

# Error estimation and anisotropic mesh refinement for 3d laminar aerodynamic flow simulations

Tobias Leicht, Ralf Hartmann \*

*Institute of Aerodynamics and Flow Technology, DLR (German Aerospace Center), Lilienthalplatz 7, 38108 Braunschweig, Germany*  
*Institute of Scientific Computing, TU Braunschweig, Germany*

## ARTICLE INFO

### Article history:

Received 16 July 2009  
 Received in revised form 8 June 2010  
 Accepted 11 June 2010  
 Available online 25 June 2010

### Keywords:

Discontinuous Galerkin discretization  
 3d Compressible Navier–Stokes equations  
 Error estimation  
 Anisotropic mesh refinement  
 Symmetry boundary condition

## ABSTRACT

This article considers *a posteriori* error estimation and anisotropic mesh refinement for three-dimensional laminar aerodynamic flow simulations. The optimal order symmetric interior penalty discontinuous Galerkin discretization which has previously been developed for the compressible Navier–Stokes equations in two dimensions is extended to three dimensions. Symmetry boundary conditions are given which allow to discretize and compute symmetric flows on the half model resulting in exactly the same flow solutions as if computed on the full model. Using duality arguments, an error estimation is derived for estimating the discretization error with respect to the aerodynamic force coefficients. Furthermore, residual-based indicators as well as adjoint-based indicators for goal-oriented refinement are derived. These refinement indicators are combined with anisotropy indicators which are particularly suited to the discontinuous Galerkin (DG) discretization. Two different approaches based on either a heuristic criterion or an anisotropic extension of the adjoint-based error estimation are presented. The performance of the proposed discretization, error estimation and adaptive mesh refinement algorithms is demonstrated for 3d aerodynamic flows.

© 2010 Elsevier Inc. All rights reserved.

## 1. Introduction

In this article, we extend a symmetric interior penalty discontinuous Galerkin (SIPG) method previously developed for the 2d compressible Navier–Stokes equations in [18] to 3d laminar flows, see Sections 2 and 3 for the governing equations and the discretization. This discretization is consistent, adjoint consistent [1,12,13] and of optimal order [18]. In particular, it uses an adjoint consistent discretization of boundary conditions, an optimal order interior penalty term and is combined with an adjoint consistent discretization of the aerodynamic force coefficients.

The discretization is augmented with an accurate symmetry boundary condition. Here, by “accurate” we mean that the symmetry boundary condition allows to discretize and compute symmetric flows on the half (computational) model resulting in exactly the same flow solutions as if computed on the full model. To this end, mirrored state variables and gradients of the state variables are given which allow to devise an accurate symmetry boundary for the considered DG discretization of arbitrary high order.

Important quantities in aerodynamic flow simulations are the aerodynamic force coefficients like the drag, lift and moment coefficients. In addition to the exact approximation of these quantities it is of increasing importance, in particular in the field of uncertainty quantification, to estimate the error in the computed quantities. By employing a duality argument

\* Corresponding author at: Institute of Aerodynamics and Flow Technology, DLR (German Aerospace Center), Lilienthalplatz 7, 38108 Braunschweig, Germany.

E-mail addresses: [Tobias.Leicht@dlr.de](mailto:Tobias.Leicht@dlr.de) (T. Leicht), [Ralf.Hartmann@dlr.de](mailto:Ralf.Hartmann@dlr.de) (R. Hartmann).

error estimates can be derived for estimating the error measured in terms of the 3d aerodynamic force coefficients, see Section 5. The error estimate (Type I error bound) includes local residuals multiplied by the solution to an adjoint problem related to the force coefficient. The error estimate can be decomposed into a sum of local adjoint-based indicators, see Section 7, which can be employed to drive a goal-oriented adaptive mesh refinement algorithm specifically tailored to the accurate and efficient approximation of the aerodynamic force coefficient. We note that the error estimates and adjoint-based indicators derived in this work represent an extension of the error estimates developed for the SIPG discretization of the 2d compressible Navier–Stokes equations in [17] to the optimal order SIPG discretization of the 3d compressible Navier–Stokes equations considered here.

Provided the adjoint solution related to an arbitrary target functional is sufficiently smooth the corresponding error representation can be bounded from above by an error estimate (Type II error bound) which includes the primal residuals introduced in Section 4 but is independent of the adjoint solution. By localizing this error estimate so-called residual-based indicators can be derived, see Section 6. Mesh refinement based on these indicators leads to meshes which resolve *all* flow features irrespective of any specific target quantity. We note that in [17,14] these indicators have been derived for SIPG discretizations of the 2d compressible Navier–Stokes equations. In the current work we derive the residual-based indicators associated with the SIPG discretization of the 3d compressible Navier–Stokes equations, including symmetry boundary conditions.

Flow phenomena may exhibit a strong directional behavior. In particular, in boundary layers or interior layers like shocks the flow variables change rapidly in the direction orthogonal to the layer, whereas the change parallel to the layer is much smaller. Highly stretched elements should be used for an optimal resolution of these features.

Considerable work has been devoted to anisotropic refinement for linear finite elements on simplex meshes where the information of an approximated Hessian-based mesh metric field is used within remeshing algorithms, see for example the work by Formaggia et al. [9], Frey and Alauzet [10], Huang [20] or Sahni et al. [29]. Here, the metric field approximates the interpolation error of the solution and is used to determine the local mesh density as well as the local element rotation and stretching in a remeshing algorithm.

Whereas the use of an interpolation based error estimate is easily applied to any kind of equation it might yield non-optimal refined meshes w.r.t. the efficient approximation of a given target functional. Venditti and Darmofal [32] have combined the directional information of the metric approach with a scaling based on adjoint-based error indicators, resulting in dual weighted metrics. Similar techniques have been applied frequently, e.g. by Jones et al. [21]. Relying on the Hessian to determine the basic metric field this technique is not readily extendible to discretizations of higher than second order. Furthermore, the metric-based approach can easily be combined with different error estimates, see the work of Bourgault et al. [5] based on a *posteriori* error estimates or the work of Loseille et al. [27] based on a combination of the metric approach with an *a priori* goal-oriented error estimate. The latter three references present results for shock-dominated flows governed by the Euler equations. However, as the metric-based approach provides information for a remeshing process it cannot be applied to local element subdivision algorithms, thus most of the work in this field is of little practical use for our specific application.

One common approach to anisotropy detection in the context of element subdivision is to use several trial refinements which include anisotropic cases that split only part of the edges of an element to form child elements with modified aspect ratio. From these discrete choices the case which reduces the error most effectively can be selected, this has been proposed by Sun and Wheeler [30] as well as Kurtz and Demkowicz [25], for example. However, such approaches seem unreasonably expensive, especially if they require solutions on globally refined meshes. Solving only local problems and including goal-oriented refinement has been considered by Houston et al. [19].

In this work we concentrate on anisotropy indicators which come computationally almost for free, i.e. no additional auxiliary problems shall be solved for obtaining anisotropic refinement information. Furthermore, these indicators shall be applicable to higher order DG discretizations. In [26] anisotropic indicators have been developed for the anisotropic mesh refinement of 2d quadrilateral meshes. Being based on inter-element jumps these indicators are particularly suited for discontinuous Galerkin flow solutions. In Section 8, we extend the anisotropic indicators to be used in an anisotropic three-dimensional mesh refinement algorithm. In particular, the anisotropic indicators are combined with the residual-based as well as the adjoint-based indicators. Furthermore, in addition to this heuristic technique an anisotropic adjoint-based error estimate based on the recent work of Richter [28] is presented. This estimate decomposes the element-wise isotropic adjoint-based error estimator into directional contributions which accumulate to the isotropic one.

The main goal of this work is to extend the discretization methods, error estimates and adaptive isotropic and anisotropic mesh refinement algorithms, which have been developed for 2d laminar flows over the last several years [13,14,17,26] to three dimensions and to demonstrate that they can be successfully applied to 3d aerodynamic flow simulations. To this end, in Section 9 we apply the proposed algorithms to aerodynamic test cases which have also been considered in the European project ADIGMA [24]. In particular, we consider two laminar three-dimensional aerodynamic flows of different complexity: (a) a laminar flow around a curved and streamlined body and (b) a laminar flow around a delta wing.

The authors are aware of only one work on goal-oriented (adjoint-based) mesh refinement applied to discontinuous Galerkin discretizations of 3d aerodynamic flows. In the publications [7,8] which both are based on Fidkowski [6] the adjoint-based mesh refinement uses a cut-cell approach and is applied to the 3d compressible Euler equations with isotropic refinement. In contrast to that the current work is based on boundary-aligned meshes, provides *a posteriori* error estimates in the computed force coefficients, derives residual-based and adjoint-based mesh refinement indicators and combines them with anisotropic mesh refinement for the 3d compressible Navier–Stokes equations.

## 2. The 3d compressible Navier–Stokes equations

We consider the three-dimensional steady-state compressible Navier–Stokes equations

$$\nabla \cdot (\mathcal{F}^c(\mathbf{u}) - \mathcal{F}^v(\mathbf{u}, \nabla \mathbf{u})) = 0 \quad \text{in } \Omega \quad (1)$$

subject to various boundary conditions including no-slip wall boundary conditions with vanishing velocity  $\mathbf{v} = (v_1, v_2, v_3)^\top = 0$  at isothermal walls  $\Gamma_{\text{iso}}$  where  $T = T_{\text{wall}}$ , or at adiabatic walls  $\Gamma_{\text{adia}}$  where  $\mathbf{n} \cdot \nabla T = 0$ . The vector of conservative variables  $\mathbf{u}$  is given by  $\mathbf{u} = (\rho, \rho v_1, \rho v_2, \rho v_3, \rho E)^\top$  and the convective fluxes  $\mathcal{F}^c(\mathbf{u}) = (\mathbf{f}_1^c(\mathbf{u}), \mathbf{f}_2^c(\mathbf{u}), \mathbf{f}_3^c(\mathbf{u}))^\top$  by

$$\mathbf{f}_1^c(\mathbf{u}) = \begin{bmatrix} \rho v_1 \\ \rho v_1^2 + p \\ \rho v_1 v_2 \\ \rho v_1 v_3 \\ \rho H v_1 \end{bmatrix}, \quad \mathbf{f}_2^c(\mathbf{u}) = \begin{bmatrix} \rho v_2 \\ \rho v_2 v_1 \\ \rho v_2^2 + p \\ \rho v_2 v_3 \\ \rho H v_2 \end{bmatrix}, \quad \text{and} \quad \mathbf{f}_3^c(\mathbf{u}) = \begin{bmatrix} \rho v_3 \\ \rho v_3 v_1 \\ \rho v_3 v_2 \\ \rho v_3^2 + p \\ \rho H v_3 \end{bmatrix},$$

where  $\rho$ ,  $p$  and  $E$  denote the density, pressure and specific total energy, respectively. Additionally,  $H$  is the total enthalpy given by

$$H = E + \frac{p}{\rho} = e + \frac{1}{2} \mathbf{v}^2 + \frac{p}{\rho}, \quad (2)$$

where  $e$  is the specific static internal energy, and the pressure is determined by the equation of state of an ideal gas

$$p = (\gamma - 1) \rho e, \quad (3)$$

where  $\gamma = c_p/c_v$  is the ratio of specific heat capacities at constant pressure,  $c_p$ , and constant volume,  $c_v$ . Furthermore, the viscous fluxes  $\mathcal{F}^v(\mathbf{u}, \nabla \mathbf{u}) = (\mathbf{f}_1^v(\mathbf{u}, \nabla \mathbf{u}), \mathbf{f}_2^v(\mathbf{u}, \nabla \mathbf{u}), \mathbf{f}_3^v(\mathbf{u}, \nabla \mathbf{u}))^\top$  are defined by

$$\mathbf{f}_i^v(\mathbf{u}, \nabla \mathbf{u}) = \begin{pmatrix} 0 \\ \tau_{1i} \\ \tau_{2i} \\ \tau_{3i} \\ \tau_{ij} v_j + \kappa T_{x_i} \end{pmatrix}, \quad i = 1, 2, 3.$$

Here  $T$  denotes the temperature given by  $e = c_v T$ ,  $\kappa$  is the thermal conductivity coefficient and  $\underline{\tau}$  is the viscous stress tensor defined by

$$\underline{\tau} = \mu \left( \nabla \mathbf{v} + (\nabla \mathbf{v})^\top - \frac{2}{3} (\nabla \cdot \mathbf{v}) \mathbf{I} \right), \quad (4)$$

where  $\mu$  is the dynamic viscosity coefficient. For the purposes of discretization, we rewrite the compressible Navier–Stokes equations (1) in the following (equivalent) form:

$$\frac{\partial}{\partial x_i} \left( \mathbf{f}_i^c(\mathbf{u}) - G_{ij}(\mathbf{u}) \frac{\partial \mathbf{u}}{\partial x_j} \right) = 0 \quad \text{in } \Omega, \quad (5)$$

where  $G_{ij}(\mathbf{u}) = \partial \mathbf{f}_i^v(\mathbf{u}, \nabla \mathbf{u}) / \partial u_{x_j}$ , for  $i, j = 1, 2, 3$ , are the homogeneity matrices with  $\mathbf{f}_i^v(\mathbf{u}, \nabla \mathbf{u}) = G_{ij}(\mathbf{u}) \partial \mathbf{u} / \partial x_j$ ,  $i = 1, 2, 3$ .

