

Adaptive computation of aeroacoustic sources for a rudimentary landing gear

Rodrigo Vilela de Abreu^{1,*},†, Niclas Jansson² and Johan Hoffman²

¹*Computational Technology Laboratory, Department of High Performance Computing and Visualization, School of Computer Science and Communication and Linné FLOW Centre, KTH, SE-10044 Stockholm, Sweden*

²*Computational Technology Laboratory, Department of High Performance Computing and Visualization, School of Computer Science and Communication, KTH, SE-10044 Stockholm, Sweden*

SUMMARY

We present our simulation results for the benchmark problem of the flow past a rudimentary landing gear using a General Galerkin FEM, also referred to as adaptive DNS/LES. In General Galerkin, no explicit sub-grid model is used; instead, the computational mesh is adaptively refined with respect to an *a posteriori* error estimate of a quantity of interest in the computation, in this case, the drag force on the rudimentary landing gear. Turbulent boundary layers are modeled using a simple wall-layer model with the shear stress at walls proportional to the skin friction, which here is assumed to be small and, therefore, can be approximated by zero skin friction. We compare our results with experimental data and other state of the art computations, where we find good agreement in sound pressure levels, surface velocities, and flow separation. We also compare with detailed surface pressure experimental data where we find largely good agreement, apart from some local differences for which we discuss possible explanations. Copyright © 2013 John Wiley & Sons, Ltd.

Received 31 August 2012; Revised 19 July 2013; Accepted 12 October 2013

KEY WORDS: aeroacoustics; aerodynamics; finite element; incompressible flow; LES, large Eddy simulations; turbulent flow

1. INTRODUCTION

This work may be seen as an extension of our contribution to the BANC-I NASA/AIAA workshop held in conjunction with the 16th AIAA/CEAS Aeroacoustics Conference in Stockholm, 2010. One of the principal goals of the workshop was to assess available computational methods for the computation of aeroacoustic sources by comparing simulation results with experimental work. The workshop consisted of a set of four benchmark problems, and we chose to solve the problem of the flow past a rudimentary landing gear (RLG).

The RLG geometry was developed at Boeing Commercial Airplanes and is a non-proprietary, highly simplified version of a four-wheel landing gear. It consists of a square-based sustaining post, a wheel frame with square-based axles, and the four wheels (Figure 1). One of the reasons of using such a simplified geometry in a workshop is that by being non-proprietary, it allows for unrestricted sharing of data and open publication of results. Furthermore, the boundary layers are ‘tripped’ to turbulent on the surface of all four wheels, so that separation is mainly defined by the sharp edges of the geometry, simplifying the comparison between different turbulence models (for a detailed description of the boundary layer tripping method, see [1]).

*Correspondence to: Rodrigo Vilela de Abreu, Computational Technology Laboratory, Department of High Performance Computing and Visualization, School of Computer Science and Communication and Linné FLOW Centre, KTH, SE-10044 Stockholm, Sweden.

†E-mail: rvda@kth.se

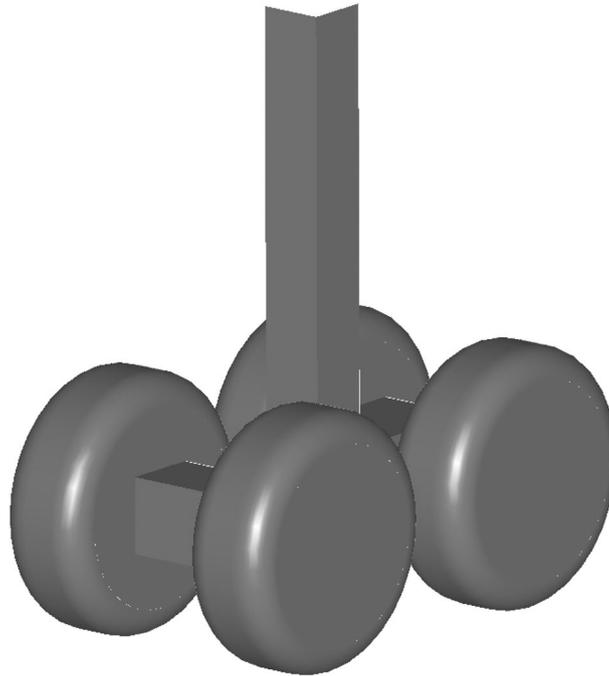


Figure 1. The rudimentary landing gear geometry.

The reason why we chose the RLG problem is closely related to the boundary layers being turbulent on most of the wheel surfaces. This weakens the Reynolds number (Re) dependence of the flow, approaching the inviscid limit of $Re \rightarrow \infty$. We are then able to test our hypothesis where we question the importance of resolving turbulent boundary layers at high Re flows [2]. This has generally been regarded as highly controversial, and the RLG benchmark offered, therefore, an excellent opportunity for us to test our assumptions. Moreover, from an industrial perspective, landing gears are highly complex geometries, and there are certainly a few CFD frameworks that could currently produce accurate, time-resolved solutions for it (time-resolved solutions are required to capture the aeroacoustic aspects of the flow), mainly due to the limitations imposed by the complexity of the geometries involved and by mesh size requirements.

The General Galerkin (G2) method [3] is an adaptive FEM that may be seen as an implicit LES, because no explicit subgrid stress model is used. No LES filter is used either; instead, turbulent kinetic energy is dissipated through the numerical stabilization of G2. One of the main strengths of this method is the use of unstructured tetrahedral meshes that are automatically refined by an underlying adaptive algorithm. This generates meshes that are optimally refined with respect to some integral flow quantity, which makes the computation of turbulent G2 solutions cost-effective if compared with well-established, standard CFD methods. Therefore, we believe that the G2 method is a competitive and a novel approach to the computation of aeroacoustic sources.

General Galerkin has previously been validated for academic benchmark problems. This is the first time, however, that a problem derived directly from a real industrial application can be solved and compared with detailed experimental data. Thus, a careful analysis and presentation of our results are a valuable contribution to the field, opening up for further development of G2-related methodologies in the field of computational aeroacoustics (CAA).

The goals of our paper may be summarized as follows:

1. to introduce our method to the aeroacoustics community and to validate its capabilities, solving, for the first time, a realistic industrial problem and comparing the results with carefully designed experiments;
2. to test the assumptions underlying the method, specially regarding the lack of importance of a resolved boundary layer for high Re flows;

3. to confirm the cost-effectiveness and the flexibility of G2, which we believe are high, mainly due to the use of an optimized, unstructured mesh.

The paper is divided in five sections, including this introduction. Section 2 summarizes the computational method, briefly explaining the adaptive mesh refinement, and stating our choice of boundary conditions. In Section 3, we present our results, including the computational mesh and detailed comparison of simulation and experiments. Section 3.6 brings a description of the utilized computational resources and, finally, concluding remarks (Section 4) summarize the paper.

2. SIMULATION METHODOLOGY

The simulation methodology uses a FEM with piecewise linear approximations in space and time and with a numerical stabilization in the form of a weighted least squares method based on the residual. The incompressible Navier–Stokes equations (INSE) are discretized directly, without applying any filter. Thus, the method does not approximate an LES-filtered solution but is instead an approximation of a *weak solution* (a weak solution is a function that satisfies the weak form of the INSE, which means that less regularity—obtained by integration by parts of the weak form—is demanded than for a classical, strong solution, see [3] for a detailed discussion). For this method, we have *a posteriori* error estimates of quantities of interest with respect to a weak solution, which form the basis for our adaptive mesh refinement algorithm. The *a posteriori* error estimates are based on the solution of an associated adjoint problem, similar to the optimal control problem of finding the smallest mesh possible such that the error in a quantity of interest is smaller than a chosen tolerance. We refer to this method as G2 or adaptive DNS/LES. It is a time-resolved method, being closely related to an implicit LES, as no explicit subgrid model is used. The numerical stabilization of G2 can be seen as an implicit subgrid model that dissipates energy in the turbulent parts of the flow identified by the residual. G2 is validated for a number of standard benchmark problems in the literature [4–7].

