COMPUTATION OF MEAN DRAG FOR BLUFF BODY PROBLEMS USING ADAPTIVE DNS/LES

JOHAN HOFFMAN

Abstract. We compute the time average of the drag in two benchmark bluff body problems: a surface mounted cube at Reynolds number 40000, and a square cylinder at Reynolds number 22000, using Adaptive DNS/LES. In Adaptive DNS/LES the Galerkin least-squares finite element method is used, with adaptive mesh refinement until a given stopping criterion is satisfied. Both the mesh refinement criterion and the stopping criterion are based on a posteriori error estimates of a given output of interest, in the form of a space-time integral of a computable residual multiplied by a dual weight, where the dual weight is obtained from solving an associated dual problem computationally, with the data of the dual problem coupling to the output of interest. No filtering is used, and in particular no Reynolds stresses are introduced. We thus circumvent the problem of closure, and instead we estimate the error contribution from subgrid modeling a posteriori, which we find to be small. We are able to predict the mean drag with an estimated tolerance of a few percent using about $10^5$ mesh points in space, with the computational power of a PC.

Key words. Adaptive DNS/LES, adaptive finite element method, duality, a posteriori error estimate, turbulence, large eddy simulation LES, direct numerical simulation DNS, bluff body problem, surface mounted cube, square cylinder

AMS subject classifications. 65M60, 76F65

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. 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 $Re^{-3/4}$, 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 resolve all the features of a turbulent 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 for higher Reynolds numbers one seeks to reduce the problem by some sort of Computational Turbulence Modeling CTM. In a traditional approach to CTM, such as Large Eddy Simulation LES, one seeks to computationally approximate an averaged solution of the Navier-Stokes equations. This is typically done by averaging (or filtering) the Navier-Stokes equations to derive a new set of modified equations, for the averaged variables, to solve. By averaging the non linear term in the Navier-Stokes equations one is left with the problem of closure; to model the so called Reynolds stresses in a subgrid model, which is the major open problem of Computational Fluid Dynamics CFD today. There is an extensive amount of work on LES, in particular regarding the closure problem and the construction of various subgrid models, and we refer to [22] and the references therein for details.

*Courant Institute of Mathematical Sciences, New York University, 251 Mercer Street, New York, NY 10012-1185, USA (hoffman@cims.nyu.edu).
Another important aspect of turbulence simulation is in what sense a solution is accurate, that is, what output from a solution can be considered to be well determined. The computational version of this question is what output can be computed to what tolerance to what cost? This connects to questions of weak uniqueness for the Navier-Stokes equations, investigated in [16, 17, 13]. In particular, full resolution does not automatically give pointwise accuracy, and thus also in the case of DNS questions of computability and predictability are crucial. For example, mean value output are typically less expensive to compute than more pointwise output, see [12, 11, 16, 17, 13].

In this paper we compute the mean drag of two bluff bodies in turbulent flow, using Adaptive DNS/LES, a new approach to CTM. In Adaptive DNS/LES we do not use any filtering, instead we use the Galerkin/least squares space-time finite element method, here referred to as the General Galerkin G2-method, to compute approximate weak solutions to the exact Navier-Stokes equations directly, adaptively refining the computational mesh until a given tolerance is reached with respect to the error in a certain output of interest. Parts of the computational domain are resolved in a DNS, and other parts are left unresolved in a LES with the least-squares stabilization in $G^2$ acting as a dissipative subgrid model. In particular, the adaptive algorithm in itself determines the degree of resolution needed in different parts of the domain to satisfy the stopping criterion. For the computational examples in this paper, the flow is typically fully resolved in a DNS in laminar parts of the domain, whereas the smallest mesh size in the turbulent wake is larger than the theoretically estimated smallest scale of the turbulent flow, and thus we regard the wake as being left unresolved in a LES. The adaptive mesh refinement is based on a posteriori error estimates in the form of a space-time integral of a residual times a dual weight, where the dual weight is obtained by solving an associated linear dual problem with data connecting to the output of interest. The stability of the dual weight under the mesh refinement is crucial, and is controlled computationally through the adaptive algorithm, assuring reliability of the a posteriori error estimates.

Adaptive DNS/LES thus provides an adaptive method for the computation of chosen output, based on quantitative error control, with a key ingredient being the output sensitivity information obtained from the dual problem. No Reynolds stresses are introduced, and we thus circumvent the problem of closure.

