

where we only accounted for the $\varphi_h$ term. We notice in Figure 14 that $\|\nabla\varphi_h\|$ is of moderate size during the refinement suggesting that indeed $e_{M,h}$ may get below a moderate tolerance under refinement without reaching a DNS.

Altogether, we conclude that the crucial computed dual weights show a stability under mesh refinement which indicates that the a posteriori error estimate (22) for the discretization may indeed be reliable and also reasonably efficient.

Concerning the crucial step of replacing $u$ by $U_h$ in the dual problem, which may correspond to locally a large perturbation since $U_h$ cannot be expected to pointwise approximate $u$, we have in particular given evidence that the net effect on the dual weights may be small.

## 11. Averaged Navier-Stokes equations and Reynolds stresses

Instead of comparing, as we have done here, the computed solution $(U_h, P_h)$ directly to the exact solution $(u, p)$, one may attempt to compare with a local mean value $(u^h, p^h)$ which in principle could be pointwise approximated by $(U_h, P_h)$. The dual convection velocity would then be $u^h$, and the replacement by $U_h$ would then correspond only to a small perturbation (cf. Hoffman, 2002a; Hoffman, 2002b; Hoffman and Johnson, 2003). However, comparing with $(u^h, p^h)$ necessarily brings in a Reynolds stress term, which has to be modeled or at least be estimated. Since anyway $(u^h, p^h)$ will be replaced by $(U_h, P_h)$ in the dual problem, the computed dual solution would effectively be the same. Estimating as suggested the Reynolds stress term with the stabilization term would then in fact bring us back to the setting above. We may thus, alternatively, view our computational method as effectively using a Reynolds stress subgrid model for the averaged Navier-Stokes equations and bounding the (small) contribution to the output error from the true Reynolds stresses by the contribution from the subgrid model.

Trying instead to model the Reynolds term would lead us into the major open problem of CFD. Because of this difficulty, we have preferred to live without the Reynolds stress term, at the expense of having to deal with the possibly large perturbation from $u$ to $U_h$ in the dual problem, an issue which we address computationally as indicated. This issue which reflects a certain stability of the dual weights, may be less difficult to deal with than the open problem of accurately modeling the Reynolds stresses, or showing that they can be bounded by the subgrid model.

## 12. The subgrid model from stabilization

We have noted that the subgrid model in Adaptive DNS/LES corresponds to the $SD(\delta, U_h, P_h, v, q)$-term in $G^2$. By here choosing $(v, q) = (U_h, P_h)$, we get the global dissipation $SD(\delta, U_h, P_h, U_h, P_h)$ which is large (of unit size) for the turbulent bluff body problems. On the other hand choosing $(v, q) = (\varphi_h, \theta_h)$, we get the subgrid modeling term $e_{M,h} = SD(\delta, U_h, P_h, \varphi_h, \theta_h)$ in the a posteriori error estimate, which we compute and find to be small (of size $= 0.1$), a fact that we may describe as a "miracle". We conclude that the $SD$-term thus in one (strong) sense is large (giving a large total dissipation) and in another (weak) sense is small (giving a small contribution in the a posteriori error estimate). The key to the observed fact that the $SD$-contribution in the weak sense is small, is that

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

the dual solution in fact shows some regularity, which we discover by computing the dual solution. Intuitively, the observed regularity of the dual solution should relate to the fact that although the reaction coeffcient in the dual problem is large, it is fluctuating and it is conceivable that its net effect is small, if the data for the dual problem is smooth corresponding to mean value outputs. Thus we may intuitively understand that a miracle may happen, but it is only computation of the dual solution that shows that the miracle actually happens ("in the Best of (Leibnizian) Worlds").

A pertinent question in this context is the choice of the parameter $\delta$ in the $SD$-term. The rule is to choose $\delta(x) \sim h(x)$, but how do we know that this is correct? Well, it is in fact possible to choose $\delta$ to be larger, at the expense of possibly having to refine more to decrease in particular $e_{M,h}$. However, choosing $\delta$ smaller eventually makes the numerics explode and $e_{D,h}$ becomes large. We are thus lead to adaptively choosing $\delta$ so that both $e_{D,h}$ and $e_{M,h}$ are small.

## 13. Computability in transition to turbulence

We now give some more aspects of computability of fluid flow. We have seen that the computational error of an output quantity may be expressed as space-time integrals of residuals times (derivatives of) the solution to an associated linearized dual problem. The residuals measure how well the computed solution satisfies the Navier-Stokes equations, and the solution of the dual problem determines how the residual influences the particular output considered. The size of (the derivatives of) the dual solution (the dual weights) directly couples to computability: the larger these quantities are the higher is the computational cost.

The dependence of the stability factors on the length $T$ of the simulation of course couples to computability; if the stablity factors grow quickly in time then only short time simulation is possible, while if the stability factors grow slowly then long-time simulation is feasible. In our related Encyclopedia presentation Eriksson *et al.*, 2003, on Parabolic problems we propose to use the time-dependence of stability factors as a means of classification: In particular, we use the term *parabolic* to identify a problem with the stability factors being bounded for all $T$ (up to possibly a slow logarithmic growth). Roughly speaking this connects to *diffusion-dominated convection-diffusion-reaction* problems, while in *convection-dominated* problems we may meet a linear (or faster) growth. For highly demanding problems, such as the computation of a point value in a turbulent flow, the stability factors may locally grow exponentially. See Eriksson *et al.*, 2003, for a study of the Lorenz system as a simple model for the Navier-Stokes equations.

Of particular interest is the growth of stability factors for mean-value quantities in turbulent flow. A fundamental observation from these studies is that mean-values in turbulent flow appear to be computable with desk-top computational power, thus indicating very good prospects for CFD.