2.1. The cG(1)cG(1) method

As the basic model for incompressible Newtonian fluid flow, we consider the INSE with constant kinematic viscosity $\nu > 0$, enclosed in $\Omega \subset \mathbb{R}^3$, with boundary Γ , over a time interval $I = (0, T]$:

$$\begin{aligned} \dot{u} + (u \cdot \nabla)u + \nabla p - \nu \nabla \cdot \epsilon(u) &= f, & (x, t) \in \Omega \times I, \\ \nabla \cdot u &= 0, & (x, t) \in \Omega \times I, \\ u(x, 0) &= u^0(x), & x \in \Omega, \end{aligned} \quad (1)$$

with $u(x, t)$ as the velocity vector, $p(x, t)$ as the pressure, $u^0(x)$ as the initial data, and $f(x, t)$ as a body force. Moreover, $\sigma_{ij} = -\nu \epsilon_{ij}(u) + p \delta_{ij}$ is the stress tensor, with the strain rate tensor $\epsilon_{ij}(u) = 1/2(\partial u_i / \partial x_j + \partial u_j / \partial x_i)$, and δ_{ij} is the Kronecker delta function. The relative importance of viscous and inertial effects in the flow is determined by the *Re* $Re = UL/\nu$, where U and L are characteristic velocity and length scales, respectively.

We use the cG(1)cG(1) method, which is based on the continuous Galerkin method cG(1) in space and time. With cG(1) in time, the trial functions are continuous piecewise linear, and the test functions are piecewise constant; cG(1) in space, on the other hand, corresponds to both test functions and trial functions being continuous piecewise linear. Let $0 = t_0 < t_1 < \dots < t_N = T$ 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 let $W \subset H^1(\Omega)$ be a finite element space consisting of continuous piecewise linear functions on a tetrahedral mesh $\mathcal{T} = \{K\}$ of mesh size $h(x)$, with W_w the functions $v \in W$ satisfying the Dirichlet boundary condition $v|_{\Gamma} = w$.

We seek $\hat{U} = (U, P)$, continuous piecewise linear in space and time, and the cG(1)cG(1) method for INSE with homogeneous Dirichlet boundary conditions reads: for $n = 1, \dots, N$, find

$(U^n, P^n) \equiv (U(t_n), P(t_n))$, with $U^n \in V_0 \equiv [W_0]^3$ and $P^n \in W$, such that

$$\begin{aligned} & ((U^n - U^{n-1})k_n^{-1} + \bar{U}^n \cdot \nabla \bar{U}^n, v) + (2\nu\epsilon(\bar{U}^n), \epsilon(v)) - (P^n, \nabla \cdot v) \\ & + (\nabla \cdot \bar{U}^n, q) + SD_\delta^n(\bar{U}^n, P^n; v, q) = (f, v), \quad \forall \hat{v} = (v, q) \in V_0 \times W, \end{aligned} \quad (2)$$

where $\bar{U}^n = \frac{1}{2}(U^n + U^{n-1})$ and P^n are piecewise constant in time over I_n , with the stabilizing term

$$\begin{aligned} SD_\delta^n(\bar{U}^n, P^n; v, q) \equiv \\ (\delta_1(\bar{U}^n \cdot \nabla \bar{U}^n + \nabla P^n - f), \bar{U}^n \cdot \nabla v + \nabla q) + (\delta_2 \nabla \cdot \bar{U}^n, \nabla \cdot v), \end{aligned} \quad (3)$$

and

$$\begin{aligned} (v, w) &= \sum_{K \in \mathcal{T}_n} \int_K v \cdot w \, dx, \\ (\epsilon(v), \epsilon(w)) &= \sum_{i,j=1}^3 (\epsilon_{ij}(v), \epsilon_{ij}(w)), \end{aligned}$$

with the stabilization parameters $\delta_1 = \kappa_1 (k_n^{-2} + |U^{n-1}|^2 h_n^{-2})^{-1/2}$ and $\delta_2 = \kappa_2 U^{n-1} h_n$, where κ_1 and κ_2 are positive constants of unit size. For turbulent flow, we choose a time step size $k_n \sim \min_{x \in \Omega} h_n / |U^{n-1}|$.

2.2. The adjoint Navier–stokes equations

To construct an appropriate adjoint problem for a *posteriori* error estimation of approximate solutions (U, P) using a G2 method, there are several possibilities: we may (i) use a weak formulation of Equation (1) to construct a problem where the errors $(u - U, p - P)$, act as testfunctions for a set of adjoint variables (φ, θ) , or (ii) we may instead use a similar approach with the discrete weak form, Equation (2).

Using the first approach, we obtain the following formulation for the continuous adjoint problem (see [3] for a detailed discussion):

$$\begin{aligned} -\dot{\varphi} - (u \cdot \nabla)\varphi + \nabla U \cdot \varphi + \nabla \theta - \nu \Delta \varphi &= \psi_1 && \text{in } \Omega \times I \\ \nabla \cdot \varphi &= \psi_2 && \text{in } \Omega \times I \\ \varphi &= \psi_3 && \text{on } \Gamma \times I, \\ \varphi(\cdot, T) &= \psi_4 && \text{in } \Omega, \end{aligned} \quad (4)$$

where we find that the structure is similar to the primal INSE equations, except that the adjoint problem is linear, the transport is backward in space time, and that we have a reaction term $(\nabla U \cdot \varphi)_j = U_{,j} \cdot \varphi$, not present in the primal INSE.

A weak formulation of Equation (4) takes the form: find $(\varphi, \theta) \in L_2(I; [H_{\psi_3}^1(\Omega)]^3 \times L_2(\Omega))$, with $\dot{\varphi} \in L_2(I; [L_2(\Omega)]^3)$ and $\varphi(\cdot, T) = \psi_4$, such that

$$\begin{aligned} \int_I -(v, \dot{\varphi}) - (v, (u \cdot \nabla)\varphi) + (v, \nabla U_h \cdot \varphi) + (2\nu\epsilon(v), \epsilon(\varphi)) \\ -(q, \nabla \cdot \varphi) + (\nabla \cdot v, \theta) \, dt = \int_I (v, \psi_1) + (q, \psi_2) \, dt, \end{aligned} \quad (5)$$

for all $(v, q) \in L_2(I; [H_0^1(\Omega)]^3 \times L_2(\Omega))$, given the data $\psi_1 \in L_2(I; [L_2(\Omega)]^3)$, $\psi_2 \in L_2(I; L_2(\Omega))$, $\psi_3 \in L_2(I; [L_2(\Gamma)]^3)$, and $\psi_4 \in [L_2(\Omega)]^3$. The data ψ_i correspond to error estimates of different output functionals. For example, nonzero ψ_1 and ψ_2 correspond to point values or mean values of the velocity or the pressure, respectively; nonzero ψ_4 corresponds to error estimates of the velocity at the final time, T ; nonzero ψ_3 corresponds to error estimates in surface integrals, for example, drag and lift forces.

In the present computations, the mesh was refined to control the error in the drag force, which corresponds to $\psi_1 = \psi_2 = \psi_4 = 0$ and $\psi_3 = (1, 0, 0)$. To compute numerical approximations of the adjoint solution, we use a similar cG(1)cG(1) method as for the primal problem [3].