## 3. The SIPG discretization of the 3d compressible Navier–Stokes equations

In this section we extend the consistent and adjoint-consistent interior penalty discontinuous Galerkin discretization as derived for the two-dimensional compressible Navier–Stokes equations in [13,18] to three dimensions.

First, we begin by introducing some notation. We assume that  $\Omega$  can be subdivided into shape-regular meshes  $\mathcal{T}_h = \{\kappa\}$  consisting of hexahedral elements  $\kappa$ . Here,  $h$  denotes the piecewise constant mesh function defined by  $h_{-\kappa} \equiv h_\kappa = \text{diam}(\kappa)$  for all  $\kappa \in \mathcal{T}_h$ . Let us assume that each  $\kappa \in \mathcal{T}_h$  is an image of a fixed reference element  $\hat{\kappa}$ , that is,  $\kappa = \sigma_\kappa(\hat{\kappa})$  for all  $\kappa \in \mathcal{T}_h$ , where  $\hat{\kappa}$  is the open unit cube in  $\mathbb{R}^3$ . Furthermore the mapping  $\sigma_\kappa$  of the reference element  $\hat{\kappa}$  to the element  $\kappa$  in real space is assumed to be bijective and smooth, with the eigenvalues of its Jacobian matrix being bounded from below and above. In the simplest case this mapping is tri-linear. In order to achieve a good approximation of strongly curved geometries on coarse meshes mappings can be based on polynomials of higher degree instead of linear functions, see [23] for more details about curved elements. On the reference element  $\hat{\kappa}$  we define spaces of tensor product polynomials of degree  $p \geq 0$  as follows:

$$\mathcal{Q}_p(\hat{\kappa}) = \text{span}\{\hat{\mathbf{x}}^\alpha : 0 \leq \alpha_i \leq p, \quad i = 1, 2, 3\}, \quad (6)$$

where  $\alpha$  denotes a multi-index and  $\hat{\mathbf{x}}^\alpha = \prod_{i=1}^3 \hat{x}_i^{\alpha_i}$ . Finally, we introduce the finite element space  $\mathbf{V}_h^p$  consisting of discontinuous vector-valued tensor product polynomial functions of degree  $p \geq 0$ , defined by

$$\mathbf{V}_h^p = \{\mathbf{v}_h \in [L_2(\Omega)]^5 : \mathbf{v}_h|_\kappa \circ \sigma_\kappa \in [Q_p(\hat{\kappa})]^5, \kappa \in \mathcal{T}_h\}. \tag{7}$$

Suppose that  $\mathbf{v}|_\kappa \in [H^1(\kappa)]^m$ ,  $m \geq 1$ , for each  $\kappa \in \mathcal{T}_h$ . We now define average and jump operators for vector- and matrix-valued functions. To this end, let  $\kappa^+$  and  $\kappa^-$  be two adjacent elements of  $\mathcal{T}_h$  and  $\mathbf{x}$  be an arbitrary point on the interior face  $f = \partial\kappa^+ \cap \partial\kappa^- \subset \Gamma_I$ . Moreover, let  $\mathbf{v}$  and  $\underline{\tau}$  be vector- and matrix-valued functions, respectively, that are smooth inside each element  $\kappa^\pm$ . By  $\mathbf{v}^\pm := \mathbf{v}|_{\partial\kappa^\pm}$  and  $\underline{\tau}^\pm := \underline{\tau}|_{\partial\kappa^\pm}$  we denote the traces of, respectively,  $\mathbf{v}$  and  $\underline{\tau}$  on  $f$  taken from within the interior of  $\kappa^\pm$ , respectively. Then, we define the averages at  $\mathbf{x} \in f$  by  $\{\mathbf{v}\} = (\mathbf{v}^+ + \mathbf{v}^-)/2$  and  $\{\underline{\tau}\} = (\underline{\tau}^+ + \underline{\tau}^-)/2$ . Similarly, the jump at  $\mathbf{x} \in f$  is given by  $[\underline{\mathbf{v}}] = \mathbf{v}^+ \otimes \mathbf{n}_{\kappa^+} + \mathbf{v}^- \otimes \mathbf{n}_{\kappa^-}$ . On a boundary face  $f \subset \Gamma$ , we set  $\{\mathbf{v}\} = \mathbf{v}$ ,  $\{\underline{\tau}\} = \underline{\tau}$  and  $[\underline{\mathbf{v}}] = \mathbf{v} \otimes \mathbf{n}$ . For matrices  $\underline{\sigma}, \underline{\tau} \in \mathbb{R}^{m \times n}$ ,  $m, n \geq 1$ , we use the standard notation  $\underline{\sigma} : \underline{\tau} = \sum_{k=1}^m \sum_{l=1}^n \sigma_{kl} \tau_{kl}$ ; additionally, for vectors  $\mathbf{v} \in \mathbb{R}^m$ ,  $\mathbf{w} \in \mathbb{R}^n$ , the matrix  $\mathbf{v} \otimes \mathbf{w} \in \mathbb{R}^{m \times n}$  is defined by  $(\mathbf{v} \otimes \mathbf{w})_{kl} = v_k w_l$ .

The discontinuous Galerkin discretization of the 3d compressible Navier–Stokes equations (1) is given by: Find  $\mathbf{u}_h \in \mathbf{V}_h^p$  such that

$$\begin{aligned} \mathcal{N}(\mathbf{u}_h, \mathbf{v}) \equiv & \int_\Omega \mathcal{F}^c(\mathbf{u}_h) : \nabla_h \mathbf{v} \, d\mathbf{x} + \sum_{\kappa \in \mathcal{T}_h} \int_{\partial\kappa \setminus \Gamma} \mathcal{H}(\mathbf{u}_h^+, \mathbf{u}_h^-, \mathbf{n}^+) \cdot \mathbf{v}^+ \, ds + \int_\Omega \mathcal{F}^v(\mathbf{u}_h, \nabla_h \mathbf{u}_h) : \nabla_h \mathbf{v} \, d\mathbf{x} \\ & - \int_{\Gamma_I} \{\mathcal{F}^v(\mathbf{u}_h, \nabla_h \mathbf{u}_h)\} : [\underline{\mathbf{v}}] \, ds - \int_{\Gamma_I} \{G^\top(\mathbf{u}_h) \nabla_h \mathbf{v}\} : [\underline{\mathbf{u}}_h] \, ds + \int_{\Gamma_I} \underline{\delta}(\mathbf{u}_h) : [\underline{\mathbf{v}}] \, ds \\ & + \mathcal{N}_\Gamma(\mathbf{u}_h, \mathbf{v}) + \mathcal{N}_{\Gamma_{\text{sym}}}(\mathbf{u}_h, \mathbf{v}) = 0 \end{aligned} \tag{8}$$

for all  $\mathbf{v}$  in  $\mathbf{V}_h^p$ . Here, the numerical flux  $\mathcal{H}(\cdot, \cdot, \cdot)$  may be chosen to be any two-point monotone Lipschitz function which is consistent, i.e.  $\mathcal{H}(\mathbf{v}, \mathbf{v}, \mathbf{n}) = \mathcal{F}^c(\mathbf{v}) \cdot \mathbf{n}$ , and conservative, i.e.  $\mathcal{H}(\mathbf{v}, \mathbf{w}, \mathbf{n}) = -\mathcal{H}(\mathbf{w}, \mathbf{v}, -\mathbf{n})$ . The penalization term is given by

$$\underline{\delta}(\mathbf{u}_h) = \underline{\delta}^{\text{ip}}(\mathbf{u}_h) = C_{\text{IP}} \frac{p^2}{h_f} \{G(\mathbf{u}_h)\} [\underline{\mathbf{u}}_h], \tag{9}$$

where  $h_f$  represents the element dimension orthogonal to the face  $f$  of the elements  $\kappa^+$  and  $\kappa^-$  adjacent to  $f$ . Furthermore,  $C_{\text{IP}}$  is a positive constant, which, for reasons of stability, must be chosen sufficiently large. Finally, the boundary terms included in  $\mathcal{N}_\Gamma(\mathbf{u}_h, \mathbf{v})$  are given by

$$\begin{aligned} \mathcal{N}_\Gamma(\mathbf{u}_h, \mathbf{v}) = & \int_\Gamma \mathcal{H}_\Gamma(\mathbf{u}_h^+, \mathbf{u}_\Gamma(\mathbf{u}_h^+), \mathbf{n}^+) \cdot \mathbf{v}^+ \, ds + \int_\Gamma \underline{\delta}_\Gamma(\mathbf{u}_h^+) : \mathbf{v} \otimes \mathbf{n} \, ds \\ & - \int_\Gamma \mathbf{n} \cdot \mathcal{F}_\Gamma^v(\mathbf{u}_h^+, \nabla_h \mathbf{u}_h^+) \mathbf{v}^+ \, ds - \int_\Gamma (G_\Gamma^\top(\mathbf{u}_h^+) \nabla_h \mathbf{v}_h^+) : (\mathbf{u}_h^+ - \mathbf{u}_\Gamma(\mathbf{u}_h^+)) \otimes \mathbf{n} \, ds, \end{aligned} \tag{10}$$

where the penalization term at the boundary is given by

$$\underline{\delta}_\Gamma(\mathbf{u}_h) = \underline{\delta}_\Gamma^{\text{ip}}(\mathbf{u}_h) = C_{\text{IP}} \frac{p^2}{h_f} G_\Gamma(\mathbf{u}_h)(\mathbf{u}_h - \mathbf{u}_\Gamma(\mathbf{u}_h)) \otimes \mathbf{n}. \tag{11}$$

Here, the viscous boundary flux  $\mathcal{F}_\Gamma^v$  and the corresponding homogeneity tensor  $G_\Gamma$  are defined by

$$\mathcal{F}_\Gamma^v(\mathbf{u}_h, \nabla_h \mathbf{u}_h) = \mathcal{F}^v(\mathbf{u}_\Gamma(\mathbf{u}_h), \nabla_h \mathbf{u}_h) = G_\Gamma(\mathbf{u}_h) \nabla_h \mathbf{u}_h = G(\mathbf{u}_\Gamma(\mathbf{u}_h)) \nabla_h \mathbf{u}_h. \tag{12}$$

Furthermore, on adiabatic boundaries  $\Gamma_{\text{adia}} \subset \Gamma_w$ ,  $\mathcal{F}_\Gamma^v$  and  $G_\Gamma$  are modified such that  $\mathbf{n} \cdot \nabla T = 0$ . Finally, we define

$$\mathcal{H}_\Gamma(\mathbf{u}_h^+, \mathbf{u}_\Gamma(\mathbf{u}_h^+), \mathbf{n}) = \mathbf{n} \cdot \mathcal{F}_\Gamma^c(\mathbf{u}_h^+) = \mathbf{n} \cdot \mathcal{F}^c(\mathbf{u}_\Gamma(\mathbf{u}_h^+)), \tag{13}$$

where the boundary function  $\mathbf{u}_\Gamma(\cdot)$  is given by  $\mathbf{u}_\Gamma(\mathbf{w}) = (w_1, 0, 0, 0, w_5)^\top$  on  $\Gamma_{\text{adia}}$ , and by  $\mathbf{u}_\Gamma(\mathbf{w}) = (w_1, 0, 0, 0, w_1 c_v T_{\text{wall}})^\top$  on  $\Gamma_{\text{iso}}$ , see [18] for the treatment of other boundary conditions. Finally, we note that the boundary function  $\mathbf{u}_\Gamma(\cdot)$  is consistent, i.e. on all boundary parts,  $\mathbf{u}_\Gamma(\cdot)$  is chosen such that the exact solution  $\mathbf{u}$  to (1) satisfies  $\mathbf{u}_\Gamma(\mathbf{u}) = \mathbf{u}$ . As a consequence also  $\underline{\delta}_\Gamma(\cdot)$  as defined in (11) is consistent. In fact, the exact solution  $\mathbf{u}$  to (1) satisfies  $\underline{\delta}_\Gamma(\mathbf{u}) = 0$ .

A substantial number of 3d aerodynamic flows can be treated as symmetric if the side slip angle vanishes. Symmetry boundary conditions should be defined such that the discretization on the half domain resembles the discretization on the full domain. This can be achieved if the boundary conditions are derived considering the discrete problem, not the continuous one.

To this end, we replace  $\mathbf{u}_h^-$  on the symmetry boundary  $\Gamma_{\text{sym}}$  by the boundary function  $\mathbf{u}_\Gamma(\mathbf{u}_h^+)$  defined by

$$\mathbf{u}_\Gamma(\mathbf{u}) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 - 2n_1^2 & -2n_1 n_2 & -2n_1 n_3 & 0 \\ 0 & -2n_1 n_2 & 1 - 2n_2^2 & -2n_2 n_3 & 0 \\ 0 & -2n_1 n_3 & -2n_2 n_3 & 1 - 2n_3^2 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \mathbf{u} \quad \text{on } \Gamma_{\text{sym}}. \tag{14}$$

Here,  $\mathbf{u}_\Gamma(\mathbf{u})$  is chosen in a way to ensure that scalar physical quantities are symmetric in a classical sense, i.e.  $\rho_\Gamma = \rho^+$  and  $(\rho E)_\Gamma = (\rho E)^+$ , whereas vector-valued physical quantities like the velocity are symmetrical in a vectorial sense, i.e.