The linearized dual Navier-Stokes equations are closely related to the linearized Navier-Stokes equations, where the linearized dual Navier-Stokes equations describe the propagation of errors coupling to the question of computability, and the linearized Navier-Stokes equations describe the propagation of physical perturbations coupling to predictability and hydrodynamic stability.

### *13.1. Computability of space-time averages*

In Hoffman and Johnson, 2002b, a computational study of transition to turbulence in shear flow is conducted for plane Couette flow at $Re = \nu^{-1} = 10.000$, and in Hoffman, 2002a, we present solutions to the associated time dependent linearized dual Navier-Stokes equations in 3d for various

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

data corresponding to estimates of the error in different space-time averages of the velocity. In the computations cG(1)cG(1) is used on the unit cube with a regular tetrahedral mesh with $65 \times 65 \times 65$ nodes. Periodic boundary conditions were used in the streamwise $x_1$-direction and in the spanwise $x_3$-direction, and on top and bottom the streamwise velocity is $\pm 1$.

In the computations of the dual problem we use cG(1)cG(1) on a regular tetrahedral mesh with $33 \times 33 \times 33$ nodes. We linearize the dual problem at $U_h$, projected onto this mesh. We sample $U_h$ at 41 points in time over a time interval of length 10, and we use linear interpolation in time for intermediate values.

We investigate the dual solution corresponding to the estimation of the error in various space-time averages of the solution. The data for this dual problem is a source term in the dual momentum equation:

$$\psi = (\chi_{\omega\times[30-d(\omega),30]}/|\chi_{\omega\times[30-d(\omega),30]}|, 0, 0) \tag{24}$$

Here $\chi_D$ is the characteristic function for $D \subset \Omega \times I$, and $|D|$ denotes the space-time volume of $D$. That is, we are interested in the error in an average of $u_1$ over the space-time domain $\omega\times[30-d(\omega), 30]$, with $\omega$ being a spatial cube with side length $d(\omega)$ centered at $(0.5, 0.5, 0.5)$, corresponding to a space-time cube with side length $d(\omega)$ centered at $(x, t) = (0.5, 0.5, 0.5, 30 - d(\omega)/2)$.

The dual problems are solved backwards in time, and we find that although data from the spatial cube $\omega$ is spread throughout the computational domain by the convective, diffusive, and reactive mechanisms of the dual equation (19), the dual solution does not grow exponentially as predicted by worst case analytical estimates.

In Figure 15 we plot the $L_1$-norms of the dual solutions for $d(\omega) = 0.5, 0.25, 0.125$. In the initial phase (for backward time) the dual solutions grow through the action of the source term $\psi$ over the time interval $[30 - d(\omega), 30]$, and this initial growth is larger for smaller $d(\omega)$.

In the next phase, for $t < 30 - d(\omega)$, when the source term $\psi$ is zero, there is a growth in the dual solution due to the reaction term $\nabla U_h \cdot \varphi$, which is connected to the irregularity of the computed solution $U_h$. When the dual solution is spread over a larger part of the domain, by convective and diffusive mechanisms, the net growth is weakened by cancelations, which is also the reason why the growth is weaker for larger $d(\omega)$.

The results in this example support the intuitive idea that larger space-time averages are less computationally demanding than small, which implies that we may, for example, be able to compute an approximation of a time average of a certain quantity with a small error to an acceptable computational cost, even though it may be computationally very expensive to approximate this quantity at a specific time.

For the error in space-time averages of the solution over $\omega \times [20, 30]$, the $L_1$-norm of the dual solutions for various $d(\omega)$ are plotted in Figure 15, where now the source term $\psi$ is active over the whole time interval in the solution of the dual problem. We find that in this case the dual solution is smaller than in the previous examples with smaller time averages, as expected.

In Figure 16 we plot the $L_1$-norm of the dual solution linearized at the laminar Couette flow $u = (2(y - 0.5), 0, 0)$, with $Re = \nu^{-1} = 100$, where we have no growth from the reaction term, since diffusive mechanisms dominate. Instead the dual solution is quickly damped followed by a slow further decrease caused by diffusive mechanisms.

Here we also plot the case of the dual solution linearized at the same laminar flow, but now for $Re = \nu^{-1} = 10000$, which corresponds to a highly unstable laminar flow. In this case we get an initial growth of the dual solution due to the reaction term in the dual problem, since the diffusive mechanisms are weaker, after which we get a similar scenario as in the case of $Re = \nu^{-1} = 100$. That is, in this