2.3. *A posteriori* error estimate for cG(1)cG(1)

For G2, we have an *a posteriori* error estimate based on the residual of Equation (1) and on the solution of a corresponding adjoint problem, Equation (4), which constitutes the kernel of G2’s mesh refinement algorithm. With that, we intend to estimate the error in a functional of interest, $M(\hat{U})$, which may be seen as a weighted average of the solution \hat{U} of Equation (2) over space and time, for example, the average drag force of a flow obstacle. Thus, we do not aim at correctly computing the solution pointwise but rather at solving the INSE in an ‘average sense’, which is more meaningful for chaotic, highly unsteady turbulent flows. This technique has been primarily used under the mathematical framework of various FEMs and is extensively described in, for example, [8–10].

Theorem 1

If $\hat{U} = (U, P)$ solves Equation (2), $\hat{u} = (u, p)$ is a weak INSE solution, and $\hat{\varphi} = (\varphi, \theta)$ solves an associated adjoint problem with data $M(\cdot)$, then we have the following *a posteriori* error estimate for the output $M(\hat{U})$ with respect to the reference output $M(\hat{u})$:

$$|M(\hat{u}) - M(\hat{U})| \leq \sum_{n=1}^N \left[\int_{I_n} \sum_{K \in \mathcal{T}_n} |R_1(\hat{U})|_K \cdot \omega_1 dt + \int_{I_n} \sum_{K \in \mathcal{T}_n} |R_2(U)|_K \omega_2 dt + \int_{I_n} \sum_{K \in \mathcal{T}_n} |SD_\delta^n(\hat{U}; \hat{\varphi})_K| dt \right]$$

with

$$\begin{aligned} R_1(\hat{U}) &= \dot{U} + (U \cdot \nabla)U + \nabla P - \nu \Delta U - f, \\ R_2(U) &= \nabla \cdot U, \end{aligned} \tag{6}$$

where $SD_\delta^n(\cdot; \cdot)_K$ is a local version of the stabilizing form, Equation (3), and the stability weights are given by

$$\begin{aligned} \omega_1 &= C_{n,K}^h h_{n,K} |\nabla \varphi|_K + C_{n,K}^k k_{n,K} |\dot{\varphi}|_K, \\ \omega_2 &= C_{n,K}^h h_{n,K} |\nabla \theta|_K + C_{n,K}^k h_{n,K} |\dot{\theta}|_K, \end{aligned}$$

where $h_{n,K}$ is the diameter of element K in the mesh \mathcal{T}_n , and $C_{n,K}^h$ represents interpolation constants. Moreover, $|w|_K \equiv (\|w_1\|_K, \|w_2\|_K, \|w_3\|_K)$, with $\|w\|_K = (w, w)_K^{1/2}$ and $(v, w)_K = \int_K v \cdot w dx$, and the dot denotes the scalar product in \mathbb{R}^3 .

We note that the *a posteriori* error estimate naturally splits into two parts: (i) a standard Galerkin error contribution and (ii) a modeling error contribution from the perturbation of the equations by the stabilization. For the proof of Theorem 1, see [3] (pp.223-225).

2.4. *Adaptive algorithm*

The adaptive mesh refinement algorithm is based on the *a posteriori* error estimate:

$$|M(\hat{u}) - M(\hat{U})| \leq \sum_{K \in \mathcal{T}_n} \mathcal{E}_K,$$

with the error indicator

$$\mathcal{E}_K \equiv \sum_{n=1}^N \left[\int_{I_n} |R_1(\hat{U})|_K \cdot \omega_1 dt + \int_{I_n} |R_2(U)|_K \omega_2 dt + \int_{I_n} |SD_\delta^n(\hat{U}; \hat{\varphi})_K| dt \right]$$

for each element K in the mesh \mathcal{T}_n . Starting from a coarse mesh \mathcal{T}_0 , we mark a certain fraction of all elements for refinement based on the error indicator. The algorithm reads

1. for the mesh \mathcal{T}_n , compute the primal problem and the adjoint problem;
2. mark the 10% of the elements with highest \mathcal{E}_K for refinement;
3. generate the refined mesh \mathcal{T}_{n+1} , and go to 1.

The development of a sharp stop criterion for the refinement algorithm previously mentioned is work in progress; we typically stop refining the mesh when convergence in $M(\hat{U})$ is observed.

2.5. Turbulent boundary layers and wall boundary conditions

Resolving turbulent boundary layers for a realistic industrial problem is not yet feasible, and therefore, modeling is applied in order to reduce the need for mesh resolution near walls and to correctly apply boundary conditions. The earliest wall models for LES were developed for simple geometry flows, such as channel and pipe flows, and have been extended and developed throughout the years to cover more complex cases.

Our choice of wall model, which has been documented in previous works [2, 11], dates back to Schumann in 1975, who did LES of turbulent flows in plane channels and annuli [12]. In his work, Schumann assumed the shear stress at the wall to be proportional to the tangential velocity in the first cell adjacent to the wall, with constant of proportionality depending on the local mean wall shear stress and the mean tangential velocity. In our simulations, closely following Schumann's idea, we append the INSE, Equation (1), with the following boundary conditions:

$$u \cdot n = 0, \quad (7)$$

$$\beta u \cdot \tau_k + n^T \sigma \tau_k = 0, \quad k = 1, 2, \quad (8)$$

for $(x, t) \in \Gamma_{solid} \times I$, with $n = n(x)$ an outward unit normal vector, $\tau_k = \tau_k(x)$ orthogonal unit tangent vectors of the solid boundary Γ_{solid} , and β a friction parameter, which needs to be chosen. We use matrix notation with all vectors v being column vectors, and the corresponding row vector is denoted as v^T .

With skin friction boundary conditions, the rate of kinetic energy dissipation in cG(1)cG(1) has a contribution of the form

$$\sum_{k=1}^2 \int_0^T \int_{\Gamma_{solid}} |\beta^{1/2} \bar{U} \cdot \tau_k|^2 ds dt,$$

from the kinetic energy which is dissipated as friction in the boundary layer.

For the present work, because the boundary layers are tripped to fully turbulent on the wheels of the RLG, we assume the flow to be equivalent to a very high Re flow, approaching the inviscid limit of $Re \rightarrow \infty$. In this idealized situation, the viscous stresses at the walls vanish, which corresponds to a slip boundary condition for the tangential velocity. For that reason, we put $\beta = 0$ in our computations, reducing, thus, the need of high mesh resolution near walls.

2.6. Simulation geometry and boundary conditions

Flow measurements on the RLG were performed in a wind tunnel with a square test section. In order to reproduce the configuration in the experiments, we have also constructed a wind tunnel around the landing gear for the simulations. When applying inlet and outlet boundary conditions, however, one has to be careful not to position them too close to the model, in order to avoid any undesired numerical effects on the flow region of interest. In Figure 2, a sketch of the computational domain is presented, defining all relevant dimensions. Distances and lengths are normalized with respect to the wheel diameter, D .

A Dirichlet type condition of constant inflow velocity was prescribed at the inlet of the domain. At the outlet, we have assumed a zero stress boundary condition, which corresponds to a homogeneous Robin type condition for the stress, $\sigma \cdot n = \nu \nabla u \cdot n - pn = 0$. If the viscosity is small, and if

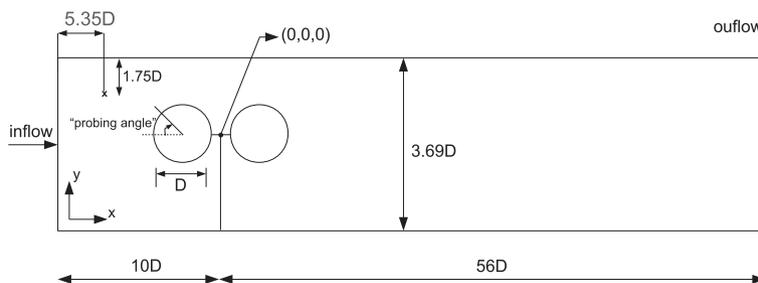


Figure 2. Geometry sketch and definition of ‘probing angles’.

there are no large gradients of velocity in the flow direction at the outlet, this can be approximated by a homogeneous Dirichlet boundary condition for the pressure. As discussed previously, a slip condition has been applied on the remaining boundaries (the tunnel walls and the RLG surface).

