Adaptive Discontinuous Galerkin Finite Element Methods for the Compressible Euler Equations

Ralf Hartmann

Institute of Applied Mathematics, University of Heidelberg, Im Neuenheimer Feld 293, D-69120 Heidelberg, Germany (e-mail: Ralf.Hartmann@iwr.uni-heidelberg.de).

and

Paul Houston

Department of Mathematics & Computer Science, University of Leicester, Leicester LE1 7RH, UK (e-mail: Paul.Houston@mcs.le.ac.uk).

In this paper a recently developed approach for the design of adaptive discontinuous Galerkin finite element approximations is applied to physically relevant problems arising in inviscid compressible fluid flows governed by the Euler equations of gas dynamics. In particular, we employ so–called weighted or Type I *a posteriori* error bounds to drive adaptive finite element algorithms for the estimation of the error measured in terms of general linear and nonlinear target functionals of the solution; typical examples considered here include the point evaluation of a component of the solution vector, and the drag and lift coefficients of a body immersed in an inviscid fluid. This general approach leads to the design of economical finite element meshes specifically tailored to the computation of the target functional of interest, as well as providing reliable and efficient error estimation. Indeed, the superiority of the proposed approach over standard mesh refinement algorithms which employ *ad hoc* error indicators will be illustrated by a series of numerical experiments; here, we consider transonic flow through a nozzle, as well as subsonic, transonic and supersonic flows around different airfoil geometries.

Key Words: Discontinuous Galerkin, *a posteriori* error analysis, adaptivity, compressible Euler equations

1. INTRODUCTION

One of the main challenges in the field of computational partial differential equations is the design and implementation of efficient adaptive finite element/finite volume methods which approximate the solution to complex multi-dimensional problems with guaranteed error control. While for problems arising in solid mechanics several software packages exist which are capable of achieving this goal, for fluid flow problems involving partial differential equations of hyperbolic/nearly–hyperbolic character, the situation is far less advanced. Indeed, the majority of adaptive CFD algorithms will refine or adjust the computational grid according to an *ad hoc* criterion, such as a large gradient in one of the field variables. Although this intuitive approach has had some success, it does not provide practical and sharp error estimation. An alternative to this heuristic approach is to derive reliable and efficient *a posteriori* error bounds which can successfully drive adaptive algorithms; see, for example, [2, 7, 8, 9, 10, 12, 13, 16, 18, 19, 20].

The purpose of this paper is to apply the recently developed approach of weighted *a posteriori* error estimation for discontinuous Galerkin finite element methods (see [9, 10, 12]) to the discretisation of physically relevant problems arising in inviscid compressible fluid flows governed by the Euler equations of gas dynamics; for related work, we refer to [2, 7, 8, 13, 16, 18, 19, 20], for example. Here, by employing a hyperbolic duality argument, weighted or Type I (cf. [14]) *a posteriori* error bounds for general functionals of the solution of practical relevance may be derived; typical examples include the drag and lift coefficients of a body immersed into an inviscid fluid, the local mean value of the field or its flux through the outflow boundary of the computational domain, and the pointwise evaluation of a component of the solution vector at a given point in the computational domain. In these error estimates the element–residuals are multiplied by local weights involving the solution of a certain dual problem. These weights provide invaluable information concerning the global transport of the error and the interaction of the error in different components of the solution. Indeed, the elimination of these weights *en route* to a Type II *a posteriori* error bound generally leads to an inefficient adaptive mesh refinement algorithm. In this article we consider the performance of weighted *a posteriori* error indicators for both interior and exterior inviscid compressible flows of practical interest. In particular, we demonstrate the superiority of the proposed approach over standard mesh refinement algorithms which employ *ad hoc* error indicators: weighted *a posteriori* error estimates lead to the design of economical finite element meshes specifically tailored to the computation of the target functional of interest, as well as providing reliable and efficient error estimation.

The outline of this paper is as follows. In order to highlight the key ideas pursued in this paper, in Section 2 we present an overview of duality–based *a posteriori* error estimation. After introducing in Section 3 the compressible Euler equations of gas dynamics, in Section 4 we formulate their discontinuous Galerkin finite element approximation. Then, in Section 5 we derive *a posteriori* error bounds for general linear and nonlinear functionals of the solution based on the approach outlined in Section 2; cf. [9, 10, 12]. The performance of the proposed Type I *a posteriori* error estimator is then studied in Section 6 through a series of numerical experiments; here, we consider transonic flow through a nozzle, as well as subsonic, transonic and supersonic flows around different airfoil geometries. In particular, we demonstrate the superiority of using our weighted *a posteriori* error indicators over traditional *ad hoc* error indicators which do not require the solution of an auxiliary or dual problem. Finally, in Section 7 we summarise the work presented in this paper and draw some conclusions.