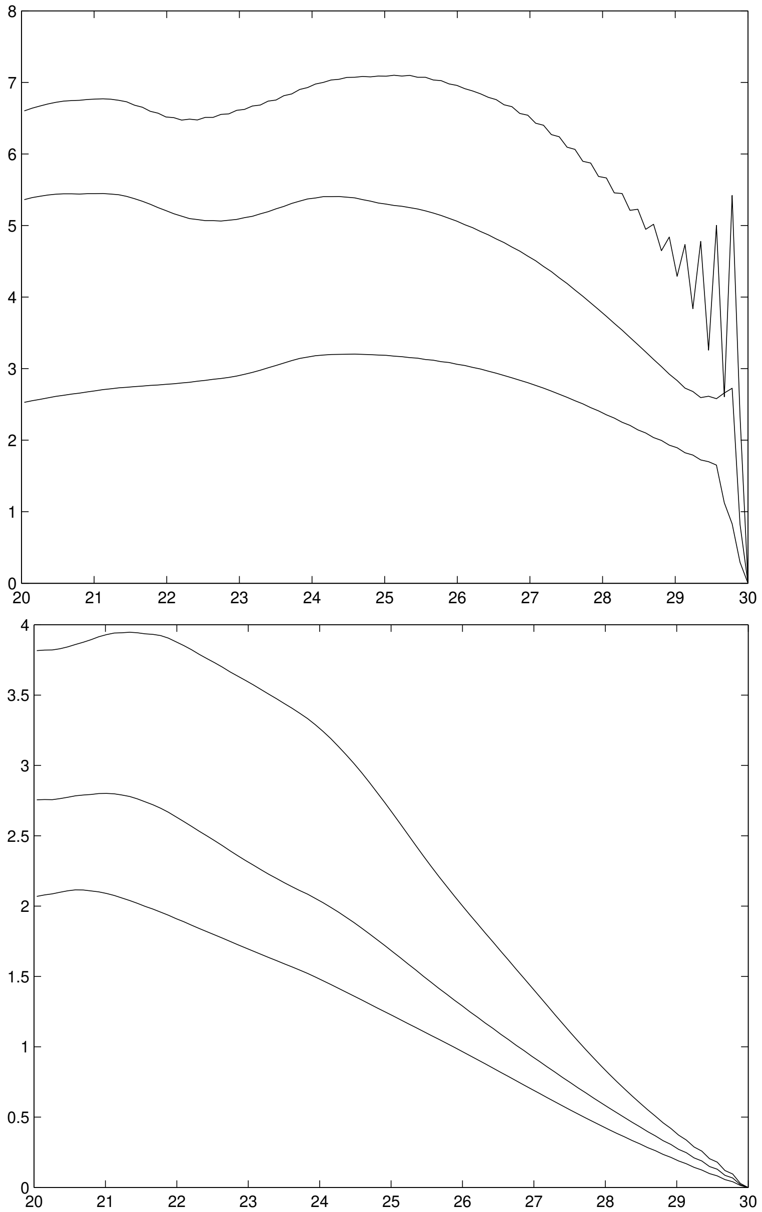


Figure 15. $\|\varphi\|_1$ for $d(\omega) = 0.5, 0.25, 0.125$ linearized at a turbulent flow (upper), and a time average over $\omega \times [20, 30]$ for a turbulent flow (lower), as functions of time.

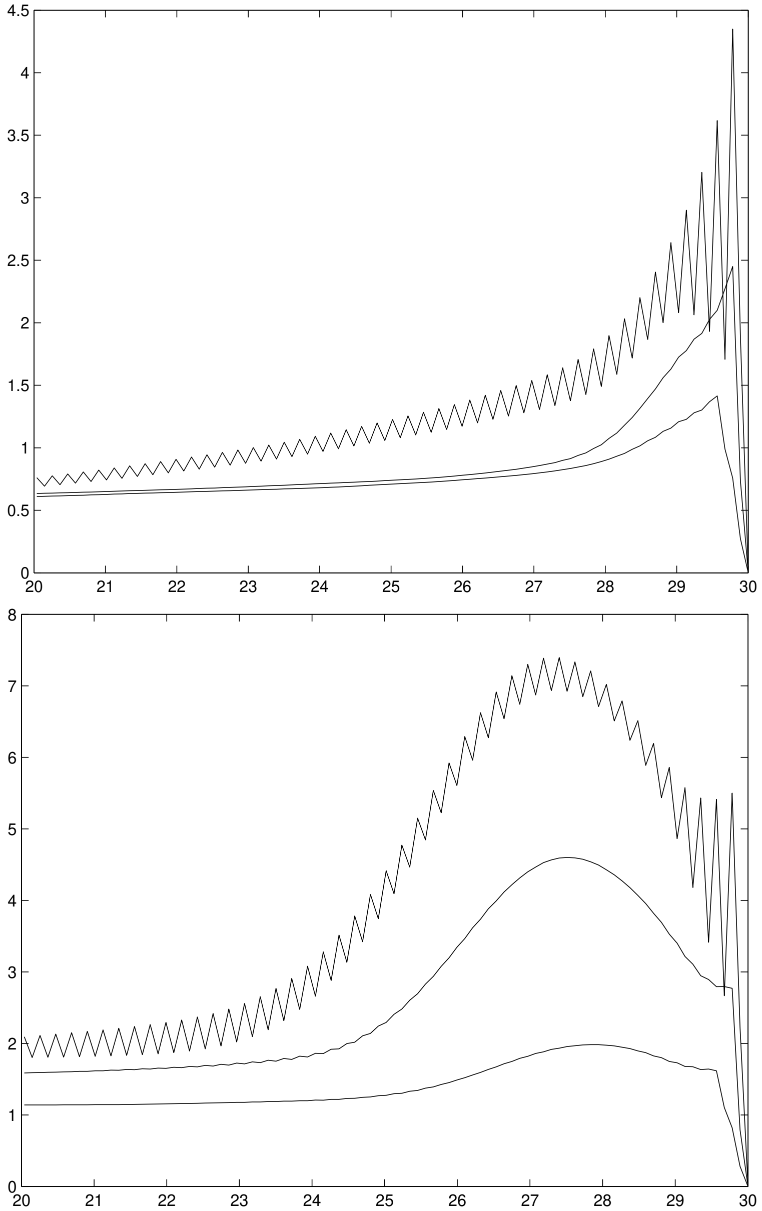


Figure 16. $\|\varphi\|_1$ for $d(\omega) = 0.5, 0.25, 0.125$ linearized at a laminar flow with $Re = \nu^{-1} = 100$ (upper), and a laminar flow with $Re = \nu^{-1}10000$ (lower), as functions of time.