3. RESULTS

3.1. Normalization and simulation times

In the experiments, the velocity at the wind-tunnel inlet was 40 m/s , and the Mach number was $M = 0.12$. The wheel diameter was $D = 0.4064\text{ m}$ and $Re = 10^6$.

However, all computations have been performed in non-dimensional units, assuming incompressible flow, and the involved quantities have been normalized according to the instructions in the problem statement [13]. The free-stream velocity in our computations is therefore $1U_\infty$, and the wheel diameter is $1D$. Time units are given in D/U_∞ . Pressure coefficients are referenced to the static pressure at $-4.65D$ upstream of the RLG ($5.35D$ downstream of the domain inlet), in the center of the tunnel in the spanwise direction (i.e., $Z = 0$), ca. $1.75D$ from the top wall, see Figure 2. The pressure fluctuation levels are presented in dB , assuming a dynamic pressure of 960 Pa .

The total simulation time of the ‘final run’ (the one with the finest mesh, where we sampled statistics) was $240D/U_\infty$, of which approximately $130D/U_\infty$ were considered as the start-up phase. In other words, statistical quantities were sampled over $110D/U_\infty$, which we consider long enough to obtain converged averages and fluctuation levels. The time-step used in the last part of the simulation was $0.0013D/U_\infty$.

Prior to the final run, the adaptive mesh refinement algorithm was executed as follows: for each iteration, the primal equations were solved for $40D/U_\infty$, and the corresponding adjoint equations were solved for the last $5D/U_\infty$ of the primal solution; both equations were discretized using the same mesh, and the time-step was also the same.

3.2. Computational mesh

We have chosen to present our mesh under results, because it was obtained by computing successive solutions of the INSE, and corresponding adjoint problems, as described in Section 2.

Our initial volume mesh consisted of ca. 500,000 cells only. Using adaptive mesh refinement with respect to the pressure drag, we have obtained, after seven iterations of the adaptive algorithm, a final mesh consisting of ca. 6,000,000 cells. The smallest mesh sizes on the surface of the RLG are approximately $0.01D$. We have used this final mesh to obtain the data required from CFD.

Some of the features of the adaptive refinement can already be seen in Figure 3. At the wind tunnel bottom wall, downstream of the RLG’s post, we see a refined region extending up to approximately $5D$ in the wake. This corresponds to a region of high levels of wall pressure fluctuation, the refinement correctly ‘reveals’ relevant flow features throughout the domain, which are not always known *a priori*.

Figure 4 shows the surface mesh on the RLG alone. Again, this figure shows how the adaptive refinement works. A good example is the difference in cell size between the stagnation region on

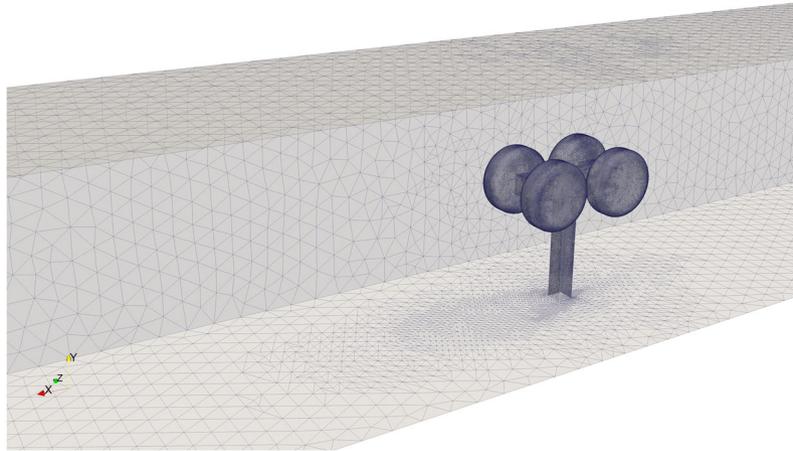


Figure 3. Overview of the mesh.

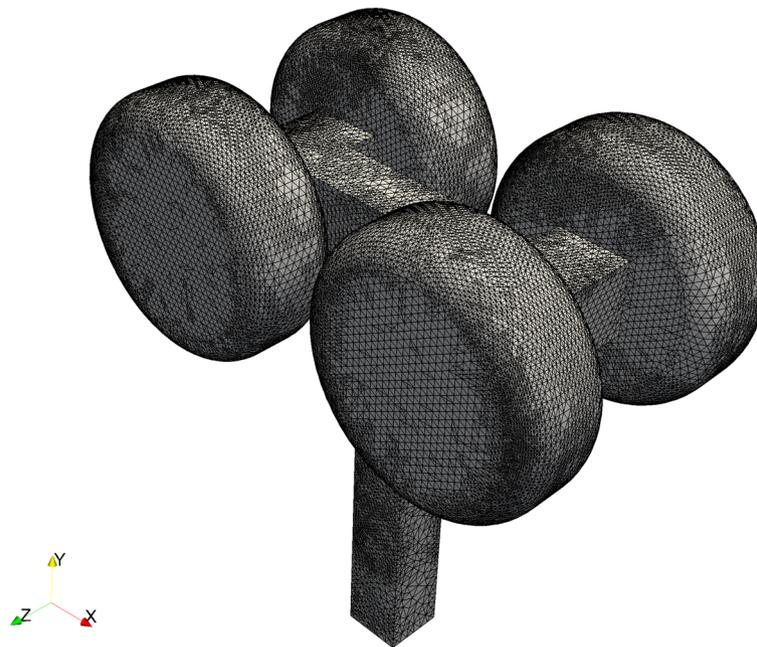


Figure 4. Surface mesh on rudimentary landing gear.

the front wheels and its corresponding wheel shoulders, where the flow accelerates creating a low pressure zone, whose limits coincide with the mesh refinement pattern.

Figure 5 shows two longitudinal cuts of the volume mesh, one at the center of the tunnel and another passing through the wheels at one side of the model. Clearly, the refinement is more dense near the center of the tunnel, where turbulence fluctuations are expected to be more intense. Another interesting feature observed in Figure 5 is that the mesh roughly follows the separation pattern at the center of the model thus, resolving important features of the unsteady turbulent flow in that region.

Conclusively, we would like to emphasize that, by controlling the error of a specific flow quantity (in this case the drag), we are able to generate an optimally refined, unstructured mesh in a much shorter time—a considerable amount of time is saved by avoiding long-lasting meshing procedures, common to the generation of the structured meshes required in other computational methods. Moreover, due to our choice of boundary layer model, we do not need to construct a boundary layer mesh, which further simplifies the mesh generation.

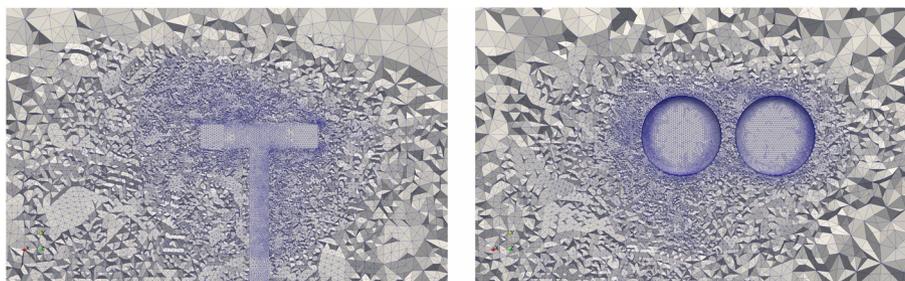


Figure 5. Features of the volume mesh.