2. A PARADIGM FOR A POSTERIORI ERROR ESTIMATION

In this section we present an overview of the general theoretical framework of duality–based *a posteriori* error estimation developed by C. Johnson and R. Rannacher and their collaborators. For a detailed discussion, we refer to the series of articles [2, 7, 8, 9, 13, 16, 19, 20], and the references cited therein.

Let X and Y be two Hilbert spaces. Further, we write $\mathcal{N}(\cdot,\cdot)$ to denote a semi-linear form (nonlinear in its first argument, but linear in its second), with derivative $\mathcal{N}'(\cdot;\cdot,\cdot)$. We suppose that u is the unique solution to the variational problem: find u in X such that

$$\mathcal{N}(u,v) = 0 \quad \forall v \in Y. \tag{1}$$

In order to construct a Galerkin approximation to this problem, we consider a sequence of finite–dimensional spaces $\{X_h\}$, parameterised by the positive discretisation parameter h; for the sake of simplicity we suppose that $X_h \subset X$ for each h. Simultaneously, consider a sequence of finite–dimensional spaces $\{Y_h\}$, with Y_h contained in Y for each h. For the purposes of this paper, X_h and Y_h can be thought of as finite element spaces consisting of piecewise polynomial functions on a partition, of granularity h, of the computational domain. The Galerkin approximation u_h of u is then sought in X_h as the solution of the finite–dimensional problem

$$\mathcal{N}(u_h,v_h) = 0 \quad \forall v_h \in Y_h. \tag{2}$$

For simplicity of presentation, we assume that X_h and Y_h are suitably chosen finite element spaces to ensure the existence of a unique solution u_h to (2).

In many problems of physical importance the quantity of interest is a target or error functional $J(\cdot)$ of the solution. Relevant examples include the mean flow across a line, a point value of the solution, or the drag and lift coefficients of a body immersed into an inviscid fluid. In order to obtain a computable *a posteriori* bound on the error between the true value of the functional $J(u)$ and the computed value $J(u_h)$, we begin by noting the Galerkin orthogonality of the discretisation (2):

$$\mathcal{N}(u,v_h) - \mathcal{N}(u_h,v_h) = 0 \quad \forall v_h \in Y_h. \tag{3}$$

This will be a key ingredient in the following *a posteriori* error analysis. Additionally, let us introduce some notation. Assuming that the functional of interest $J(\cdot)$ defined on X is differentiable, we write $\bar{J}(\cdot;\cdot)$ to denote the mean value linearisation of $J(\cdot)$ defined by

$$\bar{J}(u,u_h;u-u_h) = J(u) - J(u_h) = \int_0^1 J'[\theta u + (1-\theta)u_h](u-u_h)\,\mathrm{d}\theta, \tag{4}$$

where $J'[w](\cdot)$ denotes the functional or Fréchet derivative of $J(\cdot)$ evaluated at some w in X.

Analogously, we write $\mathcal{M}(u,u_h;\cdot,\cdot)$ to denote the mean–value linearisation of the semi–linear form $\mathcal{N}(\cdot,\cdot)$ given by

$$\begin{aligned}\mathcal{M}(u,u_h;u-u_h,v) &= \mathcal{N}(u,v) - \mathcal{N}(u_h,v)\\ &= \int_0^1 \mathcal{N}'[\theta u + (1-\theta)u_h](u-u_h,v)\,\mathrm{d}\theta\end{aligned} \tag{5}$$

for all v in Y. With this notation, we note that the Galerkin orthogonality property (3) may be expressed in the form

$$\mathcal{M}(u, u_h; u - u_h, v_h) = 0 \quad \forall v_h \in Y_h. \tag{6}$$

We now introduce the following *dual* or *adjoint* problem: find $z \in Y$ such that

$$\mathcal{M}(u, u_h; w, z) = \bar{J}(u, u_h; w) \quad \forall w \in X. \tag{7}$$

We assume that (7) possesses a unique solution. Clearly, the validity of this assumption depends on both the definition of $\mathcal{M}(u, u_h; \cdot, \cdot)$ and the choice of the functional under consideration. Important examples which are covered by our hypothesis are discussed in [10], [14] and [20].