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

example *it is more computationally demanding to compute a numerical approximation of the unstable laminar flow with a larger Reynolds number, even though the exact solution is the same in both cases.*

**Remark 7.** *The computations of the linearized dual problems in this section were all performed with various characteristic functions as data, without the scaling corresponding to the space-time volume of the domain over which the data acted. The solutions were then postprocessed to obtain the proper scaling. This procedure was unfortunate since the relative numerical errors were amplified in the computations for small space-time averages, leading to the wiggles in the graphs in Figure 15 and Figure 16. We accept the computations here since we are mainly interested in the qualitative properties of the dual solutions.*

## 14. Applications to stationary benchmark problems in 3d

We conclude with some results for a collection of benchmark problems for laminar flow around a cylinder in 2d and 3d presented in Schäfer and Turek, 1996, with contributions from 17 research groups. In Hoffman, 2003, we consider the case of 3d stationary flow around a cylinder with square cross-section $D = 0.1$, centered at $(0.5, 0.2, 0.205)$ aligned in the $x_3$-direction, in a channel of dimensions $2.5 \times H \times H$, with $H = 0.41$. We have no slip boundary conditions on the cylinder and the channel walls. At the outflow boundary we use a transparant outflow condition, and the inflow condition is given by $u(0, x_2, x_3) = (16U_m x_2(H - x_2)x_3(H - x_3)/H^4, 0, 0)$. The kinematic viscosity is $\nu = 10^{-3}$ and $U_m = 0.45$, which gives a Reynolds number $Re = \bar{U}D/\nu = 20$, with $\bar{U} = 4U_h(0, H/2, H/2)/9$.

The benchmark problems concern the computation of drag and lift coefficients, and the computation of a pressure difference upstream and downstream of the cylinder. We compute approximations of these functionals using an adaptive algorithm based on error indicators of the form

$$\mathcal{E}_K = C_K h_K^2 \sum_{i=1}^{3} |R_i(U_h, P_h)|_K \cdot \omega_{K,i}, \tag{25}$$

where $R_i(U_h, P_h)$ are residuals corresponding to the stationary Navier-Stokes equations, and $\omega_{K,i}$ are dual weights corresponding to the particular output of interest.

Starting from the coarse initial mesh $\mathcal{T}_0^h$ in Figure 17 with 1414 nodes and 5904 elements we refine approximately 50% of the elements with largest element indicators $\mathcal{E}_K$ in each step of the adaptive algorithm. In the computations below we use a pseudo time step approach based on cG(1)dG(0), solving on the same mesh for the dual and the primal problems in each step of the adaptive algorithm.

Figure 17. Cross-sections of the initial mesh in the $x_1x_2$-plane at $x_3 = 0.205$ (left) and in the $x_1x_3$-plane at $x_2 = 0.2$ (right).

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

### 14.1. *Residual based refinement*

To evaluate the performance of the duality based adaptive mesh refinement, we compare with a similar adaptive algorithm with a refinement criterion

$$\mathcal{E}_K = \sum_{i=1}^{3} \|R_i(U_h, P_h)\|_K, \tag{26}$$

based only on the size of the residuals, coupling to energy type estimates, see e.g. Ainsworth and Oden, 1997. In Figure 18 we show the mesh after 4 refinements, with a mesh refinement criterion based only on the size of $R_1(U_h, P_h)$ and $R_2(U_h, P_h)$, shown in Figure 19.

**Remark 8.** *In the computations we have assumed the residual $R_3(U_h, P_h)$, corresponding to jumps over internal element boundaries of the $\nu$-dissipative term, to be neglible compared to $R_1(U_h, P_h)$ and $R_2(U_h, P_h)$ due to the multiplication by $\nu = 10^{-3}$.*

Figure 18. Cross-sections of the mesh in the $x_1x_2$-plane at $x_3 = 0.205$ (left) and in the $x_1x_3$-plane at $x_2 = 0.2$ (right), after 4 adaptive mesh refinements based on the residuals.

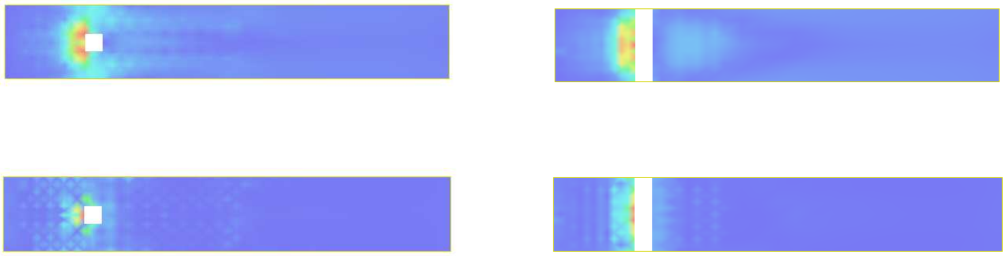

Figure 19. Magnitudes of $R_1(U_h, P_h)$ (upper) and $R_2(U_h, P_h)$ (lower) after 4 adaptive mesh refinements (based on these residuals), in the $x_1x_2$-plane at $x_3 = 0.205$ (left) and the $x_1x_3$-plane at $x_2 = 0.2$ (right).

### 14.2. *Computation of the drag force*

The computational goal is to approximate the drag coefficient $c_D$, and the values of $c_D$ obtained by the different participants for this problem are in the (quite wide) interval $[6.08, 8.09]$, and the interval $[7.5, 7.7]$ is given by the authors to be most likely to contain the exact value. Based on the results in Schäfer and Turek, 1996, we choose $c_D = 7.6$ as our reference value.