$\mathbf{v}_r \cdot \mathbf{t} = \mathbf{v}^+ \cdot \mathbf{t}$  for  $\mathbf{t} \cdot \mathbf{n} = 0$ , and  $\mathbf{v}_r \cdot \mathbf{n} = -\mathbf{v}^+ \cdot \mathbf{n}$ , where  $\mathbf{n} = (n_1, n_2, n_3)^\top$  is the unit outward normal vector on  $\Gamma_{\text{sym}}$ , i.e. the normal component of vector-valued quantities is antisymmetric if measured with the same normal vector  $\mathbf{n}$ .

In order to obtain the gradient  $\nabla \mathbf{u}_h^-$  we have to take into account the linear transformation of the state variables (14) as well as the fact that the gradient of a scalar quantity is a vector-valued quantity that has to be treated just like the velocity vector. Combining these two ingredients we arrive at the following expression for  $(\nabla \mathbf{u})_r(\mathbf{u}_h^+)$  which replaces  $\nabla \mathbf{u}_h^-$ :

$$(\nabla \mathbf{u})_{r,jl}(\mathbf{u}_h) = \partial_{u_m} \mathbf{u}_r^j(\mathbf{u}_h) \partial_{x_n} \mathbf{u}_h^m (\delta_{nl} - 2n_n n_l). \quad (15)$$

This gradient could also be computed by evaluating the local derivative on a ghost cell with state variables created through the symmetry condition above. However, using (15) we obtain the same result without the necessity to actually construct a ghost cell.

The discretization on the symmetry boundary is given by

$$\begin{aligned} \mathcal{N}_{\Gamma_{\text{sym}}}(\mathbf{u}_h, \mathbf{v}) &= \int_{\Gamma_{\text{sym}}} \mathcal{H}(\mathbf{u}_h^+, \mathbf{u}_r(\mathbf{u}_h^+), \mathbf{n}^+) \cdot \mathbf{v}^+ \, ds - \int_{\Gamma_{\text{sym}}} \frac{1}{2} (\mathcal{F}^\nu(\mathbf{u}_h^+, \nabla_h \mathbf{u}_h^+) + \mathcal{F}^\nu(\mathbf{u}_r(\mathbf{u}_h^+), (\nabla \mathbf{u})_r(\mathbf{u}_h^+))) : \mathbf{v}^+ \otimes \mathbf{n} \, ds \\ &\quad - \int_{\Gamma_{\text{sym}}} \frac{1}{2} (G^\top(\mathbf{u}_h^+) \nabla_h \mathbf{v}^+) : (\mathbf{u}_h^+ - \mathbf{u}_r(\mathbf{u}_h^+)) \otimes \mathbf{n} \, ds + \int_{\Gamma_{\text{sym}}} \underline{\delta}_{\Gamma_{\text{sym}}}(\mathbf{u}_h) : \mathbf{v}^+ \otimes \mathbf{n} \, ds, \end{aligned}$$

where the penalization term is given by

$$\underline{\delta}_{\Gamma_{\text{sym}}}(\mathbf{u}_h) = \underline{\delta}_{\Gamma_{\text{sym}}}^{\text{ip}}(\mathbf{u}_h) = C_{\text{ip}} \frac{p^2}{h_f} \frac{1}{2} (G(\mathbf{u}_h^+) + G(\mathbf{u}_r(\mathbf{u}_h^+)))(\mathbf{u}_h - \mathbf{u}_r(\mathbf{u}_h)) \otimes \mathbf{n}. \quad (16)$$

#### 4. The primal residual form

Using integration by parts in (8) we obtain the primal residual form (see Hartmann and Houston [18]) given by: find  $\mathbf{u}_h \in \mathbf{V}_h^p$  such that

$$\begin{aligned} \mathcal{R}(\mathbf{u}_h, \mathbf{v}) &\equiv \int_{\Omega} \mathbf{R}(\mathbf{u}_h) \cdot \mathbf{v} \, dx + \sum_{\kappa \in \mathcal{T}_h} \int_{\partial \kappa \setminus \Gamma} \mathbf{r}(\mathbf{u}_h) \cdot \mathbf{v}^+ + \underline{\rho}(\mathbf{u}_h) : \nabla \mathbf{v}^+ \, ds + \int_{\Gamma} \mathbf{r}_r(\mathbf{u}_h) \cdot \mathbf{v}^+ + \underline{\rho}_r(\mathbf{u}_h) : \nabla \mathbf{v}^+ \, ds \\ &\quad + \int_{\Gamma_{\text{sym}}} \mathbf{r}_{\Gamma_{\text{sym}}}(\mathbf{u}_h) \cdot \mathbf{v}^+ + \underline{\rho}_{\Gamma_{\text{sym}}}(\mathbf{u}_h) : \nabla \mathbf{v}^+ \, ds = 0 \quad \forall \mathbf{v} \in \mathbf{V}_h^p, \end{aligned} \quad (17)$$

where the primal element, interior face and boundary residuals are given by

$$\begin{aligned} \mathbf{R}(\mathbf{u}_h) &= -\nabla \cdot \mathcal{F}^c(\mathbf{u}_h) + \nabla \cdot \mathcal{F}^\nu(\mathbf{u}_h, \nabla_h \mathbf{u}_h) \quad \text{in } \kappa, \kappa \in \mathcal{T}_h, \\ \mathbf{r}(\mathbf{u}_h) &= \mathbf{n} \cdot \mathcal{F}^c(\mathbf{u}_h^+) - \mathcal{H}(\mathbf{u}_h^+, \mathbf{u}_r^-, \mathbf{n}^+) - \frac{1}{2} \llbracket \mathcal{F}^\nu(\mathbf{u}_h, \nabla_h \mathbf{u}_h) \rrbracket - \mathbf{n} \cdot \underline{\delta}(\mathbf{u}_h), \\ \underline{\rho}(\mathbf{u}_h) &= \frac{1}{2} (G(\mathbf{u}_h) \llbracket \mathbf{u}_h \rrbracket)^\top \quad \text{on } \partial \kappa \setminus \Gamma, \kappa \in \mathcal{T}_h, \\ \mathbf{r}_r(\mathbf{u}_h) &= \mathbf{n} \cdot (\mathcal{F}^c(\mathbf{u}_h^+) - \mathcal{F}_r^c(\mathbf{u}_h^+) - \mathcal{F}^\nu(\mathbf{u}_h^+, \nabla_h \mathbf{u}_h^+) + \mathcal{F}_r^\nu(\mathbf{u}_h^+, \nabla_h \mathbf{u}_h^+)) - \mathbf{n} \cdot \underline{\delta}_r(\mathbf{u}_h), \\ \underline{\rho}_r(\mathbf{u}_h) &= (G_r^\top(\mathbf{u}_h^+) : (\mathbf{u}_h^+ - \mathbf{u}_r(\mathbf{u}_h^+)) \otimes \mathbf{n})^\top \quad \text{on } \Gamma, \\ \mathbf{r}_{\Gamma_{\text{sym}}}(\mathbf{u}_h) &= \mathbf{n} \cdot \mathcal{F}^c(\mathbf{u}_h^+) - \mathcal{H}(\mathbf{u}_h^+, \mathbf{u}_r(\mathbf{u}_h^+), \mathbf{n}^+) - \frac{1}{2} \mathbf{n} \cdot (\mathcal{F}^\nu(\mathbf{u}_h^+, \nabla_h \mathbf{u}_h^+) - \mathcal{F}^\nu(\mathbf{u}_h^+, (\nabla \mathbf{u})_r(\mathbf{u}_h^+))) - \mathbf{n} \cdot \underline{\delta}_{\Gamma_{\text{sym}}}(\mathbf{u}_h), \\ \underline{\rho}_{\Gamma_{\text{sym}}}(\mathbf{u}_h) &= \frac{1}{2} (G^\top(\mathbf{u}_h^+) : (\mathbf{u}_h^+ - \mathbf{u}_r(\mathbf{u}_h^+)) \otimes \mathbf{n})^\top \quad \text{on } \Gamma_{\text{sym}}. \end{aligned} \quad (18)$$

As shown in [18] the exact solution  $\mathbf{u}$  to (1) satisfies

$$\mathbf{R}(\mathbf{u}) = 0, \quad \mathbf{r}(\mathbf{u}) = 0, \quad \underline{\rho}(\mathbf{u}) = 0, \quad \mathbf{r}_r(\mathbf{u}) = 0, \quad \underline{\rho}_r(\mathbf{u}) = 0.$$

Furthermore, the exact solution  $\mathbf{u}$  to (1) satisfies

$$\mathbf{r}_{\Gamma_{\text{sym}}}(\mathbf{u}) = 0, \quad \underline{\rho}_{\Gamma_{\text{sym}}}(\mathbf{u}) = 0.$$

Thereby, the discretization (8) is consistent, i.e. the exact solution  $\mathbf{u} \in \mathbf{V}$  satisfies the following equation:

$$\mathcal{N}(\mathbf{u}, \mathbf{v}) = 0 \quad \forall \mathbf{v} \in \mathbf{V}, \quad (19)$$

where  $\mathbf{V}$  is some suitably chosen function space including the exact solution  $\mathbf{u} \in \mathbf{V}$  to the primal problem (1) and satisfying  $\mathbf{V}_h \subset \mathbf{V}$ , see [1,13] for the choice of  $\mathbf{V}$  in the case of discontinuous Galerkin methods.

## 5. A posteriori error estimation

We are interested in estimating the error in following aerodynamic force coefficients: the total drag and lift coefficients which are given by

$$\frac{1}{q_\infty \bar{A}} \int_{\Gamma_w} (\mathbf{p}\mathbf{n} - \underline{\boldsymbol{\tau}}\mathbf{n}) \cdot \boldsymbol{\psi} \, ds. \quad (20)$$

Here,  $q_\infty = \frac{1}{2} \gamma p_\infty M_\infty^2 = \frac{1}{2} \gamma \frac{|\mathbf{v}_\infty|^2}{c_\infty^2} p_\infty = \frac{1}{2} \rho_\infty |\mathbf{v}_\infty|^2$ , where  $M$  denotes the Mach number,  $c$  the speed of sound defined by  $c^2 = \gamma p/\rho$  and  $\bar{A}$  denotes a reference area. The subscripts  $\infty$  indicate free-stream quantities. Finally, in case of vanishing sideslip and roll angles  $\beta$  and  $\gamma$ ,  $\boldsymbol{\psi}$  is given by

$$\boldsymbol{\psi}_d = (\cos(\alpha), 0, \sin(\alpha))^\top \quad \text{or} \quad \boldsymbol{\psi}_l = (-\sin(\alpha), 0, \cos(\alpha))^\top$$

for the drag and lift coefficient, respectively, where  $\alpha$  is the angle of attack. The side force coefficient vanishes for symmetric bodies with zero sideslip angle. According to the analysis in [13], see also [18], we modify the force coefficients in (20) as follows

$$J(\mathbf{u}_h) = \frac{1}{q_\infty \bar{A}} \int_{\Gamma_w} (p_I \mathbf{n} - \underline{\boldsymbol{\tau}}_I \mathbf{n}) \cdot \boldsymbol{\psi} \, ds + \int_{\Gamma_w} \underline{\boldsymbol{\delta}}_I(\mathbf{u}_h) : \mathbf{z}_I \otimes \mathbf{n} \, ds. \quad (21)$$

Eq. (21) is a consistent modification of the force coefficient in (20) as  $\underline{\boldsymbol{\delta}}_I(\mathbf{u}) = \mathbf{0}$  holds for the analytical solution  $\mathbf{u}$ .

Moreover, this modification ensures that the discretization in (8) in combination with the target functional (21) is adjoint consistent [13,18].

Given the discretization (8) and the target functional (21) the derivation of error estimates for  $J(\cdot)$  follows the general approach of duality-based *a posteriori* error estimates for target functionals, see e.g. [4,11,16] among many others. Following this approach but omitting details for brevity we arrive at following error representation

$$J(\mathbf{u}) - J(\mathbf{u}_h) = -\mathcal{N}(\mathbf{u}_h, \mathbf{z}) = \mathcal{R}(\mathbf{u}_h, \mathbf{z}), \quad (22)$$

where  $\mathbf{z}$  is the exact but in general unknown solution to an adjoint problem connected to the target functional  $J(\cdot)$ . Replacing  $\mathbf{z}$  by an approximate solution  $\tilde{\mathbf{z}}_h$  to a linearized adjoint problem gives rise to following approximate error representation

$$J(\mathbf{u}) - J(\mathbf{u}_h) \approx -\mathcal{R}(\mathbf{u}_h, \tilde{\mathbf{z}}_h), \quad (23)$$

where  $\tilde{\mathbf{z}}_h \in \tilde{\mathbf{V}}_h$  is the solution to following discrete adjoint problem, see e.g. [17] for more details: Find  $\tilde{\mathbf{z}}_h \in \tilde{\mathbf{V}}_h$  such that

$$\mathcal{N}'[\mathbf{u}_h](\mathbf{w}_h, \tilde{\mathbf{z}}_h) = J'[\mathbf{u}_h](\mathbf{w}_h) \quad \forall \mathbf{w}_h \in \tilde{\mathbf{V}}_h, \quad (24)$$

which is usually computed on the same mesh  $\mathcal{T}_h$  used for  $\mathbf{u}_h$ , but with a higher degree polynomial. We note, that due to replacing the exact adjoint solution  $\mathbf{z}$  in (22) by the numerical approximation  $\tilde{\mathbf{z}}_h$  the resulting formula (23) represents an approximation only of the true error. However, in a series of publications, e.g. [16,17,32] among others, it has been demonstrated that this approximation is close to the true error in the target functional. We will demonstrate in Section 9 that this holds true also for the 3d aerodynamic flows considered in this work.