The idea of using duality arguments in a posteriori error estimation goes back to Babuška and Miller [2] in the context of postprocessing ‘quantities of physical interest’ in elliptic model problems. A framework for more general situations has since then been systematically developed. For a more detailed account of the development of this framework, including references, we refer to the review papers [6, 4, 14]. For incompressible flow, applications of adaptive finite element methods based on this framework have been increasingly advanced with computation of functionals such as the drag force for 2d stationary benchmark problems in [3, 8], drag and lift forces and pressure differences for 3d stationary benchmark problems in [10], and time dependent problems in 3d in [12]. In [11], this framework is extended to turbulent flow using LES, with a posteriori error estimates with respect to the averaged solution, in terms of the Reynolds stresses that needs to be modelled. These a posteriori error estimates are then used in an adaptive algorithm to compute the drag coefficient for a surface mounted cube in [9].

In [15], the generalization to Adaptive DNS/LES is presented as a general methodology to handle laminar and turbulent flows. Adaptive DNS/LES may be thought of as a straightforward application of the framework for adaptive error control in [6]
to the Navier-Stokes equations, based on a posteriori error estimates, with an error contribution from discretization of the exact Navier-Stokes equations, and a modeling error contribution from the stabilization in $G^2$.

To study the efficiency of Adaptive DNS/LES, we here present results for two benchmark bluff body problems, described e.g. in [21]: (i) a surface mounted cube at Reynolds number 40000, for which some results using Adaptive DNS/LES are included also in [15], and (ii) a square cylinder at Reynolds number 22000. The results in this paper indicate that we are able to predict the drag coefficient in (i)-(ii) with an estimated tolerance of a few percent, using about $10^5$ mesh points in space, with the computational power of a PC.

An outline of this paper is as follows: In Section 2 we recall the Navier-Stokes equations as a model for viscous incompressible flow, and we state the corresponding discretization using $cG(1)cG(1)$, a variant of $G^2$. In Section 3 we present Adaptive DNS/LES $cG(1)cG(1)$, including a posteriori error estimates for drag, and in Section 4 we present computational results for the problems (i)-(ii) using Adaptive DNS/LES $cG(1)cG(1)$.

2. The 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 $\mathbb{R}^3$ (where we assume that $\Omega$ is a polygonal domain) with homogeneous Dirichlet boundary conditions, take the form: Find $(u, p)$ such that

$$
\begin{align*}
\dot{u} + (u \cdot \nabla)u - \nu \Delta u + \nabla p &= f & \text{in } \Omega \times I, \\
\nabla \cdot u &= 0 & \text{in } \Omega \times I, \\
u u &= 0 & \text{on } \partial \Omega \times I, \\
u u(\cdot, 0) &= u_0 & \text{in } \Omega,
\end{align*}
$$

(2.1)

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 $f, u_0, I = (0, T)$, is a given driving force, initial data and time interval, respectively. The quantity $\nu \Delta u - \nabla p$ represents the total fluid force, and may alternatively be expressed as

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

(2.2)

where $\sigma(u, p) = (\sigma_{ij}(u, p))$ is the stress tensor, with components $\sigma_{ij}(u, p) = 2 \nu \epsilon_{ij}(u) - \rho \delta_{ij}$, composed of the stress deviatoric $2\nu \epsilon_{ij}(u)$ with zero trace and an isotropic pressure: here $\epsilon_{ij}(u) = (u_{ij} + u_{ji})/2$ is the strain tensor, with $u_{ij} = \partial u_i / \partial x_j$, and $\delta_{ij}$ is the usual Kronecker delta, the indices $i$ and $j$ ranging from 1 to 3. We assume that (2.1) is normalized so that the reference velocity and typical length scale are both equal to one. The Reynolds number $Re$ is then equal to $\nu^{-1}$.

2.1. Discretization: $cG(1)cG(1)$. The $cG(1)cG(1)$ method is a variant of the $G^2$ method [19, 12] 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 trial functions being continuous piecewise linear. Let $0 = t_0 < t_1 < \ldots < t_N = T$ be a sequence of discrete time steps with associated time intervals $I_n = (t_{n-1}, t_n]$ of length $\Delta t_n = t_n - t_{n-1}$ 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 $T_n = \{K\}$ of mesh size $h_n(x)$ with $W_n$ the functions $v \in W_n$ satisfying the Dirichlet boundary condition $v|_{\partial \Omega} = w$. 

\"ADAPTIVE DNS/LES\"
We 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 (2.1), with homogeneous Dirichlet boundary conditions reads: For \(n = 1, \ldots, N\), find \((U^n_h, P^n_h) \equiv (U_h(t_n), P_h(t_n))\) with \(U^n_h \in V_0^n \equiv [W_0^n]^{3d}\) and \(P^n_h \in W^n\), such that

\[
\begin{align*}
((U^n_h - U^{n-1}_h)k^{-1}_n + \bar{U}^n_h \cdot \nabla \bar{U}^n_h, v) + (2\nu \epsilon(\bar{U}^n_h), \epsilon(v)) - (P^n_h, \nabla \cdot v) \\
+ (\nabla \cdot \bar{U}^n_h, q) + SD_b(\bar{U}^n_h, P^n_h; v, q) = (f, v) \quad \forall (v, q) \in V^n_0 \times W^n,
\end{align*}
\]