For the proceeding error analysis, we must therefore *assume* that the dual problem (7) is well–posed. Under this assumption, employing the Galerkin orthogonality property (6) we deduce the following error representation formula:

$$\begin{aligned} J(u) - J(u_h) &= \bar{J}(u, u_h; u - u_h) \\ &= \mathcal{M}(u, u_h; u - u_h, z) \\ &= \mathcal{M}(u, u_h; u - u_h, z - z_h) \\ &= -\mathcal{N}(u_h, z - z_h) \end{aligned} \tag{8}$$

for all z_h in the finite element space Y_h. Let us now decompose the right–hand side of (8) as a summation of local error indicators η_κ over the elements κ in the computational mesh $\mathcal{T}_h$, i.e. we write

$$J(u) - J(u_h) = -\mathcal{N}(u_h, z - z_h) \equiv \sum_{\kappa \in \mathcal{T}_h} \eta_\kappa.$$

Thereby, by application of the triangle inequality, we deduce the following weighted or Type I *a posteriori* error bound.

THEOREM 2.1. *Let u and u_h denote the solutions of (1) and (2), respectively, and suppose that the dual problem (7) is well–posed. Then, the following Type I* a posteriori *error bound holds:*

$$|J(u) - J(u_h)| \leq \sum_{\kappa \in \mathcal{T}_h} |\eta_\kappa|. \tag{9}$$

The Type I *a posteriori* error bound (9) involves the multiplication of the finite element *residual* depending only on u_h with local weighting terms involving the difference between the dual solution z satisfying (7) and its projection/interpolant z_h onto the finite element space Y_h. These weights represent the sensitivity of the error in the target functional $J(\cdot)$ with respect to variations of the local element residuals. Since the solution to the dual problem is usually unknown analytically it may be desirable to eliminate z from (9); indeed, by employing the Cauchy–Schwarz inequality, together with standard results from approximation theory, the right–hand side of (9) may be bounded from above in terms of powers of the mesh function

h and Sobolev seminorms of z. For then, finally z may be completely eliminated from the *a posteriori* estimate by bounding norms of z by suitable norms of the data for the dual problem (7) by employing well–posedness results. The resulting Type II (cf. [14]) *a posteriori* error bound now only involves the computable finite element residual, the discretisation parameter h, interpolation constants and the stability factor for the dual problem. These type of bounds are in the spirit of the ones derived by C. Johnson and his co-workers.

However, as we shall see in Section 6, the elimination of the weighting terms involving the difference between the dual solution z and z_h may adversely affect the efficiency of the resulting adaptive algorithm leading to uneconomical mesh design, cf. [2, 9, 10, 12, 13, 19]. Thereby, we refrain from bounding the terms on the right–hand side of (9); in particular, we shall not attempt to eliminate the dual solution z from our *a posteriori* error bound by exploiting standard approximation results and the strong stability of the dual problem. Instead, the dual problem (7) will be approximated numerically as part of the error estimation process. However, the cost of this additional calculation is in general relatively cheap in comparison to the cost of determining the numerical approximation to the original (primal) system of *nonlinear* partial differential equations (1), since the dual problem (7) comprises of a system of *linear* partial differential equations. Finally, we note that in general the *a posteriori* error bound (9) depends on the true solution u to (1) through the linearisations performed in (4) and (5). Thereby, in order to render the right–hand side of (9) fully computable, in practice, $\bar{J}(u, u_h; \cdot)$ and $\mathcal{M}(u, u_h; \cdot, \cdot)$ are approximated by $\bar{J}(u_h, u_h; \cdot)$ and $\mathcal{M}(u_h, u_h; \cdot, \cdot)$, respectively. Quantifying the size of these approximations is a rather delicate task and depends on the particular problem under consideration.