Starting from the coarse initial mesh in Figure 17 we use the error indicator (25) to adaptively compute approximations to $c_D$. In Figure 20 we show the adaptively refined meshes after 4 and 6 refinements respectively, and we plot the corresponding dual solution after 4 adaptive refinements in Figure 21.

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

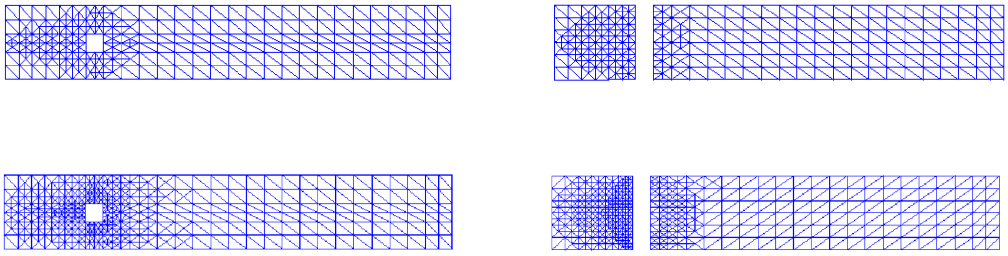

Figure 20. Cross-sections of the meshes in the $x_1x_2$-plane at $x_3 = 0.205$ (left) and in the $x_1x_3$-plane at $x_2 = 0.2$ (right), for computing the drag coefficient $c_D$, after 4 (upper) and 6 (lower) adaptive mesh refinements respectively.

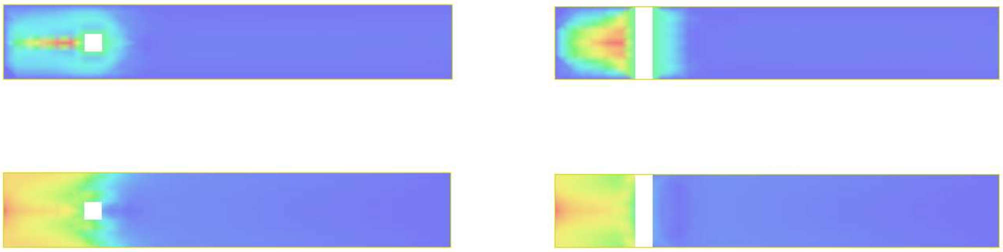

Figure 21. Dual velocity (upper) and pressure (lower) for computing the drag coefficient $c_D$ after 4 adaptive mesh refinements in the $x_1x_2$-plane at $x_3 = 0.205$ (left) and the $x_1x_3$-plane at $x_2 = 0.2$ (right).

To estimate the computational cost of computing approximations $\hat{c}_D$ to the drag coefficient $c_D$, we compute stability factors related to the relative error in $\hat{c}_D$, defined by

$$\frac{|c_D - \hat{c}_D|}{c_D} \le S_{13}^2(\|h^2 R_1(U_h, P_h)\| + \|h^2 R_3(U_h, P_h)\|) + S_2^2 \|h^2 R_2(U_h, P_h)\|, \tag{27}$$

with $S_{13}^s = \|D^s \varphi\|$ and $S_2^s = \|D^s \theta\|$. The stability factors in Table I are computed based on the dual solution on a mesh after 6 refinements and give a rough estimate of the computational cost needed to reduce the relative error below a certain given tolerance, that is how small we need to make the mesh size and the residuals.

| $s$ | $S_{13}^s$ | $S_2^s$ |
|---|---|---|
| 0 | $2.3 \cdot 10^1$ | $1.7 \cdot 10^1$ |
| 1 | $8.6 \cdot 10^2$ | $8.3 \cdot 10^1$ |
| 2 | $3.6 \cdot 10^4$ | $3.7 \cdot 10^3$ |

Table I. Stability factors for the computation of the drag coefficient $c_D$ with respect to the relative error.

To evaluate the performance of the adaptive algorithm we compare with an adaptive algorithm using the refinement criterion (26), based only on the size of the residuals. In Figure 22 we compare the convergence rates of the two approaches with respect to the reference value $c_D = 7.6$. It is obvious that the refinement criterion (25), based on both the residual and the solution to the dual problem, does a better job than the refinement criterion (26), solely based on the residuals without any information from the dual problem relating the residual to the error in the drag coefficient $c_D$.

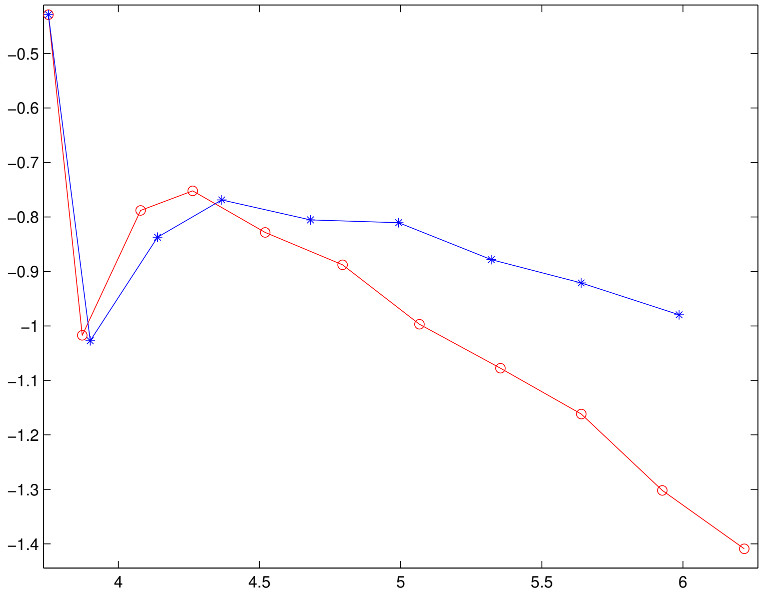


Figure 22. Convergence rates for the computation of the drag coefficient $c_D$, for duality based refinement ('o') and residual based refinement ('*'), as a log-log plot of number of unknowns versus relative errors.