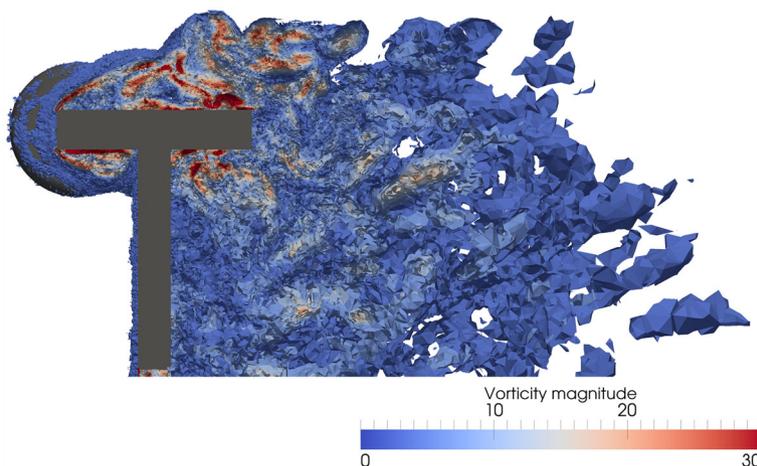


Figure 6. Vorticity magnitude.

3.3. Flow visualization

We start by showing contours of vorticity (Figure 6). We can clearly see the LES content of our simulation in the neighborhood of the RLG and up to $3D$ downstream of the origin of the cartesian coordinate system. This is a typical characteristic of our adaptive method, where the flow is resolved in the most relevant areas in order to minimize the error (in drag in this particular case). The remaining part of the domain remains under-resolved.

In Figure 7, we visualize the average velocity field on the surface of the RLG by contours of velocity magnitude and surface streamlines. From the figure, we see that the flow is completely attached to the outer surface of both front and rear wheels, forming two vortical structures on the back ‘shoulders’ of the wheels; these structures indicate where the flow separates in the simulation. Although the separation pattern in the experiments partly differs from our results, similar vortical structures are also visualized in the oil film images of the front wheels. A more detailed discussion of these results may be found in Section 3.5.

Figure 8 shows wall pressure fluctuation levels and the PSD for an unsteady pressure probe on the surface of the rear wheel, corresponding to *probe 28* for a *probing angle* of 180° (see Figure 10 for the probe numbering and the definition of probing angle). The results are presented in *dB* and *Strouhal number*, $St = fU_\infty D^{-1}$. The obtained levels of the wall pressure fluctuations lie between 110dB and 150dB , which is in good agreement with previously published computational work [1]. Also in good agreement is the butterfly-shaped pattern of high fluctuation levels (above 145dB) on the bottom wall of the wind tunnel. These are caused by the strong pressure fluctuations in the wake downstream of the RLG’s sustaining post. As described in Section 3.2, these fluctuations are correctly captured by the mesh refinement algorithm, without any *a priori* knowledge of their existence in the solution. Another important result that can be observed in Figure 8 is the areas of the RLG

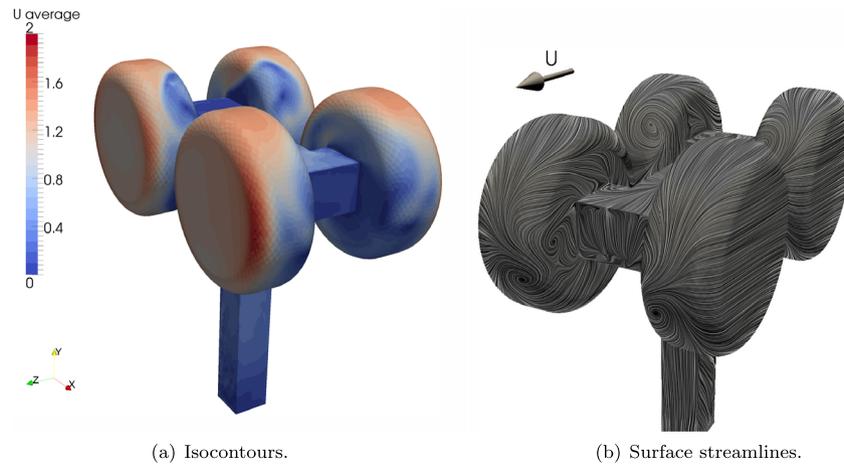


Figure 7. Average velocity field, (a) isocontours and (b) surface streamlines.

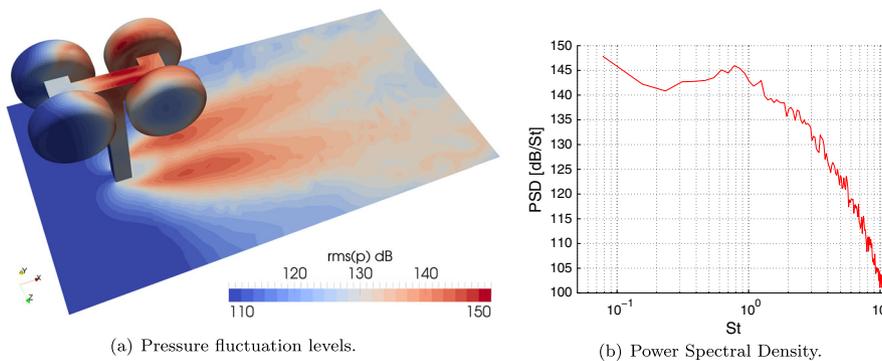


Figure 8. Wall pressure fluctuation levels (a) and PSD (b). The PSD was performed for a point on the back shoulder of the rear wheel, facing the turbulent wake downstream of the rudimentary landing gear.

with the highest pressure fluctuation levels, namely the aft part of the RLG's support structure, where the flow that separates from the frontal wheel axle re-attaches and turbulence impingement generates fluctuation levels of ca. 150dB , and the frontal, inner part of the rear wheels, where the flow re-attaches after separating from front wheels. These two areas, together with the wake-generated downstream of the sustaining post, are most probably the main sources of aeroacoustic noise for the RLG configuration. The PSD, generated by averaging over 10 windows with 50% overlap during the time $110D/U_\infty$, is shown to indicate that we have a frequency resolution that is comparable to similar computational efforts, [1].

3.4. Aerodynamic forces

We have computed the total aerodynamic force on the RLG, and we present this result in terms of non-dimensional coefficients in Table I, together with the results obtained by the other groups participating in the workshop and the measured values. Because the geometry is symmetric about the XY -plane, we expect the side force coefficient, c_z , to oscillate about zero, which is correctly predicted by the simulation (and thus omitted from the table). The same does not apply for the lift, c_y , because the geometry is not symmetric about the XZ -plane.

The flow accelerates past the gap between the wheels and the wind tunnel top wall, which generates a positive lift (with respect to the frame of reference used in the computations). In fact, the time average of c_y is 0.21. This is the result that compares worst with the experimental value for all the groups; it is worth to mention here, however, that we have refined the mesh in order to correctly compute the mean aerodynamic drag, and that, therefore, the mesh is not optimized to compute

Table I. Number of mesh cells and aerodynamic coefficients of all groups participating in the BANC-I workshop in Stockholm, 2010.