## 6. Residual-based indicators

Let  $\mathbf{u}$  and  $\mathbf{u}_h$  denote the solutions to (1) and (8), respectively. Recalling from (8) that  $\mathcal{N}(\mathbf{u}_h, \mathbf{v}_h) = \mathbf{0}$  holds for any discrete function  $\mathbf{v}_h \in \mathbf{V}_h^p$  the error representation in (22) can be rewritten as follows:

$$J(\mathbf{u}) - J(\mathbf{u}_h) = -\mathcal{N}(\mathbf{u}_h, \mathbf{z} - \mathbf{v}_h) = \mathcal{R}(\mathbf{u}_h, \mathbf{z} - \mathbf{v}_h), \quad (25)$$

for any  $\mathbf{v}_h \in \mathbf{V}_h^p$ . In particular, we can choose  $\mathbf{v}_h := P_h \mathbf{z} \in \mathbf{V}_h^p$  in (25), i.e.

$$J(\mathbf{u}) - J(\mathbf{u}_h) = \mathcal{R}(\mathbf{u}_h, \mathbf{z} - P_h \mathbf{z}), \quad (26)$$

where  $P_h \mathbf{z}$  denotes an appropriate interpolation/projection of  $\mathbf{z}$  into the discrete function space  $\mathbf{V}_h^p$  with following properties, see e.g. [2]: Suppose  $\mathbf{z} - \kappa$  in  $[H^{s_\kappa+1}(\kappa)]^5$ ,  $s_\kappa \geq 0$ , for  $\kappa \in \mathcal{T}_h$ . Then

$$\|\mathbf{z} - P_h \mathbf{z}\|_{H^m(\kappa)} \leq Ch_\kappa^{t_\kappa+1-m} |\mathbf{z}|_{H^{t_\kappa+1}(\kappa)}, \quad (27)$$

where  $t_\kappa = \min(s_\kappa, p)$ ,  $\kappa \in \mathcal{T}_h$ . Then, by the trace theorem, we have

$$\begin{aligned} \|\mathbf{z} - P_h \mathbf{z}\|_{L^2(\partial\kappa)} &\leq Ch_\kappa^{t_\kappa+1/2} |\mathbf{z}|_{H^{t_\kappa+1}(\kappa)}, \\ \|\mathbf{z} - P_h \mathbf{z}\|_{H^1(\partial\kappa)} &\leq Ch_\kappa^{t_\kappa-1/2} |\mathbf{z}|_{H^{t_\kappa+1}(\kappa)}. \end{aligned} \quad (28)$$

Using (17) we rewrite (26) as follows

$$\begin{aligned}
J(\mathbf{u}) - J(\mathbf{u}_h) &= \int_{\Omega} \mathbf{R}(\mathbf{u}_h) \cdot (\mathbf{z} - P_h \mathbf{z}) \, d\mathbf{x} + \sum_{\kappa \in \mathcal{T}_h} \int_{\partial\kappa \setminus \Gamma} \mathbf{r}(\mathbf{u}_h) \cdot (\mathbf{z} - P_h \mathbf{z})^+ + \underline{\rho}(\mathbf{u}_h) : \nabla(\mathbf{z} - P_h \mathbf{z})^+ \, ds \\
&\quad + \int_{\Gamma} \mathbf{r}_\Gamma(\mathbf{u}_h) \cdot (\mathbf{z} - P_h \mathbf{z})^+ + \underline{\rho}_\Gamma(\mathbf{u}_h) : \nabla(\mathbf{z} - P_h \mathbf{z})^+ \, ds \\
&\quad + \int_{\Gamma_{\text{sym}}} \mathbf{r}_{\Gamma_{\text{sym}}}(\mathbf{u}_h) \cdot (\mathbf{z} - P_h \mathbf{z})^+ + \underline{\rho}_{\Gamma_{\text{sym}}}(\mathbf{u}_h) : \nabla(\mathbf{z} - P_h \mathbf{z})^+ \, ds,
\end{aligned} \tag{29}$$

where the primal element residuals  $\mathbf{R}(\mathbf{u}_h)$ , the interior face residuals  $\mathbf{r}(\mathbf{u}_h)$  and  $\underline{\rho}(\mathbf{u}_h)$ , the boundary residuals  $\mathbf{r}_\Gamma(\mathbf{u}_h)$  and  $\underline{\rho}_\Gamma(\mathbf{u}_h)$ , and the symmetry boundary residuals  $\mathbf{r}_{\Gamma_{\text{sym}}}(\mathbf{u}_h)$  and  $\underline{\rho}_{\Gamma_{\text{sym}}}(\mathbf{u}_h)$  are as given in (18).

Assuming  $\mathbf{z}|_\kappa \in [H^{s_\kappa+1}(\kappa)]^5$ ,  $\kappa \in \mathcal{T}_h$ , and applying Cauchy–Schwarz inequality and the approximation estimates (27) and (28) in (29) we obtain

$$|J(\mathbf{u}) - J(\mathbf{u}_h)| \leq \left( \sum_{\kappa \in \mathcal{T}_h} (\eta_\kappa^{\text{res}})^2 \right)^{1/2}, \tag{30}$$

where  $\eta_\kappa^{\text{res}}$  are given by

$$\eta_\kappa^{\text{res}} = h_\kappa^{t_\kappa+1} \|\mathbf{R}(\mathbf{u}_h)\|_\kappa + h_\kappa^{t_\kappa+1/2} \|\mathbf{r}_{\partial\kappa}(\mathbf{u}_h)\|_{\partial\kappa} + h_\kappa^{t_\kappa-1/2} \|\underline{\rho}_{\partial\kappa}(\mathbf{u}_h)\|_{\partial\kappa} \tag{31}$$

with  $t_\kappa = \min(s_\kappa, p)$ ,  $\kappa \in \mathcal{T}_h$ . Here, we use the short notation  $\mathbf{r}_{\partial\kappa} = \mathbf{r}$  on  $\partial\kappa \setminus \Gamma$ ,  $\mathbf{r}_{\partial\kappa} = \mathbf{r}_\Gamma$  on  $\Gamma$ , and  $\mathbf{r}_{\partial\kappa} = \mathbf{r}_{\Gamma_{\text{sym}}}$  on  $\Gamma_{\text{sym}}$ , i.e.

$$\|\mathbf{r}_{\partial\kappa}(\mathbf{u}_h)\|_{\partial\kappa}^2 = \|\mathbf{r}(\mathbf{u}_h)\|_{\partial\kappa \setminus \Gamma}^2 + \|\mathbf{r}_\Gamma(\mathbf{u}_h)\|_\Gamma^2 + \|\mathbf{r}_{\Gamma_{\text{sym}}}(\mathbf{u}_h)\|_{\Gamma_{\text{sym}}}^2,$$

and analogously for  $\underline{\rho}_{\partial\kappa}$ :

$$\|\underline{\rho}_{\partial\kappa}(\mathbf{u}_h)\|_{\partial\kappa}^2 = \|\underline{\rho}(\mathbf{u}_h)\|_{\partial\kappa \setminus \Gamma}^2 + \|\underline{\rho}_\Gamma(\mathbf{u}_h)\|_\Gamma^2 + \|\underline{\rho}_{\Gamma_{\text{sym}}}(\mathbf{u}_h)\|_{\Gamma_{\text{sym}}}^2.$$

We note that  $s_\kappa$ ,  $\kappa \in \mathcal{T}_h$ , depends on the smoothness of the adjoint solution  $\mathbf{z}$ . However, in practice we cannot expect  $\mathbf{z}$  to be better than  $\mathbf{z} \in [H^1(\Omega)]^5$ . We choose  $s_\kappa = 0$ , and thus  $t_\kappa = 0$ ,  $\kappa \in \mathcal{T}_h$ , in (31) and obtain the following residual-based indicators:

$$\eta_\kappa^{\text{res}} = h_\kappa \|\mathbf{R}(\mathbf{u}_h)\|_\kappa + h_\kappa^{1/2} \|\mathbf{r}_{\partial\kappa}(\mathbf{u}_h)\|_{\partial\kappa} + h_\kappa^{-1/2} \|\underline{\rho}_{\partial\kappa}(\mathbf{u}_h)\|_{\partial\kappa}. \tag{32}$$

Note, that (30) and (32) is the extension of Corollary 4.5 in [17] for the adjoint inconsistent discretization of the 2d compressible Navier–Stokes equations given in [17] to the adjoint consistent discretization, see [18], of the 3d compressible Navier–Stokes considered in this article.

## 7. Adjoint-based indicators

The approximate error representation (23) can be rewritten as follows

$$J(\mathbf{u}) - J(\mathbf{u}_h) \approx \mathcal{R}(\mathbf{u}_h, \tilde{\mathbf{z}}_h) = \sum_{\kappa \in \mathcal{T}_h} \tilde{\eta}_\kappa. \tag{33}$$

Here, the indicators  $\tilde{\eta}_\kappa$ ,  $\kappa \in \mathcal{T}_h$ , are the so-called adjoint-based indicators given by

$$\begin{aligned}
\tilde{\eta}_\kappa &= \int_\kappa \mathbf{R}(\mathbf{u}_h) \cdot \tilde{\mathbf{z}}_h \, d\mathbf{x} + \int_{\partial\kappa \setminus \Gamma} \mathbf{r}(\mathbf{u}_h) \cdot \tilde{\mathbf{z}}_h^+ + \underline{\rho}(\mathbf{u}_h) : \nabla_h \tilde{\mathbf{z}}_h^+ \, ds + \int_{\kappa \cap \Gamma} \mathbf{r}_\Gamma(\mathbf{u}_h) \cdot \tilde{\mathbf{z}}_h^+ + \underline{\rho}_\Gamma(\mathbf{u}_h) : \nabla \tilde{\mathbf{z}}_h^+ \, ds \\
&\quad + \int_{\kappa \cap \Gamma_{\text{sym}}} \mathbf{r}_{\Gamma_{\text{sym}}}(\mathbf{u}_h) \cdot \tilde{\mathbf{z}}_h^+ + \underline{\rho}_{\Gamma_{\text{sym}}}(\mathbf{u}_h) : \nabla \tilde{\mathbf{z}}_h^+ \, ds,
\end{aligned} \tag{34}$$

where the primal element residuals  $\mathbf{R}(\mathbf{u}_h)$ , the interior face residuals  $\mathbf{r}(\mathbf{u}_h)$  and  $\underline{\rho}(\mathbf{u}_h)$ , the boundary residuals  $\mathbf{r}_\Gamma(\mathbf{u}_h)$  and  $\underline{\rho}_\Gamma(\mathbf{u}_h)$  and the symmetry boundary residuals  $\mathbf{r}_{\Gamma_{\text{sym}}}(\mathbf{u}_h)$  and  $\underline{\rho}_{\Gamma_{\text{sym}}}(\mathbf{u}_h)$  are as given in (18).

These local error indicators include the local primal residuals weighted with the discrete adjoint solution and are also denoted as dual-weighted-residual indicators (DWR indicators), see e.g. [4]. These local indicators can be used to drive an adaptive refinement (and coarsening) algorithm specifically tailored to the accurate and efficient approximation of the target quantity  $J(\mathbf{u})$ . For example, suppose that the aim of the computation is to compute  $J(\cdot)$  such that the error  $|J(\mathbf{u}) - J(\mathbf{u}_h)|$  is less than some user-defined tolerance  $\text{TOL}$ , i.e.  $|J(\mathbf{u}) - J(\mathbf{u}_h)| \leq \text{TOL}$ , then in practice we may enforce the stopping criterion  $|\sum_{\kappa \in \mathcal{T}_h} \tilde{\eta}_\kappa| \leq \text{TOL}$ . If this condition is not satisfied on the current finite element mesh  $\mathcal{T}_h$ , then the local indicators  $\tilde{\eta}_\kappa$  are employed as local error indicators to guide mesh refinement and coarsening. Based on the absolute values of the local indicators  $|\tilde{\eta}_\kappa|$  we select a fixed fraction of all elements for refinement and coarsening: typically 20% of the largest values for refinement and 10% of the smallest values for coarsening.