3. MODEL PROBLEM

We consider the two–dimensional steady state compressible Euler equations of gas dynamics. Writing ρ, (u, v), p and E to denote the density, Cartesian velocity, pressure and total energy per unit mass, respectively, the equations of motion are given by

$$\mathrm{div}\mathcal{F}(\mathbf{u}) \equiv \sum_{i=1}^{2} \frac{\partial}{\partial x_i} \mathbf{f}_i(\mathbf{u}) = 0 \quad \text{in } \Omega, \tag{10}$$

where Ω is an open bounded domain in $\mathbb{R}^2$. Here, the vector of conservative variables $\mathbf{u}$ and the fluxes $\mathbf{f}_i$, $i = 1, 2$, are defined by

$$\mathbf{u} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix}, \quad \mathbf{f}_1 = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ \rho H u \end{bmatrix} \quad \text{and} \quad \mathbf{f}_2 = \begin{bmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ \rho H v \end{bmatrix},$$

respectively. Additionally, H is the total enthalpy defined by

$$H = E + \frac{p}{\rho}.$$

The equation of state of an ideal gas is given by

$$p = (\gamma - 1)\rho \left(E - \frac{1}{2}(u^2 + v^2) \right), \tag{11}$$

where γ is the ratio of specific heats which, for dry air, is $\gamma = 1.4$.

If the analytical solution $\mathbf{u}$ is sufficiently smooth, then (10) may be written in the following non-conservative quasi-linear form

$$\sum_{i=1}^{2} A_i(\mathbf{u}) \frac{\partial \mathbf{u}}{\partial x_i} = 0 \quad \text{in } \Omega,$$

where

$$A_i(\mathbf{u}) := \nabla_{\mathbf{u}} \mathbf{f}_i(\mathbf{u}) \tag{12}$$

denote the Jacobi matrices of the fluxes $\mathbf{f}_i(\mathbf{u})$, respectively, $i = 1, 2$. Thereby,

$$A_1(\mathbf{u}) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{\gamma-3}{2}u^2 + \frac{\gamma-1}{2}v^2 & (3-\gamma)u & (1-\gamma)v & \gamma - 1 \\ -uv & v & u & 0 \\ (\gamma-1)u(u^2+v^2) - \gamma u E & \gamma E - \frac{\gamma-1}{2}(v^2 + 3u^2) & (1-\gamma)uv & \gamma u \end{bmatrix}$$

and

$$A_2(\mathbf{u}) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ -uv & v & u & 0 \\ \frac{\gamma-3}{2}v^2 + \frac{\gamma-1}{2}u^2 & (1-\gamma)u & (3-\gamma)v & \gamma - 1 \\ (\gamma-1)v(u^2+v^2) - \gamma v E & (1-\gamma)uv & \gamma E - \frac{\gamma-1}{2}(u^2 + 3v^2) & \gamma v \end{bmatrix}.$$

Given that $\Omega \subset \mathbb{R}^2$ is a bounded region, with boundary Γ, the system of conservation laws (10) must be supplemented by appropriate boundary conditions; for example at inflow/outflow boundaries, we require that

$$A^-(\mathbf{u}, \mathbf{n})\,(\mathbf{u} - \mathbf{g}) = \mathbf{0}, \tag{13}$$

where $\mathbf{n}$ denotes the unit outward normal vector to the boundary Γ and $\mathbf{g}$ is a (given) real–valued vector function. Here, $A^-(\mathbf{u}, \mathbf{n})$ denotes the negative part of

$$A(\mathbf{u}, \mathbf{n}) := \sum_{i=1}^{2} \mathbf{n}_i A_i(\mathbf{u});$$

i.e.

$$A^-(\mathbf{u}, \mathbf{n}) = R\Lambda^- L,$$

where L and R denote the 4×4 matrices of left and right eigenvectors of $A(\mathbf{u}, \mathbf{n})$, respectively, and $\Lambda^- = \mathrm{diag}(\min(\lambda_i, 0))$ the 4×4 diagonal matrix of the negative eigenvalues of $A(\mathbf{u}, \mathbf{n})$.

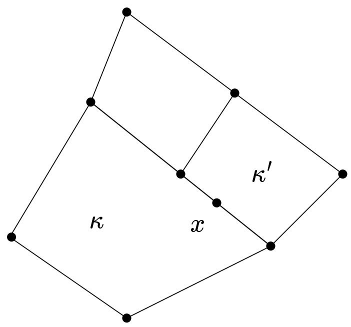


FIG. 1. A point x such that $x \in \partial\kappa$ and $x \in \partial\kappa'$.