Group name	c_x	c_y	Number of cells
KTH	1.78	0.21	6 M
CDA	1.70	0.20	29 M
NTS	1.70	0.28	10 M
KHI	1.81	0.18	41 M
EXA	1.77	0.23	36 M
TUB	1.74	0.23	11 M
Expt	1.60	0.65	—

CDA, CD-adapco; NTS, New Technologies and Services; KHI, Kawasaki Heavy Industry; EXA, EXA corporation; TUB, TU Berlin; Expt, measured.

the lift. One idea would be to refine the mesh with respect to the aerodynamic forces in the other directions too. The drag coefficient, c_x , is the highest one of the three coefficients, oscillating about 1.78. As described in Section 2.5, we do not account for surface friction in our computations, so that the computed drag is solely an effect from pressure differences arising on the surface of the RLG. This result compares relatively well with other groups' results and also with the measured value of 1.60, and we expect it to improve with further mesh refinements. Indeed, in our contribution to the 17th AIAA/CEAS Aeroacoustics Conference in Portland, 2011 [14], we have further refined the mesh and obtained $c_x = 1.59$ after a total of nine iterations (two more than in the present work). Unfortunately, a detailed comparison using those results is not possible, because the simulation was performed for a shorter time window (due to time constraints), and the pressure was not sampled on the points corresponding to the location of the experimental pressure probes.

3.5. Comparison with measurements

Comparison with experimental results will be used as a base for the validation of the method. Point-wise measurements of average pressure coefficients, c_p , on both wheels on one side of the model are available. During the simulation, we have saved the values of pressure on corresponding points and a direct, quantitative comparison is, therefore, possible.

Before proceeding with the analysis, we need to state a possible source of error in postprocessing our results. When mapping the coordinates of the ca. 20,000 available experimental points onto our mesh, we faced an unfortunate difficulty: many of the points fell outside of the domain. In order to be able to sample the pressure over the entire wheel surfaces, we had to project all points (radially when possible), onto a slightly larger and wider wheel.

Figure 9 shows the distribution of c_p on the wheel surfaces. For the front wheel, the computed and simulated distributions are similar, with a low-pressure region where the flow accelerates on

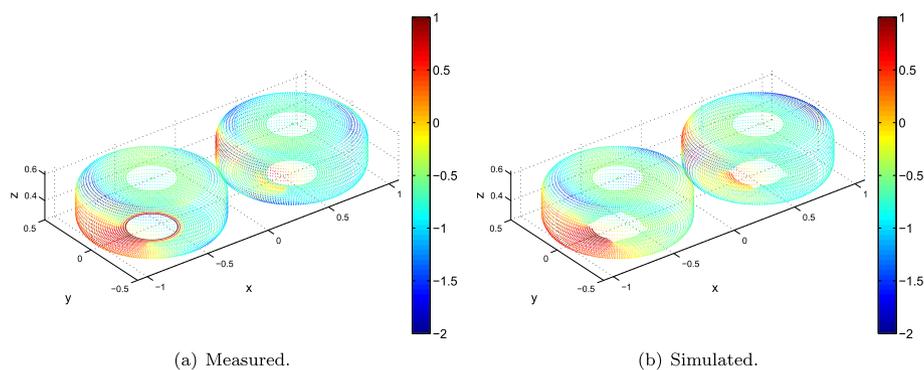


Figure 9. Average (a) measured and (b) simulated pressure coefficient.

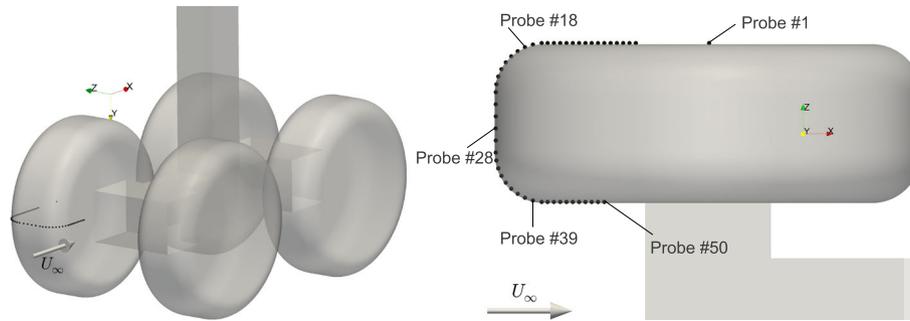


Figure 10. Probe locations and numbering at a probing angle of 0° . The remaining probing angles are obtained by incremental rotation of the wheel around the negative Z -axis 2° at a time.

the front shoulder and high pressure zones on the inner surface, due to the presence of the landing gear support frame. We believe that the high-pressure values measured on the inner ring of the inner surface, close to the wheel support, are due to errors in the measurements.

For the rear wheel, we note a somewhat stronger difference in the pressure distribution. In the experimental results, there are two regions of low pressure that are diametrically opposed, slightly deviating from the equatorial line of the model (on the clockwise direction). These low-pressure zones are not encountered in the simulation. Instead, a low-pressure zone, symmetrically placed about the equatorial line, is formed on the aft part of the outer surface of the wheel.

In order to compare the surface pressure distribution more quantitatively, Figure 12 shows the mean pressure coefficient as a function of probe number over the front and rear wheels, respectively, for different probing angles (refer to Figure 2 for the definition of probing angle). We have chosen to present the results for selected angles only, because the trends displayed in the plots are closely reproduced on the remaining directions. Figure 10 illustrates the position of the probes for 0° —probes aligned with the flow—on the front wheel. In that figure, we see that the probes are numbered from 1 to 50, starting at the center of the *outer surface* of the wheel, following around the *wheel shoulder* (located between *probes 17–40*), and ending on the *inner surface* somewhat upstream of the wheel axle. The remaining probing angles were obtained by simply rotating the wheel 2° at a time around the negative Z -axis. The same probing procedure was performed with probes installed on both front and rear wheels. *Trip disks* were used to force the transition of the boundary layer from laminar to turbulent; they were placed at all angles somewhere near *probe 17* and *probe 40*.

We start by scrutinizing the pressure distribution on the front wheel. For probing angles lower than 70° , we obtained good agreement for most of the probes. For 0° , however, we observe a slight deviation in the location of the stagnation point ($c_p = 1.0$), but the shape of our pressure distribution follows the experimental curve closely.

At 70° , we see that the curves overlap at the outer surface of the wheel, but we fail to predict both the magnitude of the pressure trough located at *probe 19* and the pressure recovery as we move towards the inner surface of the wheel (although the shape of the experimental curve is still captured).

One possible explanation for the difference in pressure magnitude at the trough could be the presence of the boundary layer trip disk around *probe 17* in the experiments; we do not model it in our simulations, but it could cause a local separation bubble in the experiments, which would then explain the lower pressure trough.

For 90° , we again have good agreement on the outer surface and up to *probe 23* on the shoulder, with the remaining probes showing poor agreement. One reason for that could be the presence of a sharp shear layer caused by separation at the frontal axle of the RLG (Figure 11): an accurate computation of its average location and thickness is crucial for correctly predicting the pressure distribution underlying it, and the mesh resolution we used in this simulation might not be sufficient to capture such phenomenon to its smallest details.