## 8. Anisotropic mesh refinement

The mesh refinement indicators presented in Sections 6 and 7 provide only the information which elements should be refined in order to improve the accuracy of the resulting solution. They do not include any directional information, thus an extension is required for anisotropic mesh refinement. Our first approach is based on an additional anisotropic indicator used to decide whether splitting just a subset of an element's edges and thus modifying the child elements' aspect ratios is preferable over splitting all edges. In the latter case the refinement is isotropic in the sense that child elements inherit the aspect ratio of the mother element. The heuristic jump indicator considered here was introduced in [26] for two-dimensional flows. For completeness, we recall the most relevant details and extend them to three-dimensional problems. The heuristic anisotropy detection of this first approach can be combined with an adjoint-based error estimator for goal-oriented refinement. Instead of using two separate indicators our second approach is based on an anisotropic extension of the *a posteriori* error estimate itself.

### 8.1. Jump indicator

One of the most characteristic features of DG methods is the possible discontinuity of its discrete solutions. In fact, a discrete solution may have jumps across the faces between neighboring elements, whereas it is smooth inside each element. These jumps allow some flexibility in approximating the local properties of the solution. In smooth parts of the solution these jumps tend to zero with successive mesh refinement as the solution is approximated with less error. Based on this observation it seems justified to assume that a large jump indicates a larger error as compared to a smaller jump. In view of an anisotropic evaluation a large jump over a face indicates that the mesh size perpendicular to this face is too coarse to sufficiently resolve the solution. In this sense inter-element jumps can be used to derive an anisotropic indicator that uses information which is specific to the numerical method used to solve the problem. Near discontinuities of the solution, like shocks, the jumps might not tend to zero under mesh refinement. However, in this case a large jump detects this discontinuity and suggests a refinement improving the resolution orthogonal to this feature, which is the correct behavior. Thus, the inter-element jumps can be used as an indicator in both smooth and non-smooth regions of the solution.

In order to obtain directional information, the average jump  $K_i$  of a function  $\phi$  over the two opposite faces  $f_i^j$ ,  $j = 1, 2$ , perpendicular to one coordinate direction  $i$  on the reference element can be evaluated as

$$K_i = \frac{\sum_j \left| \int_{f_i^j} [\phi] ds \right|}{\sum_j \text{meas}(f_i^j)}, \quad i = 1, 2, 3, \quad (35)$$

where  $[\phi] = \phi^+ - \phi^-$  denotes the jump of a scalar function  $\phi$ , the summations run over  $j = 1, 2$ , and  $\int \cdot ds$  indicates a surface integral in three dimensions. Eq. (35) provides three distinct values for each element. Let  $K_m$  denote the maximum value of  $K_i$ ,  $i = 1, 2, 3$ . We want to refine along each direction  $l$  in which the average jump is not considerably smaller than  $K_m$ . In order to quantify *considerably*, we introduce a threshold factor  $\theta > 1$ . Thus we refine along each direction  $l$  for which

$$\theta K_l > K_m, \quad l = 1, 2, 3. \quad (36)$$

Depending on the relative sizes of the average jumps in the individual directions, several cases may occur, see Fig. 1. If the jump is particularly large in one direction, the element will be refined only along that direction. If the jump in one direction is particularly small, whereas the other two values are of similar size, the element will be refined along the other two directions. If all the three average jumps have similar size we fall back to isotropic refinement.

If the solution function is vector-valued, as is the case for the flow equations, the jump of a scalar function  $\phi$  in Eq. (35) has to be replaced by an appropriate norm of the vector of jumps, for example the  $l_2$ -norm.

The empirical threshold factor  $\theta$  has to be chosen large enough to ensure that only those elements are flagged for anisotropic refinement which are located near strong anisotropic features, otherwise the error would not be reduced sufficiently. On the other hand, however, a smaller value of  $\theta$  allows more elements to be treated anisotropically, thereby leading to a reduced number of total elements. Numerical experiments showed that  $\theta = 5.0$  is a good choice for a range of test problems.

### 8.2. Anisotropic adjoint-based error indicator

In the context of mesh adaptation via local element subdivision Richter [28] recently presented an anisotropic adjoint-based *a posteriori* error estimate. The basic idea is to replace the isotropic (polynomial) enrichment of the space in which the adjoint solution  $\bar{\mathbf{z}}_h$  is computed by a space which is only enriched in one direction. This provides one error estimate for each of the coordinate directions on the reference hexahedron. This estimate indicates the part of the local error which can be reduced by refining the element along the same direction (or by increasing the polynomial degree of the discrete ansatz space in that direction). Repeating this error estimation with different enrichment directions yields three distinct indicators for each element. The selection of refinement is then applied directly to all those indicators at once, selecting from the refinement choices given by each uni-axial anisotropic refinement of each element. Isotropic refinement is only created if the three directions on a given element have independently been selected for refinement.



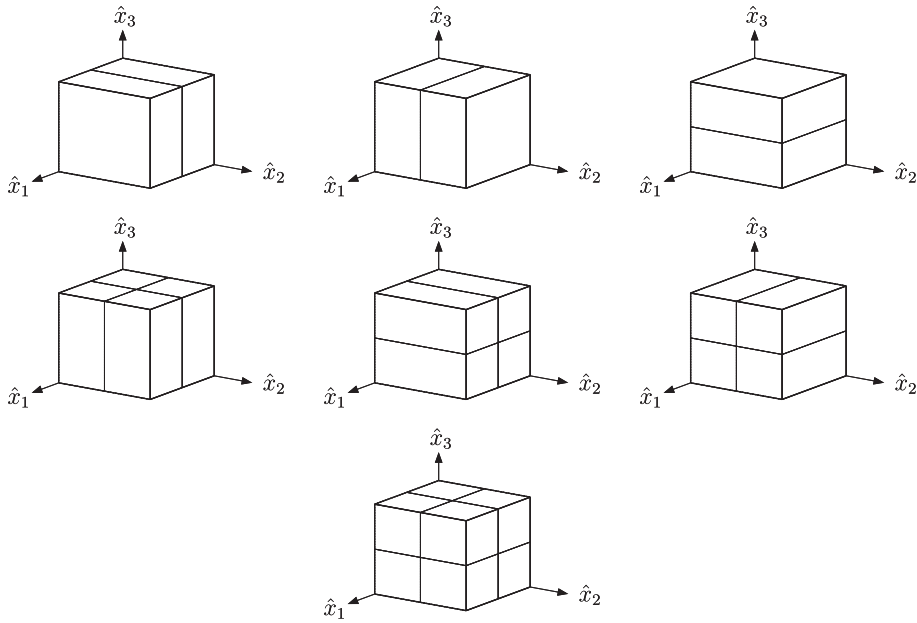


Fig. 1. Possible anisotropic and isotropic refinement cases on the 3d reference element.

In analogy to (6) and (7) let us denote the spaces of anisotropic tensor product polynomials of degrees  $p, q, r \geq 0$  on the reference element by

$$\mathcal{Q}_{p,q,r}(\hat{\kappa}) = \text{span}\{\hat{\mathbf{x}}^{\boldsymbol{\alpha}} : 0 \leq \alpha_1 \leq p, 0 \leq \alpha_2 \leq q, 0 \leq \alpha_3 \leq r\}, \tag{37}$$

and the finite element space of vector-valued anisotropic tensor product polynomial functions of degrees  $p, q, r \geq 0$  by

$$\mathbf{V}_h^{p,q,r} = \{\mathbf{v}_h \in [L_2(\Omega)]^5 : \mathbf{v}_h|_{\kappa} \circ \sigma_{\kappa} \in [\mathcal{Q}_{p,q,r}(\hat{\kappa})]^5, \kappa \in \mathcal{T}_h\}. \tag{38}$$

Note that  $\mathcal{Q}_{p,p,p}(\hat{\kappa}) = \mathcal{Q}_p(\hat{\kappa})$  and  $\mathbf{V}_h^{p,p,p} = \mathbf{V}_h^p$ . Richter [28] considers continuous finite elements on patchwise refined meshes, where the adjoint solution is computed in the continuous equivalent of  $\mathbf{V}_h^p$  and then reconstructed to  $\mathbf{V}_h^{2p,p,p}$ ,  $\mathbf{V}_h^{p,2p,p}$  and  $\mathbf{V}_h^{p,p,2p}$  in turn. Due to the continuous nature this reconstruction is little more than a reinterpretation of existing degrees of freedom on a super-element. In the discontinuous and unstructured context considered here we prefer to obtain an isotropically enriched solution in  $\mathbf{V}_h^{p+1}$ , either through direct computation or some reconstruction technique. This solution could then be  $L_2$ -projected to  $\mathbf{V}_h^{p+1,p,p}$ ,  $\mathbf{V}_h^{p,p,p+1}$  and  $\mathbf{V}_h^{p,p,p+1}$  in turn to simulate the directional reconstruction. However, the three anisotropic error indicators obtained that way for a given element do in general not sum up to the isotropic one. If a certain component of the error does only show up in a combined enrichment of the space and thus this error component can only be reduced by isotropic refinement this information will be lost using a simple projection technique. We believe that the reproduction of the isotropic estimate for isotropic cases is an important property and thus suggest to include those effects through the following reasoning.

The adjoint-based error estimate obtained for  $\mathbf{z}_h^{p+1}$  in  $\mathbf{V}_h^{p+1}$  can also be obtained by considering  $\mathbf{z}_h^{p+1} - \mathbf{z}_h^p$  with any  $\mathbf{z}_h^p$  in  $\mathbf{V}_h^p$  due to Galerkin orthogonality. Furthermore,  $\mathbf{z}_h^{p+1} - \mathbf{z}_h^p$  can be represented in a space spanned by hierarchic and orthogonal polynomials, e.g. the tensor product space of Legendre polynomials. If we choose  $\mathbf{z}_h^p$  as the  $L_2$  projection of  $\mathbf{z}_h^{p+1}$  to  $\mathbf{V}_h^p$  we can reinterpret the  $L_2$  projection from  $\mathbf{V}_h^{p+1}$  to  $\mathbf{V}_h^{p+1,p,p}$  as a modification of the coefficients in an unchanged space, in particular the projection corresponds to setting all coefficients of  $\mathbf{z}_h^{p+1}$  to zero except those for which the basis function is of degree  $p + 1$  in the  $\hat{x}_1$  direction and of smaller degree in the other two directions; analogous observations are true for the remaining directions. Summing those three projections up and comparing to the full function we lost those coefficients for which the basis function is of degree  $p + 1$  in more than one coordinate direction. We suggest to split those coefficients and add them in equal parts to the projections in the involved directions such that the three modified projections sum up to the initial function and isotropic effects are included in the anisotropic indicators. Denoting the selection-based analogy to projection in the  $\hat{x}_1$  direction by the element-wise operator  $\mathcal{S}_{\kappa}^{\mathbf{x}_1}$  we obtain the anisotropic error indicator  $\eta_{\kappa}^{\mathbf{x}_1}$  by

$$\eta_{\kappa}^{\mathbf{x}_1} = \mathcal{R}\left(\mathbf{u}_h, \mathcal{S}_{\kappa}^{\mathbf{x}_1}(\mathbf{z}_h^{p+1} - \mathbf{z}_h^p)\right), \tag{39}$$

$\eta_{\kappa}^{\mathbf{x}_2}$  and  $\eta_{\kappa}^{\mathbf{x}_3}$  are obtained correspondingly. Note that  $\eta_{\kappa}^{\mathbf{x}_1} + \eta_{\kappa}^{\mathbf{x}_2} + \eta_{\kappa}^{\mathbf{x}_3} = \tilde{\eta}_{\kappa}$  as defined in (34) due to the fact that  $\mathcal{S}_{\kappa}^{\mathbf{x}_1}(\mathbf{z}_h^{p+1} - \mathbf{z}_h^p) + \mathcal{S}_{\kappa}^{\mathbf{x}_2}(\mathbf{z}_h^{p+1} - \mathbf{z}_h^p) + \mathcal{S}_{\kappa}^{\mathbf{x}_3}(\mathbf{z}_h^{p+1} - \mathbf{z}_h^p) = (\mathbf{z}_h^{p+1} - \mathbf{z}_h^p)$ .

From an implementational point of view the transfer of the elemental adjoint solution vector to a hierarchic basis, the selection operation and the transfer back to whatever basis is used in the remaining computation can be performed by a single matrix–vector product with a matrix that does not depend on the element under consideration as all operations are done on the reference element. Thus the cost of evaluating these indicators is negligible. Furthermore, the selection operators  $S_{\kappa}^{x_i}$ ,  $i = 1, 2, 3$ , can easily be extended to spaces of complete polynomials instead of tensor product polynomials. In contrast to that a projection or even the direct anisotropic reconstruction technique is not readily available as it is already difficult to define anisotropic versions of such spaces.

A general advantage of this direct anisotropic error estimation approach is the fact that no threshold is required to distinguish strong and weak anisotropies, each single refinement direction is simply treated independently. Additionally, as the adjoint solution provides a suitable weighting of the components no special treatment is required for vector-valued solution functions.

To obtain optimal efficiency of the adaptive algorithm the fraction of refinements to be performed in one adaptation cycle w.r.t. the total number of possible refinements should be reduced compared to the value used for isotropic refinement. Depending on the strength of anisotropic features values between 10% and 15% provide good results in our experience.