4. DISCONTINUOUS GALERKIN METHOD

We begin by introducing the necessary notation. Suppose that $k \geq 1$, and $\mathcal{T}_h$ is a regular or k–irregular subdivision of Ω into open element domains κ such that $\bar{\Omega} = \cup_{\kappa\in\mathcal{T}_h}\bar{\kappa}$. Thus an edge of each element κ in $\mathcal{T}_h$ is allowed to contain at most k hanging (irregular) nodes; when $k = 1$, the hanging node is typically the barycenter of the edge. Here, h denotes the piecewise constant mesh function defined by $h|_\kappa \equiv h_\kappa = \mathrm{diam}(\kappa)$ for all $\kappa \in \mathcal{T}_h$. We shall suppose that the family of subdivisions $\mathcal{T}_h$ is shape regular and that each $\kappa \in \mathcal{T}_h$ is a smooth bijective image of a fixed reference element $\hat{\kappa}$, that is, $\kappa = F_\kappa(\hat{\kappa})$ for all $\kappa \in \mathcal{T}_h$, where $\hat{\kappa}$ is either the open unit simplex (triangle in 2 dimensions) or the open unit hypercube $(-1,1)^2$ in $\mathbb{R}^2$. On the reference element $\hat{\kappa}$ we define spaces of polynomials of degree $p \geq 0$ as follows:

$$\begin{aligned} \mathcal{Q}_p &= \mathrm{span}\left\{\hat{\mathbf{x}}^\alpha : 0 \leq \alpha_i \leq p,\ 0 \leq i \leq 2\right\}, \\ \mathcal{P}_p &= \mathrm{span}\left\{\hat{\mathbf{x}}^\alpha : 0 \leq |\alpha| \leq p\right\}. \end{aligned}$$

We now introduce the finite element space $S_{h,p}$ consisting of discontinuous vector–(of dimension 4)–valued polynomial functions of degree $p \geq 0$, defined by

$$\begin{aligned} S_{h,p} = \{\mathbf{v} \ \in \ [L_2(\Omega)]^4 : \mathbf{v}|_\kappa \circ F_\kappa \in [\mathcal{Q}_p]^4\,, \text{ if } \hat{\kappa} = F_\kappa^{-1}(\kappa) \text{ is the unit hypercube,} \\ \text{and } \mathbf{v}|_\kappa \circ F_\kappa \in [\mathcal{P}_p]^4\,, \text{ if } \hat{\kappa} = F_\kappa^{-1}(\kappa) \text{ is the unit simplex; } \kappa \in \mathcal{T}_h\}\ . \end{aligned}$$

For each $\kappa \in \mathcal{T}_h$ and any $v \in H^1(\kappa)$ we denote by v^+ the interior trace of v on $\partial\kappa$ (the trace taken from within κ). Now consider an element κ such that the set $\partial\kappa\backslash\Gamma$ is nonempty; then for each $x \in \partial\kappa\backslash\Gamma$ (with the exception of a set of 1–dimensional measure zero) there exists a unique element κ', depending on the choice of x, such that $x \in \partial\kappa'$. This is illustrated in Figure 1. Now suppose that $v \in H^1(\kappa)$ for each $\kappa \in \mathcal{T}_h$. If $\partial\kappa\backslash\Gamma$ is nonempty for some $\kappa \in \mathcal{T}_h$, then we can also define the outer trace v^- of v on $\partial\kappa\backslash\Gamma$ relative to κ as the inner trace v^+ relative to those elements κ' for which $\partial\kappa'$ has intersection with $\partial\kappa\backslash\Gamma$ of positive 1–dimensional measure.

To formulate the discontinuous Galerkin method, we first introduce a weak formulation of (10). To this end, we multiply the conservation law (10) by an arbitrary smooth function $\mathbf{v}$ and integrate by parts over an element κ in the mesh $\mathcal{T}_h$; thereby, we get

$$-\int_\kappa \mathcal{F}(\mathbf{u}) \cdot \nabla \mathbf{v}\, \mathrm{d}\mathbf{x} + \int_{\partial\kappa} \mathcal{F}(\mathbf{u}^+) \cdot \mathbf{n}_\kappa\, \mathbf{v}^+\, \mathrm{d}s = 0, \tag{14}$$

where $\mathbf{n}_\kappa$ denotes the unit outward normal vector to $\partial\kappa$.