The use of the duality based a posteriori error estimates as a refinement criterion was thus shown to be successful for this example. We now evaluate the performance of the a posteriori error estimates used as a stopping criterion for the adaptive algorithm. For this purpose we use the notion of an effectivity index $I_{eff}$ defined as

$$I_{eff} = estimated\ error/true\ error. \tag{28}$$

In Table II we present $I_{eff}$ as a function of the number of unknowns. The results indicate that the a posteriori error estimation of the true error in this case is quite sharp, after a few initial refinements the error estimate is off by less than a factor 2, and may thus be useful as a stopping criterion.

### *14.3. Computation of lift force*

The computational goal is now to approximate the lift coefficient $c_L$. In Schäfer and Turek, 1996, an interval $[0.06, 0.08]$ for $c_L$ is given, and we use $c_L = 0.07$ as a reference value. In Figure 23, we plot the dual solution after 4 adaptive refinements. To estimate the computational cost of computing approximations $\hat{c}_L$ to the lift coefficient $c_L$, we compute stability factors related to the relative error in $\hat{c}_L$, defined by

$$\frac{|c_L - \hat{c}_L|}{c_L} \leq S_{13}^2(\|h^2 R_1(U_h, P_h)\| + \|h^2 R_3(U_h, P_h)\|) + S_2^2\|h^2 R_2(U_h, P_h)\|, \tag{29}$$

with $S_{13}^s = \|D^s\varphi\|$ and $S_2^s = \|D^s\theta\|$. We note that the stability factors in Table III are approximately 100 times larger than the corresponding stabilty factors for the computation of the drag coefficient

| # dof | $I_{eff}$ |
|---|---|
| 5.656 | 3.36 |
| 7.456 | 13.54 |
| 11.996 | 4.32 |
| 18.336 | 2.53 |
| 33.120 | 2.26 |
| 62.252 | 1.41 |
| 116.616 | 1.27 |
| 225.588 | 0.92 |
| 436.444 | 0.76 |
| 844.956 | 0.66 |

Table II. Effectivity indices $I_{eff} = estimated\ error/true\ error$ for computing the drag coefficient $c_D$ as a function the number of unknowns.

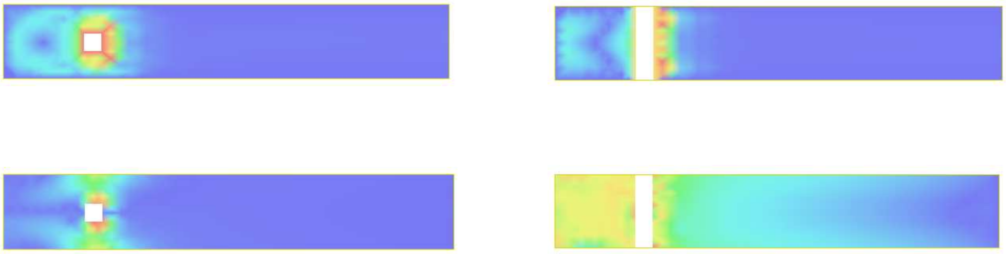

Figure 23. Dual velocity (upper) and pressure (lower) for computing the lift coefficient $c_L$ after 4 adaptive mesh refinements in the $x_1x_2$-plane at $x_3 = 0.205$ (left) and the $x_1x_3$-plane at $x_2 = 0.2$ (right).

in Table I. This indicates that the computational cost needed to obtain an approximation of the lift coefficient $c_L$ with a small relative error is much greater than in the case of the drag coefficient. In Figure 24 we see that with a similar computational cost as in the case of the computation of the drag coefficient we are not able to compute an approximation with a small relative error.

| $s$ | $S_{13}^s$ | $S_2^s$ |
|---|---|---|
| 0 | $2.4 \cdot 10^3$ | $7.6 \cdot 10^2$ |
| 1 | $1.1 \cdot 10^5$ | $1.5 \cdot 10^4$ |
| 2 | $5.3 \cdot 10^6$ | $6.1 \cdot 10^5$ |

Table III. Stability factors for the computation of the lift coefficient $c_L$ with respect to the relative error.

### *14.4. Computation of pressure difference*

We now consider the problem of computing the pressure difference in two points upstream and downstream of the cylinder, defined by $\Delta p = p(x^d) - p(x^u)$, with $(x_1^u, x_2^u, x_3^u) = (0.45, 0.20, 0.205)$ and $(x_1^d, x_2^d, x_3^d) = (0.55, 0.20, 0.205)$. In Schäfer and Turek, 1996, an interval $[0.172, 0.180]$ for $\Delta p$ is given, and we use $\Delta p = 0.176$ as a reference value. In Figure 25 we show the mesh after 4 and 6