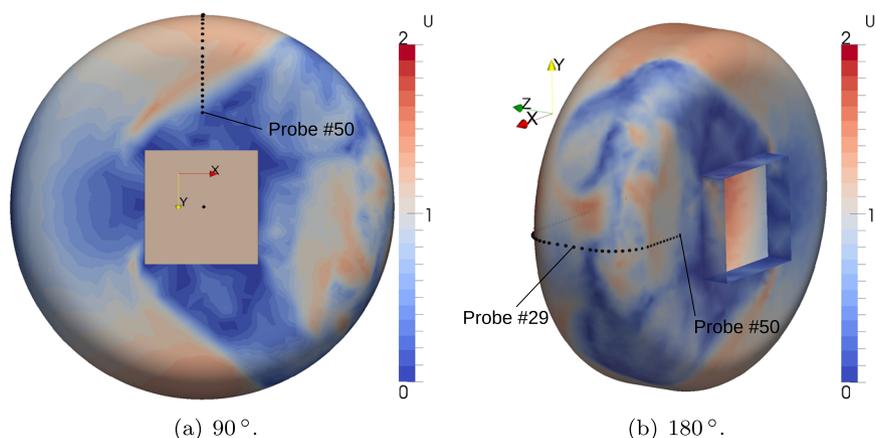


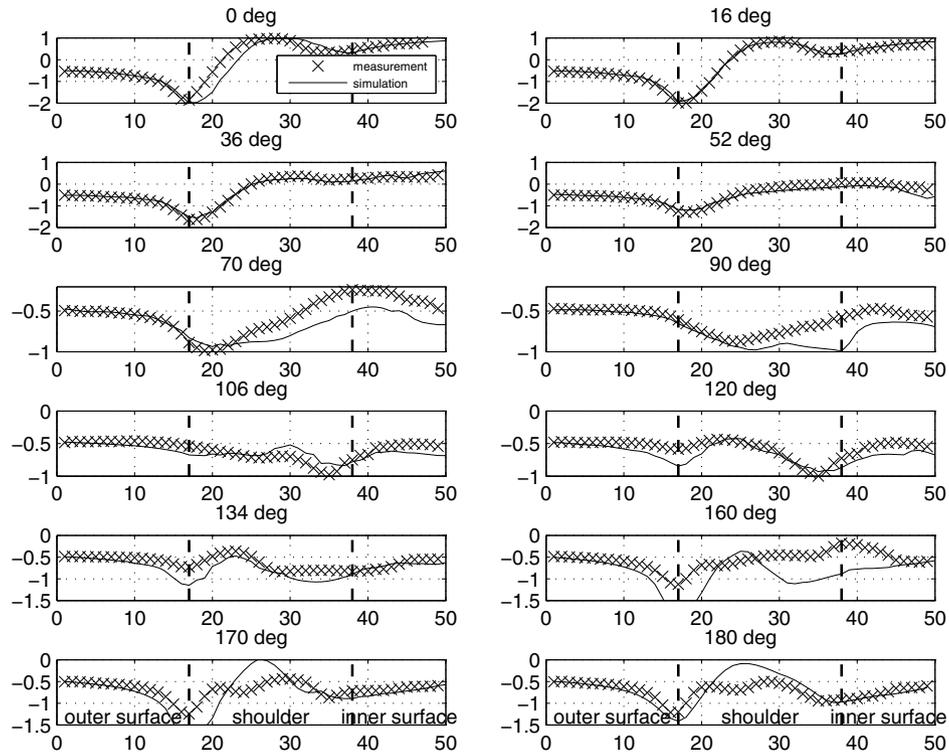
Figure 11. Contours of velocity magnitude on the front wheel. The probes are positioned at the *probing angles* of (a) 90° and (b) 180° .

One general observation that could also explain the poor agreement for the pressure distributions at probing angles of 70° and 90° is that the flow on the shoulder and inner surface for these particular angles does not contribute significantly to the pressure drag computation (our refinement target), and, thus, the algorithm does not capture the flow on those areas properly. One idea would be to refine the mesh with respect to both drag and lift forces in order to obtain better agreement of the pressure distribution for a larger part of the wheel.

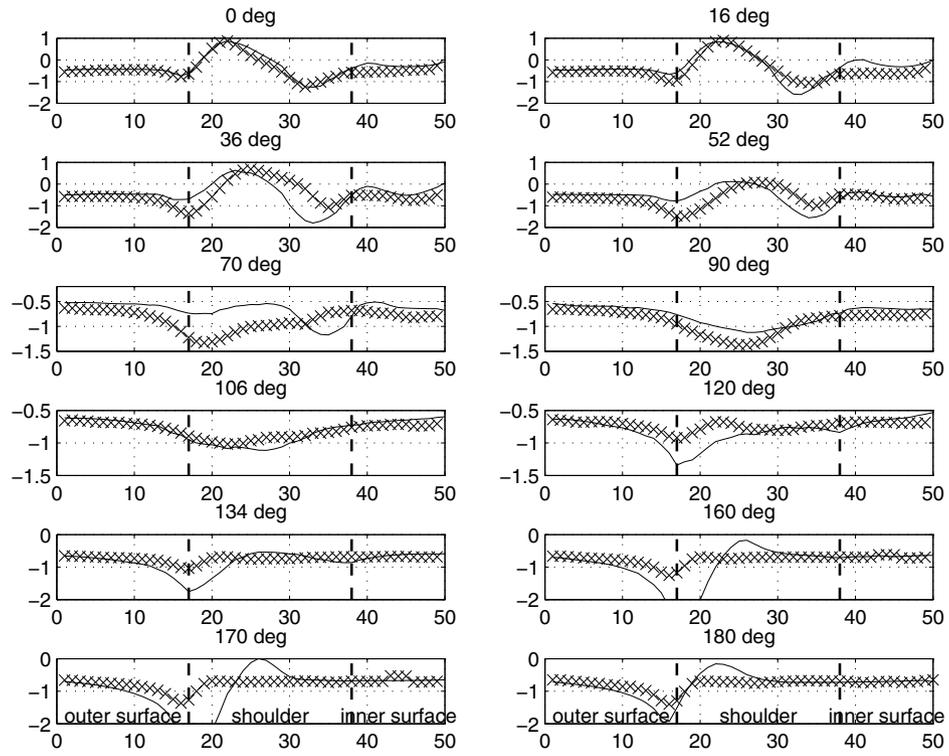
For angles larger than 90° , good agreement is again obtained, with the level and shape of the measurement curves being relatively reproduced by the simulation, this up to a probing angle of ca. 140° . The somewhat large errors in the pressure coefficient for larger angles connect to one important aspect regarding our choice of wall layer model. Oil film visualization from the experiments and simulations performed by other groups participating in the BANC-I workshop [1] indicate that separation occurs at the outer surface of both front and rear wheels, close to the wheel shoulder, at probing angles larger than 150° . Figure 7 shows surface streamlines from our simulation, and we do not see any separation as suggested by the experiments. In our case, with zero skin friction, the flow does not separate from the wheels' outer surface, but instead, it 'leaves' the back shoulder of the front wheels in a vortex that is almost parallel to the free stream flow direction—and that is clearly visualized in Figure 7.

The effect of this difference in separation pattern on the mean surface pressure distribution can be seen in the curves for 170° and 180° in Figure 12. The simulation curves show a pressure recovery in the region covered by *probes* 22–26, whereas the experimental curves present a local minimum around *probes* 24–25, possibly due to separation. Despite these differences, similar 'streamwise vortical structures' are also seen in the oil film visualization and on the surface streamlines from other simulations [1]. It is certainly difficult to pinpoint the exact 'average location' of these highly unsteady flow structures (Figure 11 for the probe positioning in the separation area) and a small deviation, due to, for example, too short sampling time in the simulation, will probably result on large variations in the curves for probing angles larger than 150° . This could also explain the large deviations of our curves from the experimental values for 160° , 170° , and 180° .

On the rear wheel, we observe good agreement for 0° and 16° , showing that we were able to correctly predict the re-attachment of the flow shed from the frontal wheels and axle. This is an important result, because it indicates that the flow field predicted in the frontal half of the model is, at least, satisfactory. Moreover, the re-attachment region is important for the accurate computation of both aerodynamic drag and turbulent sound. The remaining curves also show good agreement, specially on the inner surface of the rear wheel; major discrepancies are seen, however, on the region near the outer trip disk (around *probe* 17), mainly for probing angles larger than 120° . As we discussed previously, this is probably due to the fundamental difference in the separation mechanisms in our simulations, which models $Re \rightarrow \infty$, such that separation can only occur at point stagnation,



(a) Front wheel.