## 9. Numerical examples

In this section we demonstrate the performance of the proposed error estimation and adaptation algorithms for two laminar three-dimensional test cases. In all subsequent computations the flow solutions are computed with  $p = 1$ , the adjoint solutions are computed with  $p = 2$ , the penalty constant is chosen as  $C_{IP} = 30$ , cf. (9), and the Vijayasundaram numerical flux function, see [31], is used. The flow problems are solved with the fully implicit Newton algorithm. The linear problems arising in each Newton step as well as the linear adjoint problems are solved with an ILU preconditioned GMRES algorithm. We note that the number of elements in a mesh grows significantly with the dimension and the number of unknowns per element increases rapidly with both dimension and polynomial degree. Thus, this approach is not applicable to large-scale computations due to its memory requirements. Current and future research which is beyond the scope of this work is dedicated to memory-lean flow and adjoint solvers.

### 9.1. Some remarks on local element subdivision

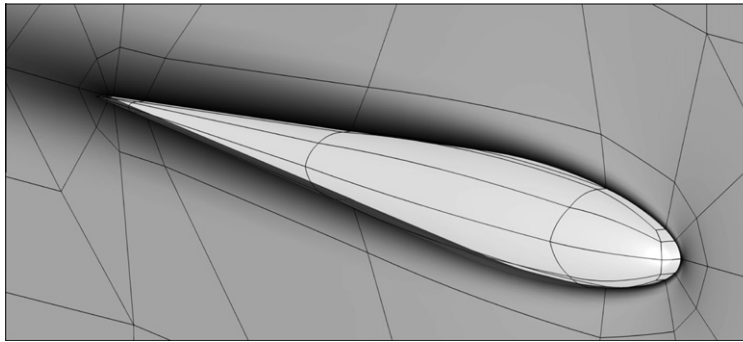
The local element subdivision approach used in our work and thus in the presented numerical examples is substantially different from the remeshing approach presented elsewhere, especially in the area of anisotropic mesh refinement. Although the focus of this work is not a thorough comparison of those approaches some remarks are in order to interpret the following results. Using a solution-adaptive remeshing the new mesh has only little in common with the initial one: aspect ratios, orientations and local node densities can be completely different without even modifying the total number of elements. This is substantially different for element subdivision: only a global change in the number of elements can produce a locally modified node density. Element orientation stays the same on refined meshes and so does the aspect ratio, except in the anisotropic case where it can be increased or decreased by a factor of two in each adaptation cycle. The latter approach guarantees that a new mesh can be generated, whereas a remeshing process might fail. However, recent improvements in mesh generation software as well as the consideration of local node insertion and mesh optimization instead of global remeshing have made the remeshing process quite robust. Thus, both approaches are quite reliable in practice.

Due to this difference our initial meshes are not isotropic and homogeneous but somehow “appropriate” to the case at hand. We believe that there is substantial experience in creating such meshes for compressible flows and that this should be exploited. Thus, simply increasing the node density globally on these meshes is already quite efficient to reduce the error. We target providing still some improvement. The examples below show that the isotropic algorithm – which still features anisotropic elements – does a good job at that. In addition to that, the anisotropic refinement can further reduce the required effort. The resulting reduction might not seem huge, but it is achieved in comparison with realistic alternatives. Comparing to intentionally bad choices might yield larger improvements but is not particularly meaningful.

### 9.2. Laminar flow around a streamlined body

First, we consider a streamlined three-dimensional body based on a 10% thick airfoil with boundaries constructed by a surface of revolution, see Fig. 2. It consists of an elliptical leading edge and straight lines. The flow is considered at laminar conditions with inflow Mach number equal to 0.5, at an angle of attack  $\alpha = 1^\circ$ , and Reynolds number  $Re = 5000$  with adiabatic no-slip wall boundary condition imposed. The geometry and the flow is relatively simple. In fact, this test case has been defined in the EU project ADIGMA [24] to enable convergence studies. A reference drag coefficient value of  $C_d^{\text{ref}} = 0.06317$  has been obtained by extrapolation of results from computations with higher order schemes on a series of finer meshes.

We note that in all subsequent computations the boundary of the curved body is approximated using piecewise bi-quadratic polynomials where the additional points required for defining these polynomials are obtained from a CAD representation of the geometry. Similarly, also the new points on the boundary required during local mesh refinement near the body are taken from the CAD representation.



**Fig. 2.** Streamlined body: Initial coarse mesh on the body surface and the symmetry plane. The symmetry plane coloring is based on the Mach number distribution computed on a fine mesh.

The target of following computations will be to efficiently approximate the drag coefficient on a sequence of locally refined meshes. To this end, we perform the error estimation algorithm described in Section 5 on locally refined meshes adapted using the adjoint-based indicators (34) where the adjoint problem (24) is connected to the drag coefficient (20). The first sequence of locally refined meshes is based on isotropic mesh refinement, i.e. upon refinement each hexahedral element is isotropically subdivided into eight hexahedral subelements. In Table 1 we collect the number of elements, the number of degrees of freedom (DoFs) of  $\mathbf{u}_h \in \mathbf{V}_h^1$ , the “true” error  $J(\mathbf{u}) - J(\mathbf{u}_h) = C_d^{\text{ref}} - C_d$  in the drag coefficient, the estimated error  $\mathcal{E} = \sum_{k \in \mathcal{T}_h} \tilde{\eta}_k$ , (33), and the quotient  $\theta = \mathcal{E} / (J(\mathbf{u}) - J(\mathbf{u}_h))$  of the estimated and the true error which is also called the effectivity index. First of all, we see that on all meshes the sign of the error is predicted correctly. On the coarsest three meshes the error estimates are not particular accurate indicated by an effectivity index  $\theta$  in the range of [0.6, 2.7]. However, as the mesh is refined the effectivity index  $\theta$  converges to one corresponding to error estimates being very close to the true errors.

Table 2 collects the corresponding data on a sequence of an isotropically refined meshes. Here, on each element depicted for local refinement by the adjoint-based indicators the anisotropic jump indicator (35) is used to determine which of the seven different refinement cases shown in Fig. 1 are applied. Here, we see that the error estimation behaves very similar to the one described for the sequence of the isotropically refined meshes in Table 1, i.e. the effectivity of the error estimation does not deteriorate on anisotropically refined meshes.

Finally, Table 3 collects the data obtained through application of the anisotropic adjoint-based error estimate (39). The results are again quite similar, also on this sequence of meshes the effectivity index is close to one after some refinement cycles, where a value larger than 0.9 indicates that the remaining part of the error which is not estimated is more than an order of magnitude smaller than the original error.

**Table 1**

Streamlined body: adaptive algorithm for the accurate approximation of the drag coefficient on a sequence of isotropically refined meshes.

# el.	# DoFs	$C_d^{\text{ref}} - C_d$	$\mathcal{E}$	$\theta$
768	30,720	-9.877e-04	-6.548e-04	0.66
1853	74,120	1.731e-03	4.690e-03	2.71
4744	189,760	-8.159e-04	-5.146e-04	0.63
12,304	492,160	-5.067e-04	-4.732e-04	0.93
32,282	1,291,280	-2.885e-04	-2.743e-04	0.95
81,688	3,267,520	-1.123e-04	-1.062e-04	0.95

**Table 2**

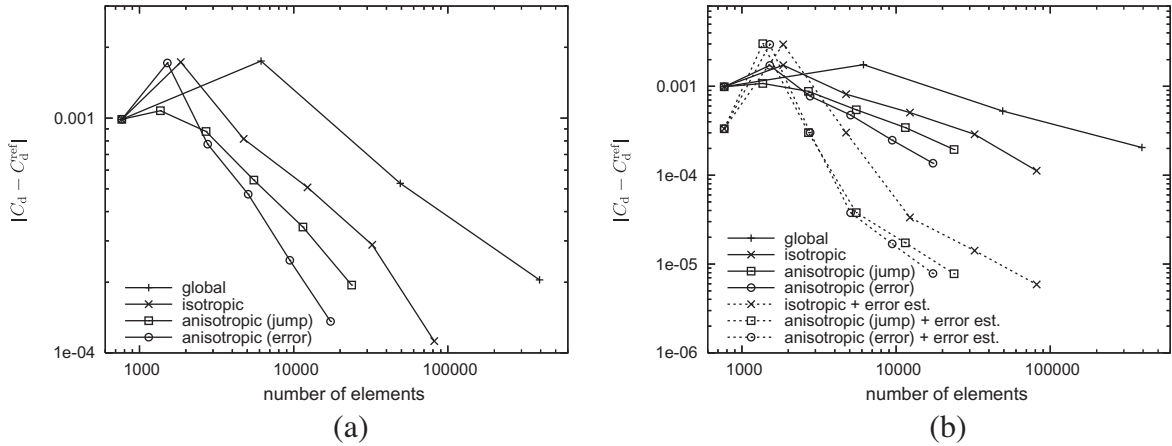
Streamlined body: adaptive algorithm for the accurate approximation of the drag coefficient on a sequence of anisotropically refined meshes based on the jump indicator.

# el.	# DoFs	$C_d^{\text{ref}} - C_d$	$\mathcal{E}$	$\theta$
768	30,720	-9.877e-04	-6.548e-04	0.66
1366	54,640	1.075e-03	4.096e-03	3.81
2700	108,000	-8.771e-04	-5.759e-04	0.66
5518	220,720	-5.446e-04	-5.067e-04	0.93
11,483	459,320	-3.434e-04	-3.261e-04	0.95
23,773	950,920	-1.946e-04	-1.868e-04	0.96

**Table 3**

Streamlined body: adaptive algorithm for the accurate approximation of the drag coefficient on a sequence of anisotropically refined meshes based on the anisotropic adjoint-based error estimate.

# el.	# DoFs	$C_d^{\text{ref}} - C_d$	$\mathcal{E}$	$\theta$
768	30,720	-9.877e-04	-6.548e-04	0.66
1517	60,680	1.719e-03	4.681e-03	2.72
2768	110,720	-7.745e-04	-4.724e-04	0.61
5073	202,920	-4.742e-04	-4.364e-04	0.92
9444	377,760	-2.479e-04	-2.310e-04	0.93
17,329	693,160	-1.362e-04	-1.284e-04	0.94



**Fig. 3.** Streamlined body: convergence of the error in the drag coefficients  $J(\mathbf{u}_h)$  for global in comparison to adjoint-based isotropic and adjoint-based anisotropic mesh refinement algorithms. Additionally, (b) shows the errors of the enhanced drag coefficients  $\tilde{J}(\mathbf{u}_h) = J(\mathbf{u}_h) + \mathcal{E}$  for the adjoint-based isotropic and anisotropic mesh refinement.

Fig. 3(a) plots the error in the drag coefficient  $|C_d^{\text{ref}} - C_d|$  against the number of elements for a sequence of globally refined meshes, the sequence of adjoint-based isotropic refined meshes, see Table 1, and the sequences of adjoint-based anisotropic refined meshes, see Tables 2 and 3. Comparing the histories of global and adjoint-based isotropic refinement we see in Fig. 3 that for this test case the adjoint-based refinement leads to meshes with a factor of about 5 less elements for a specific accuracy in the drag coefficient as compared to global refinement. Moreover, we see that compared to the isotropic adjoint-based mesh refinement there is another factor of about 2 in the mesh sizes required for a specific accuracy for the anisotropic refinement based on the jump indicator and a factor of approximately 4 for the anisotropic refinement based directly on the anisotropic error estimate.

A comparison of the resulting adapted meshes is given in Fig. 4. As anisotropic features are not particularly strong and the initial mesh already shows some anisotropy the overall effect of anisotropic refinement seems rather weak. However, we note that in some places the initial aspect ratio is further increased in the anisotropic case, mainly in the boundary layer mesh which can be seen in the symmetry plane. On the other hand, the stretching of some cells along the body with small edge length orthogonal to the flow is too pronounced in the initial mesh. During isotropic refinement this aspect ratio is inherited to all child elements. The anisotropic refinement algorithm can modify aspect ratios, however, and it does so. For some elements this yields a *reduced* aspect ratio in order to create the mesh best fitted to the (quite isotropic) problem at hand.

The error estimates in Tables 1–3 can be used to enhance the computed drag coefficients  $J(\mathbf{u}_h) \equiv C_d$  as follows:  $\tilde{J}(\mathbf{u}_h) := J(\mathbf{u}_h) + \mathcal{E}$ . If the error estimation is reliable such enhanced target quantities can be expected to be significantly more accurate than the original values  $J(\mathbf{u}_h)$ . This is confirmed in Fig. 3(b) which repeats Fig. 3(a) in a different scale and additionally shows the histories of the errors of the enhanced drag coefficients  $\tilde{J}(\mathbf{u}_h)$ . In fact, from the third mesh onwards the enhanced drag coefficients are much closer to the reference value. The large vertical distance in the convergence plot is a graphical interpretation of an effectivity index close to one.