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

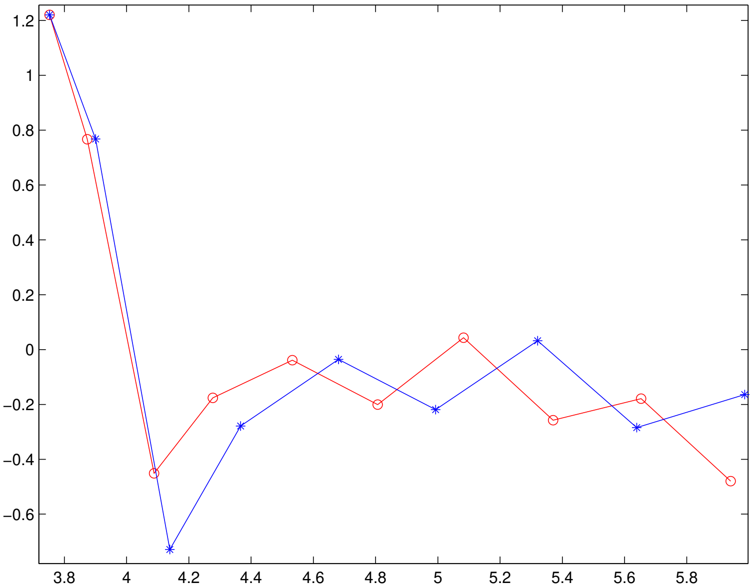


Figure 24. Convergence rates for the computation of the lift coefficient $c_L$, for duality based refinement ('o') and residual based refinement ('*'), as a log-log plot of number of unknowns versus relative errors.

adaptive refinements, with the refinement criterion (25), based on the residuals and the solution to a dual problem. We plot the approximation of the dual problem in Figure 26, after 4 adaptive refinements.

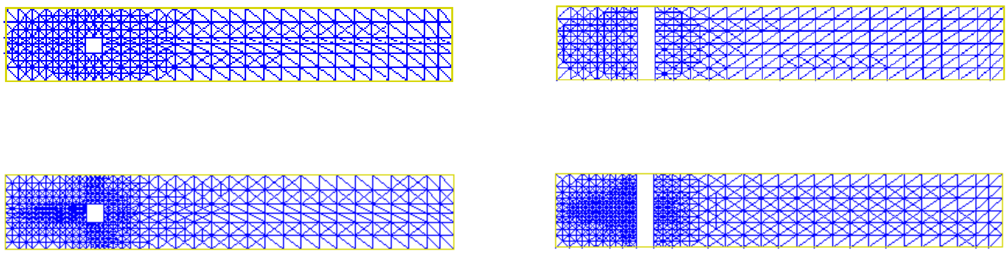

Figure 25. Cross-sections of the meshes in the $x_1x_2$-plane at $x_3 = 0.205$ (left) and in the $x_1x_3$-plane at $x_2 = 0.2$ (right), for computing the pressure difference $\Delta p$, after 4 (upper) and 6 (lower) adaptive mesh refinements.

To estimate the computational cost of computing approximations $\Delta P_h$ to the pressure difference $\Delta p$, we compute stability factors related to the relative error in $\Delta P_h$, defined by

$$\frac{|\Delta p - \Delta P_h|}{\Delta p} \leq S_{13}^2(\|h^2 R_1(U_h, P_h)\| + \|h^2 R_3(U_h, P_h)\|) + S_2^2\|h^2 R_2(U_h, P_h)\|, \qquad (30)$$

with $S_{13}^s = \|D^s\varphi\|$ and $S_2^s = \|D^s\theta\|$. The stability factors in Table IV are of the same order as in the case of the computation of the drag coefficient in Table I, and we therefore expect that we will be able to compute an approximation of $\Delta p$ with a similar relative error to a similar computational cost.

In Figure 27 we compare adaptive mesh refinement using the refinement criterion (25) versus the refinement criterion (26), based solely on the size of the residuals, and we find that the duality based

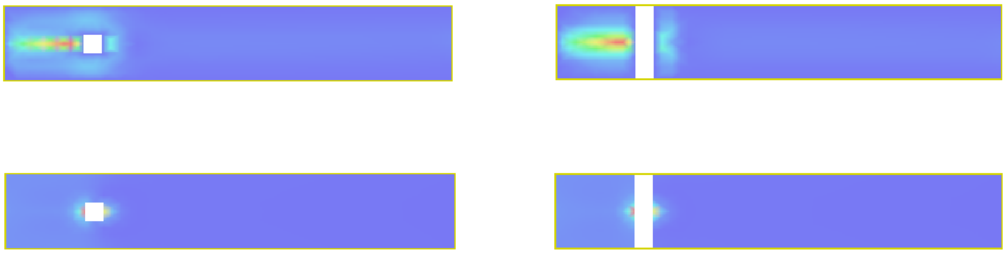

Figure 26. Dual velocity (upper) and pressure (lower) for computing the pressure difference $\Delta p$ after 4 adaptive mesh refinements in the $x_1x_2$-plane at $x_3 = 0.205$ (left) and the $x_1x_3$-plane at $x_2 = 0.2$ (right).

| $s$ | $S_{13}^s$ | $S_2^s$ |
|---|---|---|
| 0 | $2.3 \cdot 10^1$ | $1.7 \cdot 10^1$ |
| 1 | $9.2 \cdot 10^2$ | $1.2 \cdot 10^3$ |
| 2 | $4.6 \cdot 10^4$ | $8.1 \cdot 10^4$ |

Table IV. Stability factors for the computation of the pressure difference $\Delta p$ with respect to the relative error.