To discretise (14), we replace the analytical solution $\mathbf{u}$ by the Galerkin finite element approximation $\mathbf{u}_h$ and the test function $\mathbf{v}$ by $\mathbf{v}_h$, where $\mathbf{u}_h$ and $\mathbf{v}_h$ both belong to the finite element space $S_{h,p}$. In addition, since the numerical solution $\mathbf{u}_h$ is discontinuous between element interfaces, we must replace the flux $\mathcal{F}(\mathbf{u}^+) \cdot \mathbf{n}_\kappa$ by a *numerical flux* function $\mathcal{H}(\mathbf{u}_h^+, \mathbf{u}_h^-, \mathbf{n}_\kappa)$, which depends on both the inner– and outer–trace of $\mathbf{u}_h$ on $\partial\kappa$, $\kappa \in \mathcal{T}_h$, and the unit outward normal vector $\mathbf{n}_\kappa$ to $\partial\kappa$. Furthermore, to suppress the generation of non-physical oscillations, we add an artificial–viscosity/shock–capturing term which depends on both the mesh function h and the finite element residual. Thereby, summing over the elements κ in the mesh $\mathcal{T}_h$, yields the following discontinuous Galerkin finite element discretisation of (10): find $\mathbf{u}_h \in S_{h,p}$ such that

$$\begin{aligned}\mathcal{N}(\mathbf{u}_h, \mathbf{v}_h) \equiv \sum_{\kappa\in\mathcal{T}_h} \Big\{ -\int_\kappa \mathcal{F}(\mathbf{u}_h) \cdot \nabla \mathbf{v}_h\, \mathrm{d}\mathbf{x} \;+\; \int_{\partial\kappa} \mathcal{H}(\mathbf{u}_h^+, \mathbf{u}_h^-, \mathbf{n}_\kappa)\, \mathbf{v}_h^+\, \mathrm{d}s \\ + \int_\kappa \varepsilon \nabla \mathbf{u}_h \cdot \nabla \mathbf{v}_h\, \mathrm{d}\mathbf{x} \Big\} = 0 \quad \forall \mathbf{v}_h \in S_{h,p}.\end{aligned} \tag{15}$$

Here, ε denotes the artificial viscosity coefficient matrix defined by

$$\varepsilon = C_\varepsilon\, h^{2-\beta}\, |\mathrm{div}\mathcal{F}(\mathbf{u}_h)|\, I, \tag{16}$$

where C_ε and $0 < \beta < 1/2$ are positive constants and I denotes the identity matrix in $\mathbb{R}^2$. The definition of ε in (16) represents a slight modification of the artificial viscosity model introduced and analysed by Jaffre *et al.* [15]; for related work, we refer to [1, 4, 5, 6], for example.

Remark. For elements $\kappa \in \mathcal{T}_h$ whose boundary intersect that of the computational domain Ω, we replace $\mathbf{u}_h^-$ by appropriate boundary conditions on the portion of $\partial\kappa$ for which $\partial\kappa \cap \Gamma \neq \emptyset$. At inflow/outflow boundaries, the state $\mathbf{u}_h^-$ is determined via a local characteristic decomposition of the incoming and outgoing waves in the direction normal to the boundary, cf. (13). At solid walls, a symmetry technique is employed; here, $\mathbf{u}_h^-$ has the same density, total energy and *tangential* velocity as $\mathbf{u}_h^+$, but $\mathbf{u}_h^-$ is assigned the opposite *normal* velocity to $\mathbf{u}_h^+$, cf. [1], for example.

We note that the replacement of the flux $\mathcal{F}(\mathbf{u}^+)\cdot\mathbf{n}_\kappa$ by the numerical flux function $\mathcal{H}(\mathbf{u}_h^+, \mathbf{u}_h^-, \mathbf{n}_\kappa)$ on the boundary of element κ, κ in $\mathcal{T}_h$, corresponds to the weak imposition of the boundary data, cf. [1]. Furthermore, we emphasise that the choice of the numerical flux function is *independent* of the finite element space employed. Indeed, the numerical flux $\mathcal{H}(\cdot,\cdot,\cdot)$ may be chosen to be any two–point monotone Lipschitz function which satisfies the following two conditions:

(i) $\mathcal{H}(\cdot,\cdot,\cdot)|_{\partial\kappa}$ is consistent with the flux $\mathcal{F}(\cdot)\cdot\mathbf{n}_\kappa$ for each κ in $\mathcal{T}_h$; i.e.

$$\mathcal{H}(\mathbf{u},\mathbf{u},\mathbf{n}_\kappa)|_{\partial\kappa} = \mathcal{F}(\mathbf{u})\cdot\mathbf{n}_\kappa \quad \forall\kappa\in\mathcal{T}_h;$$