### 9.3. Laminar flow around a delta wing

As a second test case we consider a laminar flow around a delta wing. The delta wing has a sloped and sharp leading edge and a blunt trailing edge. A similar case has previously been considered in [22]. The geometry of the delta wing can be seen

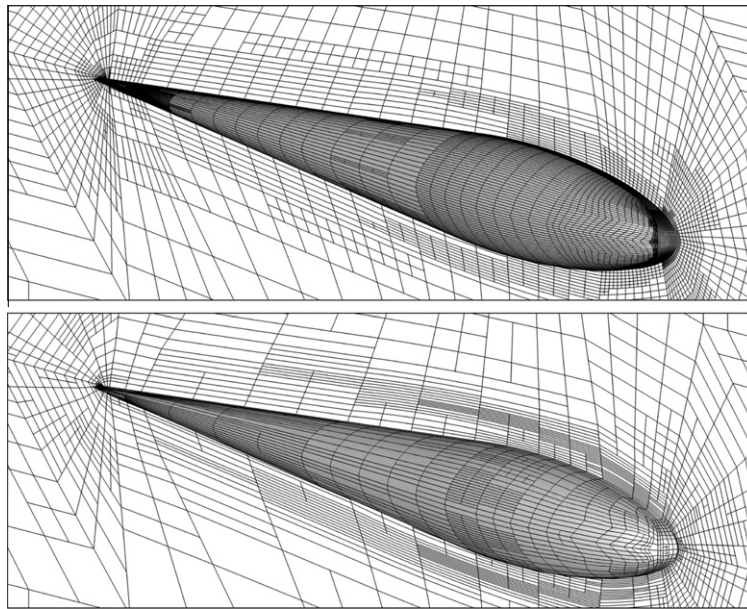


Fig. 4. Streamlined body: adapted surface meshes after five adaptation cycles. Top: isotropic refinement, bottom: anisotropic refinement.

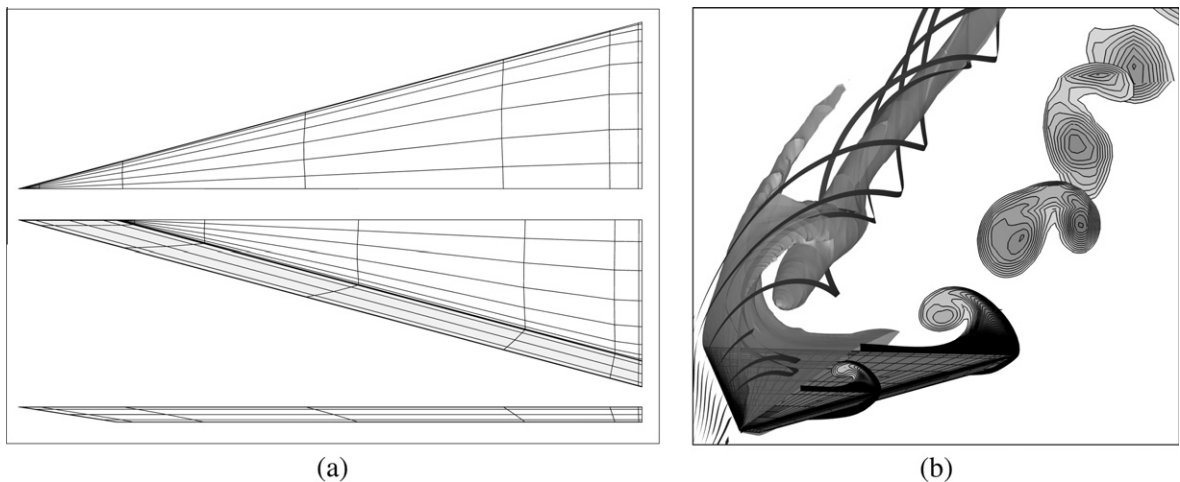


Fig. 5. Laminar delta wing: (a) initial surface mesh: top, bottom and side view of the half delta wing with straight leading edges, (b) solution plot showing streamlines and a Mach number isosurface over the left half of the wing as well as Mach number slices over the right half.

from the initial surface mesh in Fig. 5(a). The delta wing is considered at laminar conditions with inflow Mach number equal to 0.3, at an angle of attack  $\alpha = 12.5^\circ$ , and Reynolds number  $Re = 4000$  with isothermal no-slip wall boundary condition imposed on the wing geometry. This test case has been defined in the EU project ADIGMA [24]. As the flow passes the leading edge it rolls up, creates a vortex and a secondary vortex. The resulting vortex system remains over long distances behind the wing, see Fig. 5(b).

By performing higher order computations on a series of finer meshes and extrapolating the results the following reference values of the force coefficients have been obtained:  $C_d^{\text{ref}} = 0.1658$  and  $C_l^{\text{ref}} = 0.347$ .

In the following we will compare the performance in accurately approximating the drag and lift coefficients when using adjoint-based mesh refinement in comparison to residual-based and to global mesh refinement. Additionally, for the local mesh refinement strategies we will compare isotropic against anisotropic mesh refinement.

Let us first consider the drag coefficient. Performing the error estimation and adjoint-based mesh refinement algorithm with the adjoint problem connected to the drag coefficient we collect the data of the sequence of isotropically refined meshes in Table 4. Here we see that already on the coarse meshes the error estimation is quite accurate and it improves as the mesh is refined. A similar behavior we see in Table 4 for anisotropic mesh refinement based on the jump indicator and also for

**Table 4**

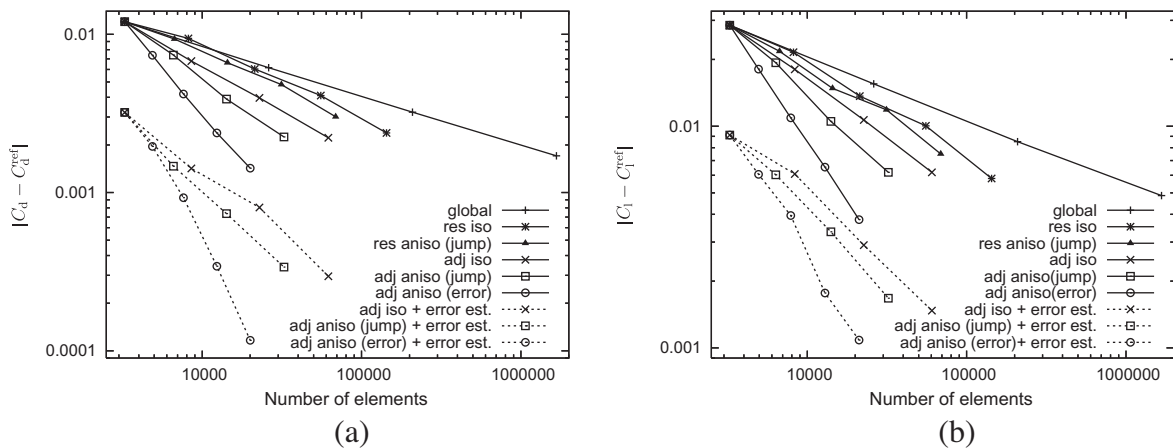
Laminar delta wing: adaptive algorithms for the accurate approximation of the drag coefficient on sequences of isotropically and anisotropically refined meshes.

Adaptive algorithm	# el.	# DoFs	$C_d^{ref} - C_d$	$\mathcal{E}$	$\theta$
Isotropic	3264	130,560	-1.202e-02	-8.808e-03	0.73
	8549	341,960	-6.772e-03	-5.352e-03	0.79
	22,885	915,400	-3.968e-03	-3.163e-03	0.80
	61,868	2,474,720	-2.221e-03	-1.925e-03	0.87
Anisotropic jump indicator	3264	130,560	-1.202e-02	-8.808e-03	0.73
	6600	264,000	-7.398e-03	-5.931e-03	0.80
	14,215	568,600	-3.895e-03	-3.160e-03	0.81
	32,621	1,304,840	-2.247e-03	-1.909e-03	0.85
Anisotropic error estimate	3264	130,560	-1.202e-02	-8.808e-03	0.73
	4866	194,640	-7.366e-03	-5.409e-03	0.73
	7622	304,880	-4.199e-03	-3.271e-03	0.78
	12,347	493,880	-2.381e-03	-2.039e-03	0.86
	20,005	800,200	-1.425e-03	-1.309e-03	0.92

anisotropic refinement based on the anisotropic error estimate. The efficiency of the error estimation does not notable degrade on anisotropically refined meshes.

Fig. 6(a) plots the error in the drag coefficient  $|C_d^{ref} - C_d|$  against the number of elements for various refinement strategies: global mesh refinement, residual-based isotropic and anisotropic mesh refinement as well as adjoint-based isotropic mesh refinement and the two anisotropic variants. We notice that drag coefficients of a specific accuracy are obtained with less elements for residual-based mesh refinement than for global mesh refinement where this advantage increases for increasing accuracy requirements. Furthermore, there is a significant decrease of the number of elements required for a specific accuracy when comparing adjoint-based against residual-based refinement. Additionally, in case of adjoint-based mesh refinement Fig. 6(a) plots the errors of the enhanced drag coefficients  $\tilde{J}(\mathbf{u}_h) := J(\mathbf{u}_h) + \mathcal{E}$ . We note that already on the coarsest mesh the enhanced drag coefficient is almost as accurate as the drag coefficients on the finest adjoint-based refined mesh. Furthermore, we see that anisotropic mesh refinement always performs better than isotropic mesh refinement. In fact, anisotropic adjoint-based refinement based on the jump indicator requires about half the number of elements for almost the same accuracy than the corresponding isotropic refinement. For the anisotropic adjoint-based error estimate this gain increases further and the number of elements can be reduced by a factor of approximately five compared to the isotropic case. Overall, compared to a global mesh refinement approach the number of elements and thus degrees of freedom required to obtain the accuracy of the final globally refined mesh can be reduced by two orders of magnitude using the best available adaptive strategy. Using the enhanced drag coefficient improves the accuracy by another order of magnitude. This accumulates to an impressive gain.

Finally, we consider the lift coefficient. Table 5 collects the data of the sequences of isotropically and anisotropically adjoint-based refined meshes. For all three algorithms we see a behavior similar to that described for the drag coefficient, although the efficiency of the error estimation is slightly reduced in this case. Nevertheless, the adaptive algorithm based

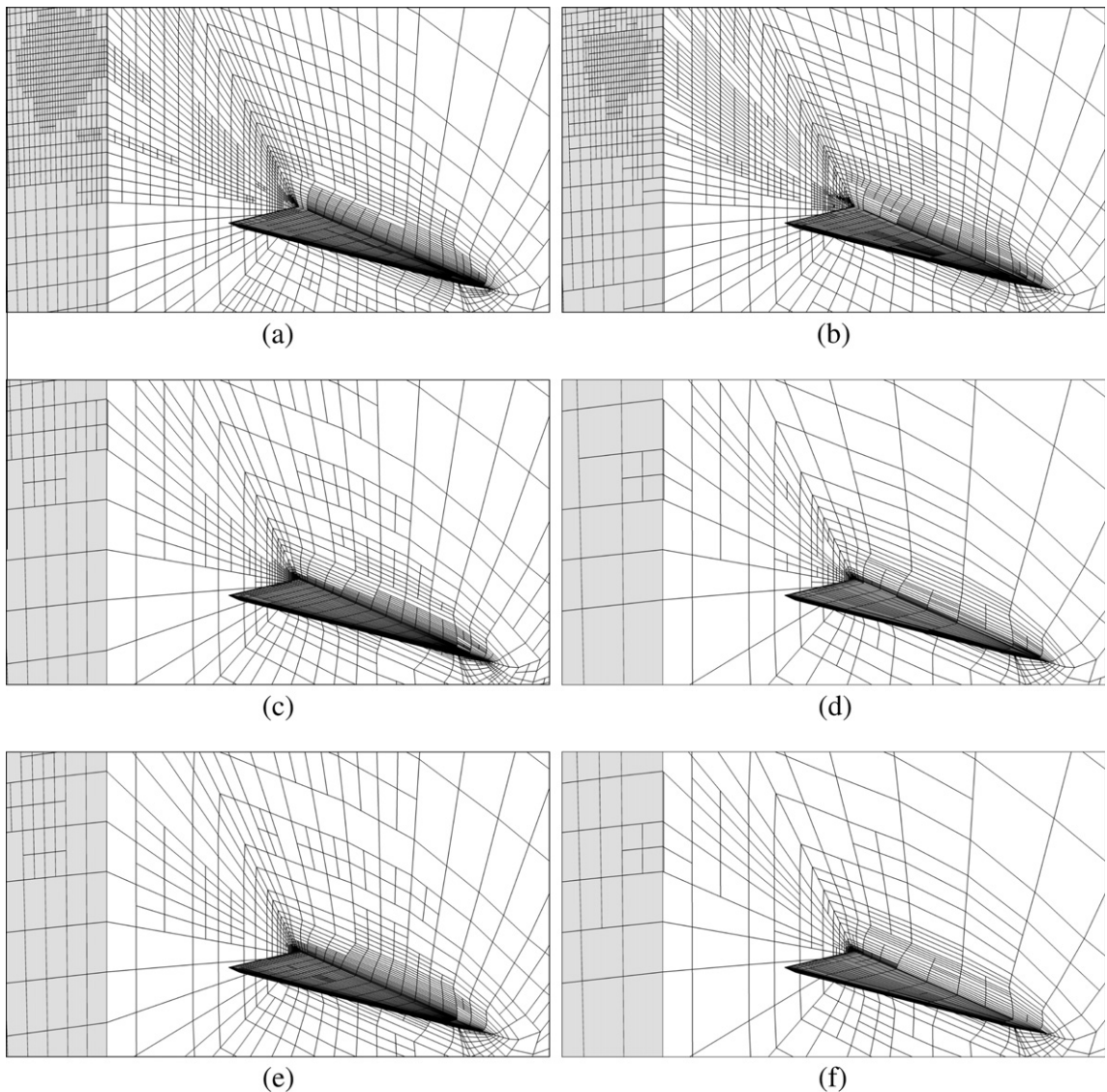


**Fig. 6.** Laminar delta wing: convergence of the error in the (a) drag and (b) lift coefficients  $J(\mathbf{u}_h)$  for global in comparison to residual-based (isotropic and anisotropic) and to adjoint-based (isotropic and anisotropic) mesh refinement. Additionally, the errors of the enhanced force coefficients  $\tilde{J}(\mathbf{u}_h) = J(\mathbf{u}_h) + \mathcal{E}$  on the sequences of adjoint-based mesh refinement are given.