(2.3)

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

\[
SD_b(\bar{U}^n_h, P^n_h; v, q) \equiv (\delta_1 (\bar{U}^n_h \cdot \nabla \bar{U}^n_h + \nabla P^n_h - f), \bar{U}^n_h \cdot \nabla v + \nabla q) + (\delta_2 \nabla \cdot \bar{U}^n_h, \nabla \cdot v),
\]

and \(\delta_1 = \frac{1}{2}(k_n^{-2} + |U|^2 h_n^{-2})^{-1/2}\) in the convection-dominated case \(\nu < \bar{U}^n_h h_n\) and \(\delta_1 = \kappa_1 h^2\) otherwise, \(\delta_2 = \kappa_2 h\) if \(\nu < \bar{U}^n_h h_n\) and \(\delta_2 = \kappa_2 h^2\) otherwise, with \(\kappa_1\) and \(\kappa_2\) positive constants of unit size (in this paper we have \(\kappa_1 = \kappa_2 = 1\)), and

\[
(v, w) = \sum_{K \in T^n} \int_K v \cdot w \, dx,
\]

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

We note that the viscous term \((2\nu \epsilon(\bar{U}^n_h), \epsilon(v))\) may alternatively occur in the form

\[
(\nu \nabla U^n_h, \nabla v) = \sum_{i=1}^3 (\nu \nabla (U^n_h)_i \cdot \nabla v_i).\]

In the case of Dirichlet boundary conditions the corresponding variational formulations are equivalent, but not so in the case of Neumann boundary conditions. If we have Neumann boundary conditions, we use the standard technique to apply these boundary conditions weakly.

3. Adaptive DNS/LES cG(1)cG(1). In Adaptive DNS/LES cG(1)cG(1) we do not apply any filter to the Navier-Stokes equations, and thus no Reynolds stresses are introduced. Instead we use cG(1)cG(1) to adaptively compute approximate solutions with the goal of satisfying a given tolerance with respect to the error in a specified output of interest.

Error indicators for mesh refinement and stopping criterions are both based on a posteriori error estimates, in the form of a space-time integral of a computable residual times a dual weight, where the dual weight is obtained from a computational approximation of the solution to an associated dual problem with data coupled to the output of interest.

3.1. 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:

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

(3.1)

where \((u, p)\) solves (2.1) 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 use a standard technique to 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 (2.1) by \( \Phi \) and integrating by parts, we get using the zero boundary condition on \( \Gamma_1 \)

\[
N(\sigma(u), p) = \frac{1}{|I|} \int_I (\dot{u} + u \cdot \nabla u - f, \Phi) - (p, \nabla \cdot \Phi) + (2\nu\epsilon(u), \epsilon(\Phi)) + (\nabla \cdot u, \Theta) \, dt, \tag{3.2}
\]

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 \( \Theta \).

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

\[
N^h(\sigma(U_h, P_h)) = \frac{1}{|I|} \int_I (\dot{U}_h + U_h \cdot \nabla U_h - f, \Phi) - (P_h, \nabla \cdot \Phi) + (2\nu\epsilon(U_h), \epsilon(\Phi)) + SD_s(U_h, P_h; \Phi, \Theta) + (\nabla \cdot U_h, \Theta) \, dt, \tag{3.3}
\]

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 \( \dot{U}_h = (U_h^n - U_h^{n-1})/\Delta t \) on \( I_n \). We note the presence of the stabilizing term \( SD_s \) in (3.3), compared to (3.2). This term is added in order to obtain the independence of \( N^h(\sigma(U_h, P_h)) \) from the choice of \((\Phi, \Theta)\), which follows from (2.3).

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

\[
-\varphi - (u \cdot \nabla) \varphi + \nabla U_h \cdot \varphi - \nu \Delta \varphi + \nabla \theta = 0 \quad \text{in } \Omega \times I, \\
\text{div} \varphi = 0 \quad \text{in } \Omega \times I, \\
\varphi(\cdot, T) = 0 \quad \text{in } \Omega, \tag{3.4}
\]

where \( (\nabla U_h \cdot \varphi)_j = (U_h)_j \cdot \varphi \). 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 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 for the problems in this paper turns out to generate slower growth, as we learn from computing approximations of the dual solution.