(ii) $\mathcal{H}(\cdot,\cdot,\cdot)$ is conservative, i.e. given any two neighbouring elements κ and κ' from the finite element partition $\mathcal{T}_h$, at each point $\mathbf{x}\in\partial\kappa\cap\partial\kappa'\neq\emptyset$, noting that $\mathbf{n}_{\kappa'}=-\mathbf{n}_\kappa$, we have that

$$\mathcal{H}(\mathbf{v},\mathbf{w},\mathbf{n}_\kappa) = -\mathcal{H}(\mathbf{w},\mathbf{v},-\mathbf{n}_\kappa).$$

There are several numerical flux functions satisfying these conditions, such as the Godunov flux, the Engquist–Osher flux, the Lax–Friedrichs flux, the Roe flux and the Vijayasundaram flux, for example; cf. Kröner [17] and Toro [21], for example, and the references cited therein.

Assuming that the analytical solution $\mathbf{u}$ to (10) is sufficiently smooth, then the Galerkin orthogonality property of the finite element method (15) is expressed as follows:

$$\mathcal{N}(\mathbf{u},\mathbf{v}_h) - \mathcal{N}(\mathbf{u}_h,\mathbf{v}_h) = 0 \quad \forall\mathbf{v}_h\in S_{h,p}; \tag{17}$$

here, we have employed the consistency condition (i) of the numerical flux function $\mathcal{H}(\cdot,\cdot,\cdot)$ and the consistency of the artificial–viscosity terms.

5. A POSTERIORI ERROR ANALYSIS

In this section, we shall be interested in controlling the error of the numerical solution measured in terms of a given target functional $J(\cdot)$, based on the general framework of *a posteriori* error estimation developed in Section 2; for a more detailed account, we refer to the article [10].

To this end, recalling the notation introduced in Section 2, we define the following *dual* or *adjoint* problem: find $\mathbf{z}\in V$ such that

$$\mathcal{M}(\mathbf{u},\mathbf{u}_h;\mathbf{w},\mathbf{z}) = \bar{J}(\mathbf{u},\mathbf{u}_h;\mathbf{w}) \quad \forall\mathbf{w}\in V, \tag{18}$$

where V is some suitably chosen function space such that $S_{h,p}\subset V$. Here, we write $\mathcal{M}(\mathbf{u},\mathbf{u}_h;\cdot,\cdot)$ to denote the mean–value linearisation of the semi–linear form $\mathcal{N}(\cdot,\cdot)$ introduced in (5), where $\mathcal{N}(\cdot,\cdot)$ is defined in Section 4, cf. equation (15). We remark that the linearisation defined in equation (5) is only a *formal* calculation, in the sense that $\mathcal{N}'[\mathbf{w}](\cdot,\mathbf{v})$ may not in general exist. Instead, a suitable approximation to $\mathcal{N}'[\mathbf{w}](\cdot,\mathbf{v})$ in the direction $\mathbf{u}-\mathbf{u}_h$ must be determined, for example, by computing appropriate finite difference quotients of $\mathcal{N}(\cdot,\mathbf{v})$, cf. [10]. For the purposes of the proceeding analysis, we *assume* that the linearisation performed in (5), with $\mathcal{N}(\cdot,\cdot)$ defined by (15), is well–defined. Moreover, as in Section 2, we assume that (18) is a well–posed problem. Under these assumptions, we have the following general result.

THEOREM 5.1. *Let $\mathbf{u}$ and $\mathbf{u}_h$ denote the solutions of (10) and (15), respectively, and suppose that the dual problem (18) is well–posed. Then, the following error representation formula holds:*

$$J(\mathbf{u}) - J(\mathbf{u}_h) = \mathcal{E}_\Omega(\mathbf{u}, \mathbf{u}_h, h, \mathbf{z} - \mathbf{z}_h) \equiv \sum_{\kappa \in \mathcal{T}_h} \eta_\kappa, \tag{19}$$

where

$$\eta_\kappa = \int_\kappa R_h\,(\mathbf{z} - \mathbf{z}_h)\,d\mathbf{x} + \int_{\partial\kappa} r_h\,(\mathbf{z} - \mathbf{z}_h)^+\,ds - \int_\kappa \varepsilon \nabla \mathbf{u}_h \cdot \nabla(\mathbf{z} - \mathbf{z}_h)\,d\mathbf{x} \tag{20}$$

for all $\mathbf{z}_h$ in $S_{h,p}$. Here, R_h and r_h denote the internal and boundary finite element residuals, respectively, defined on $\kappa \in \mathcal{T}_h$ by

$$R_h|_\kappa = -div\mathcal{F}(\mathbf{u}_h) \quad and \quad r_h|_\kappa = \mathcal{F}(\mathbf{u}_h^+) \cdot \mathbf{n}_\kappa - \mathcal{H}(\mathbf{u}_h^+, \mathbf{u}_h^-, \mathbf{n}_\kappa), \tag{21}$$

respectively.