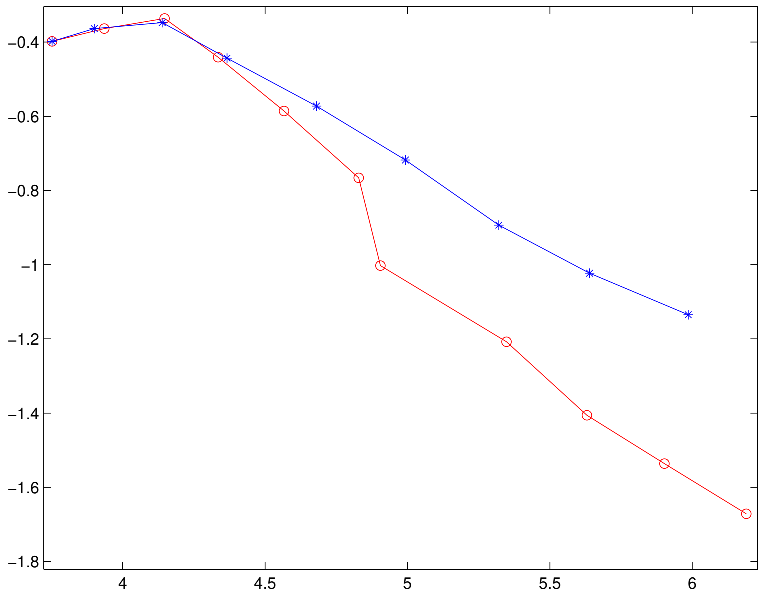


Figure 27. Convergence rates for the computation of the pressure difference $\Delta p$, for duality based refinement ('o') and residual based refinement ('*'), as a log-log plot of number of unknowns versus relative errors.

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

| # dof | $I_{eff}$ |
|---|---|
| 5.656 | 0.85 |
| 8.620 | 0.99 |
| 14.044 | 0.62 |
| 21.636 | 0.71 |
| 36.872 | 0.88 |
| 67.412 | 0.92 |
| 80.392 | 1.11 |
| 222.756 | 1.14 |
| 426.612 | 1.20 |
| 797.940 | 1.08 |

Table V. Effectivity indices $I_{eff} = estimated\, error/true\, error$ for computing the pressure difference $\Delta p$ as a function the number of unknowns.

approach is the better, and in Table V we present effectivity indices for the a posteriori error estimates, which we find to be quite sharp with $I_{eff}$ close to unity.

## Acknowledgments

The first author would like to acknowledge the support by DOE grant DE-FG02-88ER25053.

## REFERENCES

M. Ainsworth and J. T. Oden. A posteriori error estimation in finite element analysis. *Computat. Meth. Appl. Mech. Eng.*, **142**, pp.1-88, 1997.

Barth T, Hoffman J, and Johnson C. *Applied Mathematics: Body and Soul Vol V: Fluid Dynamics*, Springer, 2004.

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

Becker R and Rannacher R. An optimal control approach to a posteriori error estimation in finite element methods. Acta Numerica, pp 1-102, 2001.

Eriksson K, Estep D, Hansbo P and Johnson C. Introduction to Adaptive Methods for Differential Equations. Acta Numerica, pp 105–158, 1995.

Eriksson K, Estep D, Hansbo P and Johnson C. *Computational Differential Equations*, Cambridge University Press, 1996.

Eriksson K, Estep D and Johnson C. *Applied Mathematics: Body and Soul, Vol I-III*, Springer, 2003.

Eriksson K, Johnson C and Logg A. *Parabolic problems* in Encyclopedia of Computational Mechanics, 2003.

Giles M, Larson M, Levenstam M and Süli E. Adaptive error control for finite element approximations of the lift and drag coefficients in viscous flow, Technical Report NA-76/06, Oxford Univ. Computing Laboratory, 1997.

Hoffman J and Johnson C. Adaptive finite element methods for incompressible fluid flow. *2001 NATO-RTO Lecture Series*, Lecture Notes in Computational Science and Engineering, Springer Verlag, 2002a.

Hoffman J and Johnson C. On transition to turbulence in parallel shear flow. submitted to *Journal of Mathematical Fluid Mechanics*, 2002b.

Hoffman J. Computation of functionals in 3d incompressible flow for stationary benchmark problems using adaptive finite element methods. Submitted to *Mathematical Models and Methods in Applied Sciences (M3AS)*, 2003.

Hoffman J. Duality based a posteriori error estimation in various norms and linear functionals for LES. submitted to *SIAM Journal of Scientific Computing*, 2002a.

Hoffman J. Adaptive finite element methods for LES: Computation of the drag coefficient in a turbulent flow around a surface mounted cube. submitted to *SIAM Journal of Scientific Computing*, 2002b.

Hoffman J and Johnson C. Adaptive DNS/LES: a New Agenda in CFD. Chalmers Finite Element Center Preprint 2003-23, Chalmers University of Technology, 2003.

Hoffman J. Computation of mean drag for a square cylinder using Adaptive DNS/LES. In preparation, 2003a.

Hoffman J. Computation of mean drag for a surface mounted cube using Adaptive DNS/LES. In preparation, 2003b.

Johnson C. Adaptive Finite Element Methods for Conservation Laws. *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, Springer Lecture Notes in Mathematics, Springer Verlag, 1997.

S. Krajnović and L. Davidson. Large-Eddy Simulation of the Flow Around a Bluff Body. *AIAA Journal*, **40**, pp.927-936, 2002.

Rannacher R. Finite element methods for the incompressible Navier Stokes equations, Preprint Intsitute of Applied Mathematics, Univ. of Heidelberg, 1999.

Rodi W, Ferziger J.H, Breuer M and Pourquié M. Status of Large Eddy Simulation: Results of a Workshop. *ASME Journal of Fluids Engineering*, **119**, pp. 248-262, 1997.

Sagaut P. *Large Eddy Simulation for Incompressible Flows*. Springer-Verlag, Berlin, Heidelberg, New York, 2001.

Schäfer M, Turek S. Benchmark computations of laminar flow around cylinder, in E.H. Hirschel (editor), *Flow Simulation with High-Performance Computers II*, Volume 52 of Notes on Numerical Fluid Mechanics, pp. 547-566, Vieweg, 1996.

*Encyclopedia of Computational Mechanics*. Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.