3.3. An a posteriori error estimate. To derive a representation of the error \( N(\sigma(u), p) - N^h(\sigma(U_h, P_h)) \), we subtract (3.3) from (3.2) with \((\Phi, \Theta)\) finite element functions, to get

\[
N(\sigma(u), p) - N^h(\sigma(U_h, P_h)) = \frac{1}{|I|} \int_I (\dot{u} + u \cdot \nabla u, \Phi) - (p, \nabla \cdot \Phi) + (2\nu\epsilon(u), \epsilon(\Phi)) + (\nabla \cdot u, \Theta) - ((\dot{U}_h + U_h \cdot \nabla U_h, \Phi) - (P_h, \nabla \cdot \Phi) \, dt. \tag{3.5}
\]
With \((\varphi, \theta)\) the solution to the dual problem (3.4), we also have that
\[
\int_I \left( \frac{d}{dt} (u + u \cdot \nabla u, \varphi) - \left( p, \nabla \cdot \varphi \right) + \left( 2\nu \varepsilon(u), \varepsilon(\varphi) \right) + \left( \nabla \cdot u, \theta \right) \right) dt
\]
\[
- \left( \left( H_u + U_h \cdot \nabla U_h, \varphi \right) - \left( P_h, \nabla \cdot \varphi \right) + \left( 2\nu \varepsilon(U_h), \varepsilon(\varphi) \right) + \left( \nabla \cdot U_h, \theta \right) \right) dt
\]
\[
= \frac{1}{|I|} \int_I \left( \left( \frac{d}{dt} \varphi + u \cdot \nabla \varphi + e \cdot \nabla U_h, \varphi \right) - \left( p - P_h, \nabla \cdot \varphi \right) \right) dt
\]
\[
+ \left( 2\nu \varepsilon(e), \varepsilon(\varphi) \right) + \left( \nabla \cdot e, \theta \right) dt = 0,
\]
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 (3.5) and (3.6), we thus have that
\[
N(\sigma(u, p)) - N^h(\sigma(U_h, P_h)) = \frac{1}{|I|} \int_I \left( \left( \frac{d}{dt} \varphi + u \cdot \nabla \varphi + e \cdot \nabla U_h, \varphi - \Phi \right) - \left( P_h, \nabla \cdot (\varphi - \Phi) \right) \right) dt
\]
\[
- \left( \left( \frac{d}{dt} \varphi + u \cdot \nabla \varphi + e \cdot \nabla U_h, \nabla \cdot (\varphi - \Phi) \right) + \left( 2\nu \varepsilon(U_h), \nabla \cdot (\varphi - \Phi) \right) \right) dt
\]
\[
= \frac{1}{|I|} \int_I \left( \left( \frac{d}{dt} \varphi + u \cdot \nabla \varphi + e \cdot \nabla U_h, \varphi - \Phi \right) - \left( P_h, \nabla \cdot (\varphi - \Phi) \right) \right) dt
\]
\[
+ \left( \nabla \cdot U_h, \theta - \Theta \right) + \left( 2\nu \varepsilon(U_h), \nabla \cdot (\varphi - \Phi) \right) + SD_3(U_h, P_h; \Phi, \Theta) dt,
\]
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 3.1.** 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 (3.4), then
\[
|N(\sigma(u, p)) - N^h(\sigma(U_h, P_h))| = | \sum_{K \in T_n} E_K |,
\]
where \(E_K = e^K_D + e^K_M\) with
\[
e^K_D = \frac{1}{|I|} \int_I \left( \left( \frac{d}{dt} \varphi + u \cdot \nabla \varphi + e \cdot \nabla U_h, \varphi - \Phi \right) - \left( P_h, \nabla \cdot (\varphi - \Phi) \right) \right)_K dt
\]
\[
+ \left( \nabla \cdot U_h, \theta - \Theta \right)_K + \left( 2\nu \varepsilon(U_h), \nabla \cdot (\varphi - \Phi) \right)_K dt,
\]
\[
e^K_M = \frac{1}{|I|} \int_I SD_3(U_h, P_h; \Phi, \Theta)_K dt.
\]
We may view \(e^K_D\) as the error contribution from the discretization on element \(K\), and \(e^K_M\) as the contribution from the subgrid model (stabilization) on element \(K\).

From the error representation in Theorem 3.1 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 \(e^K_D\) from discretization as follows (cf. [11])
\[
e^K_D \leq \frac{1}{|I|} \int_I \left( \left( R_3(U_h, P_h) \right)_K + \left( R_2(U_h, P_h) \right)_K \right) \left( C_h h^2 ||D^2 \varphi||_K + C_k k ||\varphi||_K \right)
\]
\[
+ \left( R_3(U_h) \right)_K \left( C_h h^2 ||D^2 \theta||_K + C_k k ||\theta||_K \right) dt,
\]
where the residuals $R_i$ are defined by
\begin{align*}
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) &= |\text{div } U_h|,
\end{align*}
(3.7)
with
\begin{equation}
D_2(U_h)(x, t) = \max_{y \in \partial K} (h_n(x))^{-1} |[\frac{\partial U_h}{\partial n}(y, t)]| (3.8)
\end{equation}
for $x \in K$, with $[\cdot]$ the jump across the element edge $\partial K$. $D^2$ denotes second order spatial derivatives, and we write $|w|_K \equiv (|||w_1||_K, |||w_2||_K, |||w_3||_K)$, with $|||w||_K = (w, w)_K^2$, and let the dot denote the scalar product in $\mathbb{R}^3$.