**Table 5**

Laminar delta wing: adaptive algorithms for the accurate approximation of the lift coefficient on sequences of isotropically and anisotropically refined meshes.

Adaptive algorithm	# el.	# DoFs	$C_l^{\text{ref}} - C_l$	$\mathcal{E}$	$\theta$
Isotropic	3264	130,560	-2.851e-02	-1.939e-02	0.68
	8346	333,840	-1.804e-02	-1.196e-02	0.66
	22,647	905,880	-1.067e-02	-7.759e-03	0.73
	60,524	2,420,960	-6.187e-03	-4.715e-03	0.76
Anisotropic jump indicator	3264	130,560	-2.851e-02	-1.939e-02	0.68
	6339	253,560	-1.931e-02	-1.328e-02	0.69
	14,073	562,920	-1.051e-02	-7.169e-03	0.68
	32,274	1,290,960	-6.191e-03	-4.516e-03	0.73
Anisotropic error estimate	3264	130,560	-2.851e-02	-1.939e-02	0.68
	4948	197,920	-1.809e-02	-1.203e-02	0.66
	7877	315,080	-1.090e-02	-6.952e-03	0.64
	12,917	516,680	-6.531e-03	-4.763e-03	0.73
	21,141	845,640	-3.785e-03	-2.701e-03	0.71



**Fig. 7.** Laminar delta wing: adapted meshes, (a) and (b): four adaptation steps with the residual indicator, isotropic and anisotropic based on the jump indicator, (c) and (d): three adaptation steps with the adjoint-based indicator for the drag coefficient  $C_d$ , isotropic and anisotropic based on the anisotropic error estimate, (e) and (f): three adaptation steps with the adjoint-based indicator for the lift coefficient  $C_l$ , isotropic and anisotropic based on the anisotropic error estimate.

on those estimates still performs very well. Fig. 6(b) plots the errors for global, residual-based, adjoint-based, isotropic and anisotropic refinement and the errors of the enhanced lift coefficients. Here, again we see a behavior very similar to that described for the drag coefficient above.

Adapted meshes for six of the different combinations of error indicators and isotropic or anisotropic refinement are presented in Fig. 7. In order to present meshes for which the accuracy of the relevant target functional is comparable all plots are given for the last data point in the errors plots in Fig. 6, except in the case of the anisotropic error estimate for which the mesh is shown for the point prior to the last one. The outstanding effect is clearly the resolution of the vortex in the cut-plane behind the wing for the residual-based refinement indicator and the corresponding lack of resolution in this area in the case of goal-oriented refinement. It is quite obvious that the global flow field is better resolved using the first type of indicator whereas the resolution of this prominent vortex is not of much influence on the target functional values, as both the pressure at the wall and the skin friction are only weakly dependent on the downstream vortex evolution. Thus, investing more in the near-wall refinement the adjoint-based refinement indicators are capable of creating more efficient meshes for the approximation of the given target functional. We note, that the resulting refinement for the lift and drag coefficients has clear similarities, but that there are differences in the details of the created meshes. Furthermore, the effect of the anisotropic indicator can also be seen in the adapted meshes, not only in the plots and tables. Finally, we note that the refinement of the vortex footprint on the wing is particularly pronounced in the anisotropic residual-based case.

## 10. Conclusion and outlook

The numerical examples demonstrate that the SIPG discretization is capable of producing accurate results for 3d laminar flows. Furthermore, the applicability of the proposed local mesh refinement algorithm has been demonstrated. Starting from very coarse meshes the relevant features of the flow field are resolved on subsequent refined meshes, resulting in an accurate and efficient prediction of target functionals like aerodynamic force coefficients.

Both residual-based and adjoint-based adaptation are beneficial for improving the efficiency upon global mesh refinement. For the most efficient approximation of a specific target functional the goal-oriented strategy is the most effective one, especially as the availability of a global error estimate is a valuable additional feature and can be exploited to improve the computed target functional value. If, however, the goal of the computation is the global field solution of the flow under consideration, it is advisable to use a residual-based error estimation and refinement algorithm, instead.

The proposed anisotropic jump indicator is computationally very cheap, but is nevertheless able to achieve a perceptible additional reduction in the number of elements required to obtain a given accuracy. The “partitioned approach” of separating the local error estimation from the local anisotropy detection makes this indicator immediately available in both residual-based and goal-oriented refinement strategies. In the case of goal-oriented refinement the anisotropic error estimation provides even better results at virtually no cost overhead. Both versions are immediately applicable to higher order and variable order  $hp$ -discretizations. Our current focus is on boundary layers in which anisotropic features are aligned with the mesh. For shock-dominated flows those features might be oblique to the mesh and an efficient anisotropic mesh resolution cannot be achieved by selectively refining edges. In that case local node movement might be considered to create a properly aligned mesh.

A remaining problem of the current implementation is the restriction to purely hexahedral meshes. In practice, only (block-)structured meshes can fulfill this requirement. It is complicated to create such meshes for complex geometries and in particular coarse meshes often suffer from distorted elements. Thus it would be advisable to extend the discretization to hybrid meshes consisting of tetrahedra, prisms, pyramids and hexahedra.

The current flow and adjoint solvers rely on the assembly and storage of the full Jacobian matrix which is prohibitive for large-scale applications. Future research is dedicated to replace these solvers by memory-lean versions. In particular, we will consider a concurrent iteration of the primal and adjoint solutions within a  $p$ -multigrid method with line-implicit smoothing.

Most aerodynamic flows of practical interest are turbulent and transonic, thus the physical modeling should be extended to both turbulence modeling via additional transport equations and a reliable shock capturing technique. A reliable and robust extension of the discretization to these aspects will require a significant amount of future research.

## Acknowledgments

The authors acknowledge the partial financial support of both the President’s Initiative and Networking Fund of the Helmholtz Association of German Research Centres and the European project ADIGMA [24]. The initial meshes have been provided by partners of the ADIGMA consortium. Computations have been performed using the DG flow solver `PADGE` [15] based on the `deal.II` library [3], which *en route* to this work has been modified in order to allow for anisotropic refinement.

## References

- [1] D. Arnold, F. Brezzi, B. Cockburn, L. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems, *SIAM J. Numer. Anal.* 39 (5) (2002) 1749–1779.



- [2] I. Babuška, M. Suri, The *hp*-version of the finite element method with quasiuniform meshes, *Math. Model. Numer. Anal. (M<sup>2</sup>AN)* 21 (1987) 199–238.
- [3] W. Bangerth, R. Hartmann, G. Kanschat, deal.II – a general purpose object oriented finite element library, *ACM Trans. Math. Software* 33 (4) (2007).
- [4] R. Becker, R. Rannacher, An optimal control approach to a posteriori error estimation in finite element methods, *Acta Numer.* 10 (2001) 1–102.
- [5] Y. Bourgault, M. Picasso, F. Alauzet, A. Loseille, On the use of anisotropic a posteriori error estimators for the adaptive solution of 3D inviscid compressible flows, *Int. J. Numer. Methods Fluids* 59 (1) (2009) 47–74.
- [6] K.J. Fidkowski, A Simplex Cut-Cell Adaptive Method for High-Order Discretizations of the Compressible Navier–Stokes Equations, Ph.D. Thesis, Department of Aeronautics & Astronautics, Massachusetts Institute of Technology, 2007.
- [7] K.J. Fidkowski, D.L. Darmofal, A triangular cut-cell adaptive method for high-order discretizations of the compressible Navier–Stokes equations, *J. Comput. Phys.* 225 (2007) 1653–1672.
- [8] K.J. Fidkowski, D.L. Darmofal, An adaptive simplex cut-cell method for discontinuous Galerkin discretizations of the Navier–Stokes equations, in: 18th AIAA CFD Conference, AIAA-2007-3941, 2007.
- [9] L. Formaggia, S. Micheletti, S. Perotto, Anisotropic mesh adaptation in computational fluid dynamics: application to the advection–diffusion–reaction and the Stokes problems, *Appl. Numer. Math.* 51 (2004) 511–533.
- [10] P.J. Frey, F. Alauzet, Anisotropic mesh adaptation for CFD computations, *Comput. Methods Appl. Mech. Eng.* 194 (2005) 5068–5082.
- [11] M. Giles, E. Süli, Adjoint methods for PDEs: a posteriori error analysis and postprocessing by duality, *Acta Numer.* 11 (2002) 145–236.
- [12] K. Harriman, D. Gavaghan, E. Süli, The Importance of Adjoint Consistency in the Approximation of Linear Functionals using the Discontinuous Galerkin Finite Element Method, Technical Report, Oxford University Computing Laboratory, 2004.
- [13] R. Hartmann, Adjoint consistency analysis of discontinuous Galerkin discretizations, *SIAM J. Numer. Anal.* 45 (6) (2007) 2671–2696.
- [14] R. Hartmann, Multitarget error estimation and adaptivity in aerodynamic flow simulations, *SIAM J. Sci. Comput.* 31 (1) (2008) 708–731.
- [15] R. Hartmann, J. Held, T. Leicht, F. Prill, PADGE, Parallel Adaptive Discontinuous Galerkin Environment, Technical reference, DLR, Braunschweig, 2009. In preparation.
- [16] R. Hartmann, P. Houston, Adaptive discontinuous Galerkin finite element methods for the compressible Euler equations, *J. Comput. Phys.* 183 (2) (2002) 508–532.
- [17] R. Hartmann, P. Houston, Symmetric interior penalty DG methods for the compressible Navier–Stokes equations II: goal-oriented a posteriori error estimation, *Int. J. Numer. Anal. Model.* 3 (2) (2006) 141–162.
- [18] R. Hartmann, P. Houston, An optimal order interior penalty discontinuous Galerkin discretization of the compressible Navier–Stokes equations, *J. Comput. Phys.* 227 (22) (2008) 9670–9685.
- [19] P. Houston, E.H. Georgoulis, E. Hall, Adaptivity and a posteriori error estimation for DG methods on anisotropic meshes, in: G. Lube, G. Rapin (Eds.), *International Conference on Boundary and Interior Layers, BAIL2006*, 2006.
- [20] W. Huang, Metric tensors for anisotropic mesh generation, *J. Comput. Phys.* 204 (2005) 633–665.
- [21] W.T. Jones, E.J. Nielsen, M.A. Park, Validation of 3D adjoint based error estimation and mesh adaptation for sonic boom prediction, in: 44th AIAA Aerospace Sciences Meeting, AIAA 2006-1150, 2006.
- [22] C.M. Klaij, J.J.W. van der Vegt, H. van der Ven, Space–time discontinuous Galerkin method for the compressible Navier–Stokes equations, *J. Comput. Phys.* 217 (2) (2006) 589–611.
- [23] G.E. Karniadakis, S.J. Sherwin, *Spectral/hp Element Methods for CFD*, second ed., Oxford University Press, 2005.
- [24] N. Kroll, ADGIMA – A European project on the development of adaptive higher-order variational methods for aerospace applications, in: 47th AIAA Aerospace Sciences Meeting, AIAA 2009-176, 2009.
- [25] J. Kurtz, L. Demkowicz, A fully automatic *hp*-adaptivity for elliptic PDEs in three dimensions, *Comput. Methods Appl. Eng.* 196 (2007) 3534–3545.
- [26] T. Leicht, R. Hartmann, Anisotropic mesh refinement for discontinuous Galerkin methods in two-dimensional aerodynamic flow simulations, *Int. J. Numer. Meth. Fluids* 56 (11) (2008) 2111–2138.
- [27] A. Loseille, A. Dervieux, F. Alauzet, Fully anisotropic goal-oriented mesh adaptation for 3D steady Euler equations, *J. Comput. Phys.* 229 (8) (2010) 2866–2897.
- [28] T. Richter, A posteriori error estimation and anisotropy detection with the dual-weighted residual method, *Int. J. Numer. Methods Fluids* 62 (1) (2009) 90–118.
- [29] O. Sahni, J. Müller, K.E. Jansen, M.S. Shepard, C.A. Taylor, Efficient anisotropic adaptive discretization of the cardiovascular system, *Comput. Methods Appl. Mech. Eng.* 195 (2006) 5634–5655.
- [30] S. Sun, M.F. Wheeler, Anisotropic and dynamic mesh adaption for discontinuous Galerkin methods applied to reactive transport, *Comput. Methods Appl. Mech. Eng.* 195 (2006) 3382–3405.
- [31] E.F. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics*, Springer, 1997.
- [32] D.A. Venditti, D.L. Darmofal, Anisotropic grid adaption for functional outputs: application to two-dimensional viscous flows, *J. Comput. Phys.* 187 (2003) 22–46.