(b) Rear wheel.

Figure 12. Pressure coefficient as a function of probe number. Comparison with experiments is shown for selected probing angles (the remaining angles that show a similar trend as for the angles in the figure, have been omitted for simplicity). The vertical dashed lines correspond to the approximate position of the boundary layer trip disks in the experiments. (a) front wheel and (b) rear wheel.

in streamwise vortical structures, as indicated by the flow patterns seen on the back of both front and rear wheels in Figure 7, whereas the separation mechanisms in the experiments, whose flow regime is certainly not close to the inviscid limit, particularly in this region, causes the flow to separate from the aft part of the outer surface of both front and rear wheels.

3.6. Computational resources

Our simulations were performed with the open source, unified continuum mechanics solver Unicorn [15], a high-performance adaptive solver for large scale unstructured tetrahedral meshes, based on the finite element library DOLFIN [16, 17]. Both Unicorn and DOLFIN run efficiently in parallel on distributed memory machines. The parallelization is based on a fully distributed mesh approach, where everything from preprocessing, assembly, and post-processing is performed in parallel, without representing the entire problem on one core. Mesh adaption is based on the local mesh refinement algorithm, recursive longest edge bisection. The recursive algorithm is applied in parallel on the distributed meshes together with an efficient dynamic load balancer, based on a scratch and remap approach with *a priori* workload estimation and redistribution. For a more detailed description of the parallelization see [18].

In this work, we have used several different computational resources. The initial adaptive iterations were computed on a regular server with 16 GB of memory and two Intel Xeon E5410 quad core processors. When the adaptive iteration started to become too time consuming, we moved the computation to the cluster *Neolith* at NSC/LiU. This cluster consists of 805 compute nodes with 16-32 GB of memory and two Intel Xeon E5345 quad core processors, connected with a high-speed infiniband network. Due to our limited allocation on *Neolith*, the final simulations were run on *Akka* at HPC2N. This cluster consists of 672 nodes, where each node is equipped with 16 GB of memory and two Intel Xeon L5420 quad core processors, with a high-speed infiniband interconnect.

The average turn around time for an adaptive iteration has been around 12 h, using eight cores initially and 256 cores on *Neolith*. The final computation was performed using 264 cores on *Akka*, split up into 4×49 h, checkpointed jobs. The total amount of used resources for the computation, including adaptive iterations and the final run, is estimated to be 71,000 core hours.

4. CONCLUSIONS

We presented detailed results of our contribution to the BANC-I workshop held in Stockholm, 2010. One of our goals with this publication was to introduce the G2 method to the CAA community, showing its competitiveness through the use of an unstructured mesh generated by adaptivity and through results that closely reproduce the trends seen in experiments realized with the same geometry. We were able to compute a mean drag coefficient that compared well with the values predicted by the other groups participating in the BANC-I workshop by using up to six times less cells in our computational mesh. Moreover, because G2 uses unstructured meshes, we believe it is more suitable for the solution of realistic, industrial problems than several other CFD/CAA methods currently available in the literature.

Finally, the results presented here were obtained with a wall layer model that represents the idealized, inviscid flow regime of $Re \rightarrow \infty$. Although no boundary layers were resolved in our simulations, we could still obtain quantitative results that compared well with the experimental values. We expect that this model will perform better when simulating more realistic flows at higher Re (e.g., at actual flight conditions), which is the ultimate goal of CFD.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the financial support from the Swedish Foundation for Strategic Research, the European Research Council, the Swedish Research Council, and the Swedish Energy Agency. This research was conducted using the resources provided by the Swedish National Infrastructure for Computing (SNIC) at the National Supercomputer Centre in Sweden (NSC) and the High Performance Computing Center North (HPC2N).

REFERENCES

1. Spalart PR, Mejia K. Analysis of experimental and numerical studies of the rudimentary landing gear. *Proceedings for 49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, Orlando, Florida, 2011; pp. 6, 8–9.
2. Hoffman J, Jansson N. A computational study of turbulent flow separation for a circular cylinder using skin friction boundary conditions. In *Quality and Reliability of Large-Eddy Simulations II*, Vol. 16, ERCOFTAC Series. Springer Netherlands: Heidelberg, Germany, 2011; 57–68.
3. Hoffman J, Johnson C. *Computational Turbulent Incompressible Flow: Applied Mathematics Body and Soul Vol 4*. Springer-Verlag Publishing: Heidelberg, Germany, 2006.
4. Hoffman J. Computation of mean drag for bluff body problems using Adaptive DNS/LES. *SIAM Journal on Scientific Computing* 2005; **27**(1):184–207.
5. Hoffman J. Adaptive simulation of the turbulent flow past a sphere. *Journal of Fluid Mechanics* 2006; **568**:77–88.
6. Hoffman J, Johnson C. A new approach to computational turbulence modeling. *Computer Methods in Applied Mechanics and Engineering* 2006; **195**:2865–2880.
7. Hoffman J. Efficient computation of mean drag for the subcritical flow past a circular cylinder using General Galerkin G2. *International Journal for Numerical Methods in Fluids* 2009; **59**(11):1241–1258.
8. Eriksson K, Estep D, Hansbo P, Johnson C. Introduction to adaptive methods for differential equations. *Acta Numerica* 1995; **4**:105–158.
9. Hoffman J, Johnson C. Adaptive finite element methods for incompressible fluid flow. In *Error Estimation and Solution Adaptive Discretization in Computational Fluid Dynamics*, Barth TJ, Deconinck H (eds), Lecture Notes in Computational Science and Engineering. Springer-Verlag Publishing: Heidelberg, 2002; 97–158.
10. Becker R, Rannacher R. An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numerica* 2001; **10**:1–102.
11. Hoffman J, Johnson C. Resolution of d'Alembert's Paradox. *Journal of Mathematical Fluid Mechanics* 2010; **12**(3):321–334.
12. Schumann U. Subgrid-scale model for finite difference simulation of turbulent flows in plane channels and annuli. *Journal Of Computational Physics* 1975; **18**:376–404.
13. Spalart P. Rudimentary Landing Gear Workshop, 2010. BANC-1 NASA-AIAA Workshop, problem statement.
14. de Abreu RV, Jansson N, Hoffman J. Adaptive computation of aeroacoustic sources for a rudimentary landing gear using lighthill's analogy. *Proceedings for the 17th AIAA/CEAS Aeroacoustics Conference (32nd AIAA Aeroacoustics Conference)*, Portland, Oregon, 2011.
15. Hoffman J, Jansson J, Vilela de Abreu R, Degirmenci C, Jansson N, Müller K, Nazarov M, Hiromi Spühler J. Unicorn: parallel adaptive finite element simulation of turbulent flow and fluidstructure interaction for deforming domains and complex geometry. *Computer and Fluids* 2013; **80**:310–319.
16. Hoffman J, Logg A. DOLFIN: Dynamic Object oriented Library for FINite element computation, Preprint 2002-06, Chalmers Finite Element Center, Chalmers University of Technology, 2002.
17. Logg A, Wells GN. DOLFIN: Automated finite element computing. *ACM Transactions on Mathematical Software* 2010; **37**(2):1–28.
18. Jansson N. High performance adaptive finite element methods for turbulent fluid flow. *Licentiate Thesis*, Royal Institute of Technology, School of Computer Science and Communication, Stockholm, Sweden, 2011. TRITA-CSC-A 2011:02.