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, and 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.

Replacing the exact dual solution $(\varphi, \theta)$ by a computed approximation $(\varphi_h, \theta_h)$, we are led to the following a posteriori error estimate:
\begin{equation}
|N(\sigma(u, p)) - N^h(\sigma(U_h, P_h))| \approx \sum_{K \in T_h} \mathcal{E}_{K,h} (3.9)
\end{equation}
where $\mathcal{E}_{K,h} = e^K_{D,h} + e^K_{M,h}$ with
\begin{align*}
e^K_{D,h} &= \frac{1}{|I|} \int_I [(|R_1(U_h, P_h)|_K + |R_2(U_h, P_h)|_K) \cdot (C_h^2 h^2 |\varphi_h|_K + C_k k |\varphi_h|_K) \\
&\quad + |R_3(U_h)|_K \cdot (C_k h^2 |\text{div } \theta_h|_K + C_k k |\theta_h|_K)] \, dt, \\
e^K_{M,h} &= \frac{1}{|I|} \int_I SD_3(U_h, P_h; \varphi_h, \theta_h)_K \, dt,
\end{align*}
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 (stabilization) on element $K$.

Remark 3.2. Non-Dirichlet boundary conditions, such as slip conditions at lateral boundaries and transparent outflow conditions, introduce additional boundary terms in the error representation in Theorem 3.1. Since the dual solutions for the examples in this paper are small at such non-Dirichlet boundaries, we neglect the corresponding boundary terms in the computations.

Remark 3.3. In the computations we use $C_h = 1/2$ and $C_k = 1/8$ as constant approximations of the interpolation constants in Theorem 3.1. These values are motivated by simple analysis on reference elements.

3.4. An adaptive algorithm. For simplicity, we have the space mesh and time step $k$ being constant in time, with the time step being equal to the smallest element diameter in the space mesh. In the computations we use an algorithm for adaptive mesh refinement in space based on the a posteriori error estimate (3.9), of the form: Given an initial coarse computational space mesh $T^0$, start at $k = 0$, then do
\begin{enumerate}
\item Compute approximation to the primal problem using $T^k$,
\end{enumerate}
(2) Compute approximation to the dual problem using $T^k$.

(3) If $\sum_{K \in T_h} |E^k_{K,h}| < TOL$ then STOP, else:

(4) Refine a fraction of the elements in $T^k$ with largest $E^k_{K,h} \rightarrow T^{k+1}$.

(5) Set $k = k + 1$, then goto (1).

By refining the mesh size $h$, also the time step $k$ and the subgrid model from stabilization $SD_h$ are refined implicitly, since both $k$ and $\delta$ depend on $h$. The subgrid model may thus be thought of as being implicitly modified as we refine the mesh. But the error estimate (3.9) also allows for a direct adaptive modification of both the time step $k$ and the subgrid model $SD_h$, by locally refining $k$ and $\delta$, respectively. In this paper, we have not investigated the potential gain in using direct local refinement of $k$ and $\delta$, or using a space mesh varying in time.

We note that the structure of the error estimate (3.9) allows for a general interpretation of the subgrid model $SD_h$; we may think of $SD_h$ as being the part of the numerical method not being a standard Galerkin (here $cG(1)cG(1)$) discretization of the Navier-Stokes equations.

4. Numerical results. We now seek to compute the drag coefficient $c_D$, a mean value in time of a normalized drag force. Due to computational cost we have to truncate the time interval over which we average, and thus we approximate $c_D$ by $\hat{c}_D$, a normalized drag force averaged over a given time interval $I$, defined by

$$\hat{c}_D = \frac{2}{U_h^2 A} \times N(\sigma(u, p)), \quad (4.1)$$

where $U_h$ is a bulk inflow velocity, $A$ is an area, and $N(\sigma(u, p))$ is defined by (3.1). In computations, we approximate $\hat{c}_D$ by $\hat{c}^h_D$, defined by

$$\hat{c}^h_D = \frac{2}{U_h^2 A} \times N^h(\sigma(U_h, P_h)), \quad (4.2)$$

with $N^h(\sigma(U_h, P_h))$ defined by (3.3). Thus we may use a scaled version of the a posteriori error estimate (3.9) to estimate the error $|\hat{c}_D - \hat{c}^h_D|$.

Below we will show that in the case of two bluff body benchmark problems, it is possible to use Adaptive DNS/LES to compute approximations $\hat{c}^h_D$, such that the error $|\hat{c}_D - \hat{c}^h_D|$ is less than a few percent, using the computational power of a PC.

4.1. Flow around a surface mounted cube. We first consider a basic benchmark problem of CFD of computing the drag of a surface mounted cube at Reynolds number 40000. 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 in this problem is thus very complex with a combination of laminar and turbulent features including boundary layers and a large turbulent wake, see Figure 4.2.

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 (2.1) with kinematic viscosity $\nu = 2.5 \times 10^{-6}$ and a unit inlet bulk velocity corresponding to a Reynolds number of 40000, using the dimension of the cube as characteristic dimension. The inlet velocity profile is interpolated from experiments, 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 seek to approximate the drag coefficient $c_D$, given by (4.1), averaged over a time interval $I = [0, 40H]$ at fully developed flow. We set $U_h = 1$ based on the bulk inflow velocity, and the area $A = H \times H = H^2$. This is a standard benchmark problem at the CDE-Forum [1], where also the inlet velocity profile is available for download.

In Figure 4.1 we plot the approximations $c_h$ as we refine the mesh. We find that for the finer meshes $c_h \approx 1.5$, a value that is well captured already using less than $10^5$ mesh points. Without the contribution from the stabilizing terms in (3.3), we get a somewhat lower value for $c_h$, where the difference is large on coarser meshes but small (less than 5%) on the finer meshes.

We know of no experimental reference values of $c_D$, but in [5] a DNS is performed using about $70 \times 10^6$ degrees of freedom, with the same data as in this paper, giving $c_D \approx 1.42$, where the stabilizing terms in (3.3) is not used for the evaluation of $c_D$. The results in [5] should thus be compared to the somewhat lower values in Figure 4.1, resulting in a good agreement.

In [20], $c_D$ is approximated using LES. 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 [20]. Using LES with different meshes and subgrid models, approximations of $c_D$ in the interval [1.14, 1.24] are presented.

4.1.1. The dual solution. A snapshot of the dual solution is shown in Figure 4.3. We note that the dual solution, with boundary data on the cube, 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 15 adaptive mesh refinements is shown in Figure 4.3. The initial space mesh is uniform and very coarse, 384 mesh points, and 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.

4.1.2. A posteriori error estimates. In Figure 4.4 we plot the (normalized) a posteriori error estimates \(e_{D,h}\) and \(e_{M,h}\) from (3.9), as well as the true error based on the computational approximation on the finest mesh. The modeling error \(e_{M,h}\)
Fig. 4.3. Surface mounted cube: dual velocity $|\varphi|$ (upper), dual pressure $|\theta|$ (middle), and the resulting computational mesh (lower), in the $x_1x_2$-plane at $x_3 = 3.5H$ and in the $x_1x_3$-plane at $x_2 = 0.5H$. 
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 the evaluation of $e_{M,h}$ in Figure 4.4, we have set the absolute values inside the sums in space and time.

We find that once the value for $c_h$ has stabilized, the a posteriori error estimates indicate that it may be hard to further increase the precision.

4.2. Flow around a square cylinder. We now consider the benchmark problem of a square cylinder at Reynolds number 22000, based on the cylinder diameter $D = 0.1$ and the inflow velocity. 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 $x_2 = 7D$. We have a unit inflow velocity, no slip boundary conditions on the cylinder, slip boundary conditions on the lateral boundaries, and a transparent outflow boundary condition.

As described e.g. in [21, 7], characteristics of this flow are a turbulent wake of approximate diameter $1D$ attached to the trailing face of the cylinder, and two opposite shear layers periodically shedding vortices, see Figure 4.6. In addition, many authors have reported on a cycle-to-cycle variation, so called phase jitter, due to turbulence and 3d instabilities in the shear layers. In Figure 4.7, we show a time series of the vorticity for a computation on a fine mesh illustrating the transition in the shear layers.

The presence of phase jitter leads to complications when evaluating time averages, since the time averages are highly dependent of the size and location of the averaging interval, which has led to alternative ways to represent averages. One approach is to consider so called phase averages [18], where a number of shedding cycles are chosen as “typical” for the flow, over which mean values are computed.
We seek to approximate the drag coefficient $c_D$, given by (4.1), averaged over a time interval $I = [0, 100D]$ at fully developed flow, where we set $U_h = 1$ based on the bulk inflow velocity, and the area $A = 4D \times D = 4D^2$.

In Figure 4.5 we plot the computed values of $c_D^h$, as we refine the mesh. In Figure 4.5 we have also included the value for $c_D^h$ without the stabilizing term in
(3.3). For the finer meshes we get a \( c_D \) in the interval 2.0-2.4, and a value about 5% lower for the formulation without the stabilizing term. The variation in \( c_D \) could to a certain degree be explained by effects of phase jitter, as noted above. In Figure 4.5 we plot the trajectory of the normalized drag force for the finest mesh, where we note variations in amplitude and local mean of the drag.

In [21], various reference values for this problem, including mean drag, is reported. Experimental reference values for \( c_D \) is reported to be in the interval 1.9-2.1, where the experiments are carried out with slightly different parameters, such as a slightly lower Reynolds number, a longer cylinder, and a turbulent level of 2% in the inflow velocity. LES results are reported in the interval 1.66-2.77, and RANS results in the interval 1.6-2.0. To test the sensitivity in inflow data, we compare with a similar computation with 2% white noise added to the inflow velocity. The results are plotted in Figure 4.5, giving similar values for \( c_D \), although possibly somewhat lower, closer
Fig. 4.7. Square cylinder: time evolution of the vorticity $|\nabla \times u|$. 
to the experimental results. As for the rest of the output data presented in [21], our results show a very good agreement, which we present in Table 4.1, where we give our results with Adaptive DNS/LES for the finest mesh with about $1.2 \times 10^5$ mesh points in space, after 9 adaptive mesh refinements.

Again we note the sensitivity in mean drag, where we from Figure 4.5 can see that translating the averaging interval results in different $c_D^h$. The computation of $c_D^h$ in Table 4.1 corresponds to the interval $[10; 20]$ in Figure 4.5, and we see that translating this interval suitably would result in a lower $c_D^h$, closer to the experimental reference values.

<table>
<thead>
<tr>
<th>output</th>
<th>experiment</th>
<th>LES</th>
<th>RANS</th>
<th>Adaptive DNS/LES</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c_D$</td>
<td>1.9–2.1</td>
<td>1.66–2.77</td>
<td>1.637–2.004</td>
<td>2.40 (2.31)</td>
</tr>
<tr>
<td>$c_{D,\text{rms}}$</td>
<td>0.1–0.2</td>
<td>0.10–0.27</td>
<td>-</td>
<td>0.17 (0.17)</td>
</tr>
<tr>
<td>$c_L$</td>
<td>-</td>
<td>-0.3–0.007</td>
<td>-</td>
<td>-0.02</td>
</tr>
<tr>
<td>$c_{L,\text{rms}}$</td>
<td>0.1–1.4</td>
<td>0.38–1.79</td>
<td>-</td>
<td>0.84</td>
</tr>
<tr>
<td>$St$</td>
<td>0.132</td>
<td>0.066–0.15</td>
<td>0.134–0.143</td>
<td>0.14</td>
</tr>
</tbody>
</table>

**Table 4.1**

Square cylinder: reference values from [21]: drag and lift ($c_D$ and $c_L$), and corresponding rms-values ($c_{D,\text{rms}}$ and $c_{L,\text{rms}}$), and Strouhal number $St$. The values in the parenthesis correspond to evaluation of drag without contribution from the stabilizing term.

**Fig. 4.8.** Square cylinder: log$_{10}$-log$_{10}$ plot of the (normalized) a posteriori error estimates $e_{D,h}$ (‘o’) and $e_{M,h}$ (‘x’), and the true error (‘*’) based on $c_D = 2.2$, as functions of the number of mesh points in space.

4.2.1. Dual solution. A snapshot of the dual solution is shown in Figure 4.9. We note that the dual solution, as in the case of the surface mounted, is of moderate size, and is concentrated in space. The resulting computational mesh after 9 adaptive mesh refinements is shown in Figure 4.10. We notice that again the adaptive method
Fig. 4.9. Square cylinder: dual velocity $|\varphi|$ (upper), and dual pressure $|\theta|$ (lower), after 9 adaptive mesh refinements with respect to mean drag, in the $x_1x_3$-plane at $x_2 = 7D$ and in the $x_1x_2$-plane at $x_3 = 2D$, with the darker color representing high values.

automatically captures the turbulent wake, which is essential for accurately computing drag.

4.2.2. A posteriori error estimates. In Figure 4.8 we plot the (normalized) a posteriori error estimates $e_{D,h}$ and $e_{M,h}$ from (3.9), as well as an estimate of the true error based on the computational approximations on the finest meshes, suggesting that 2.2 may be a good candidate for a representative value of $\tilde{c}_D$. In Figure 4.8, we set the absolute values inside the sums in space and time of $e_{D,h}$ and $e_{M,h}$. Thus error cancellation is not possible, leading to conservative error estimates.

4.3. Reliability and efficiency of the adaptive method. We now focus on two key points relating to the reliability and efficiency of the adaptive method based on the a posteriori error estimate (3.9), which directly couples to whether this estimate
Fig. 4.10. Square cylinder: computational mesh after 9 adaptive mesh refinements with respect to mean drag, in the $x_1x_3$-plane at $x_2 = 7D$ and in the $x_1x_2$-plane at $x_3 = 2D$.

indeed gives a reasonably sharp bound of the true error, or not. The two key points are (i) replacement of $u$ by a computed velocity $U_h$ in the dual problem, and (ii) 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 (i) the convection coefficient and (ii) numerical computation.

We first consider the discretization error term $e_{D,h}$. We obtain a rough estimate of 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 $cG(1)cG(1)$ we have that $\| \sqrt{h}R_1(U_h, P_h) \|$ is bounded (recalling that $\delta_1 \sim h$, and neglecting the time derivative). 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 4.11 we display the variation of $\| hD^2\varphi_h \|$ as a function of the number mesh points in space for the bluff body problems, and we find that after some initial refinements the dual solution shows a stability of $\| hD^2\varphi_h \|$ under the mesh refinement.

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 $SD_\delta(U_h, P_h; U_h, P_h)$, Cauchy’s 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 4.12 that $||\nabla \varphi_h||$ shows a stability under the mesh refinement suggesting that indeed $\varepsilon_{M,h}$ may get below a moderate tolerance under refinement without reaching a DNS.

Thus we conclude that the crucial computed dual weights show a stability under mesh refinement which indicates that the a posteriori error estimate (3.9) may be reliable and also reasonably efficient. In particular, we have given evidence that the net effect on the dual weights from replacing $u$ by $U_h$ in the dual problem may be
small, even though this may correspond to locally a large perturbation since $U_h$ cannot be expected to approximate $u$ pointwise.

4.4. Turbulent dissipation. For accurate approximation of the drag in bluff body problems we need to capture the correct global dissipation, which follows from the basic global energy balances for the Navier-Stokes equations (2.1) and $cG(1)cG(1)$ (2.3), obtained by multiplication by $(u, p)$ and choosing $(v, q) = (U_h, P_h)$ respectively,
Fig. 4.13. Volume of the turbulent wake, defined as the part of the domain with mean dissipation intensity of $D(U_h, P_h) > 0.2$, vs $\log_{10}$ of the number of mesh points, for the surface mounted cube (upper) and the square cylinder (lower).

to get

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

$$\frac{d}{dt} ||U_h||^2 \approx \int_{\Gamma_{in}} P_h U_h \cdot n \, ds - \int_{\Gamma_{out}} P_h U_h \cdot n \, ds - (\nu \|\epsilon(U_h)\|^2 + SD(u, P_h; U_h, P_h)),$$
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 $D_v(u) = \nu \|\epsilon(u)\|^2$ represents the exact global dissipation (rate), and $D(U_h, P_h) = D_v(U_h) + SD(U_h, P_h; U_h, P_h)$ the corresponding cG(1) 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 
\[ \frac{d}{dt} \|u\|^2 \approx \frac{d}{dt} \|U_h\|^2 \sim 0, \]
we thus have that the pressure drop \( \sim \) global dissipation, and thus we expect that the drag \( \sim \) global dissipation.

In Figure 4.13 we show the development under the mesh refinement process of the volume of the turbulent wake. 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 expansion of the turbulent wake to parallel an expansion of the wake as the Reynolds number increases.

In Figure 4.14 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 \( D_v(U_h) \), which is small, and the intensity of \( SD_i(U_h, P_h; U_h, P_h) \) 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 \( D_v(u) \), 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 finer part of 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 \( D_v(u) \).

5. Summary and future directions. We have considered two bluff body benchmark problems, where we have shown that we are able to compute the drag coefficient to an estimated tolerance of a few percent, using about \( 10^5 \) mesh points in space, with the computational power of a PC.

The mesh refinement criterion and the stopping criterion of the adaptive algorithm are based on a posteriori error estimates of an output of interest, in the form of a space-time integral of a residual times a dual weight. The dual weight is obtained from the computational approximation of an associated dual problem with data coupled to the output of interest. We have presented computational evidence that crucial properties of the dual solution are stable under the adaptive mesh refinement.

In Adaptive DNS/LES no filtering is used, but instead output from the exact Navier-Stokes equations is approximated directly, using cG(1)cG(1) with the least-squares stabilization acting as a dissipative subgrid model. In particular, we circumvent the problem of closure as no modeling of Reynolds stresses is necessary. Instead we estimate the error contribution from subgrid modeling a posteriori, which we find to be small.

Encouraged by the results in this paper, we plan to extend the study of Adaptive DNS/LES to more turbulence benchmark problems. In addition, there are many potential optimizations of the adaptive algorithm to investigate. For example, one would like to consider the possible gains of using dynamically changing meshes, adaptively refined time steps, and adaptive choice of different time steps in different parts of the domain. Also, one would like to adaptively choose \( \delta \) in the stabilization \( SD_i \), or more general to adaptively choose the form of \( SD_i \).

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

REFERENCES