Proof. Choosing $\mathbf{w} = \mathbf{u} - \mathbf{u}_h$ in (18), recalling the linearisation performed in (4), and exploiting the Galerkin orthogonality property (17), we deduce from equation (8) that

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

for all $\mathbf{z}_h$ in the finite element space $S_{h,p}$. Employing the divergence theorem, we have that

$$\begin{aligned} J(\mathbf{u}) - J(\mathbf{u}_h) = \sum_{\kappa \in \mathcal{T}_h} \quad & \Big\{ -\int_\kappa \mathrm{div}\mathcal{F}(\mathbf{u}_h)(\mathbf{z} - \mathbf{z}_h)\,\mathrm{d}\mathbf{x} \\ & + \int_{\partial\kappa} \left(\mathcal{F}(\mathbf{u}_h^+) \cdot \mathbf{n}_\kappa - \mathcal{H}(\mathbf{u}_h^+, \mathbf{u}_h^-, \mathbf{n}_\kappa)\right)\,(\mathbf{z} - \mathbf{z}_h)^+\,\mathrm{d}s \\ & - \int_\kappa \varepsilon \nabla \mathbf{u}_h \cdot \nabla(\mathbf{z} - \mathbf{z}_h)\,\mathrm{d}\mathbf{x} \Big\}. \end{aligned} \tag{22}$$

Using the definition of the internal and boundary residuals in (21), gives the desired result. ■

Based on the general error representation formula derived in Theorem 5.1, *a posteriori* error estimates bounding the error in the computed functional $J(\cdot)$ may be deduced. Here, we confine ourselves to considering a Type I *a posteriori* error bound; it is a straightforward consequence of the error representation formula stated in the previous theorem, cf. Theorem 2.1.

COROLLARY 5.1. *Under the assumptions of Theorem 5.1, the following Type I* a posteriori *error bound holds:*

$$|J(\mathbf{u}) - J(\mathbf{u}_h)| \leq \mathcal{E}_{|\Omega|}(\mathbf{u}, \mathbf{u}_h, h, \mathbf{z} - \mathbf{z}_h) \equiv \sum_{\kappa \in \mathcal{T}_h} |\eta_\kappa|\,, \tag{23}$$

where η_κ is defined as in (20).

Proof. The *a posteriori* error bound (23) follows trivially from (19) by application of the triangle inequality. ■

We end this section by noting that both the error representation formula (19) and the resulting Type I *a posteriori* error bound (23) depend on the unknown analytical solution to both the primal and dual problems, $\mathbf{u}$ and $\mathbf{z}$, respectively. Thus, in order to render these error estimates computable, both $\mathbf{u}$ and $\mathbf{z}$ must be replaced by suitable approximations which do not adversely affect the reliability of the Type I *a posteriori* error bound (23), in the sense that the right–hand side of (23) does not under estimate the true error in the computed functional $J(\cdot)$. To this end, the linearisations leading to $\mathcal{M}(\mathbf{u}, \mathbf{u}_h; \cdot, \cdot)$ and $\bar{J}(\mathbf{u}, \mathbf{u}_h; \cdot)$ are performed about $\mathbf{u}_h$ and the dual solution $\mathbf{z}$ is replaced by a discontinuous Galerkin approximation $\hat{\mathbf{z}}$ computed on the same mesh $\mathcal{T}_h$ used for $\mathbf{u}_h$, but with a higher degree polynomial, i.e., $\hat{\mathbf{z}} \in S_{h,\hat{p}}$, where $\hat{p} > p$.

Notwithstanding these approximations, we shall show through numerical experimentation in Section 6, that the reliability of the Type I *a posteriori* error bound (23) is not compromised, in the sense that

$$\hat{\mathcal{E}}_{|\Omega|} \equiv \mathcal{E}_{|\Omega|}(\mathbf{u}_h, \mathbf{u}_h, h, \hat{\mathbf{z}}_h - \mathbf{z}_h) = \sum_{\kappa \in \mathcal{T}_h} |\hat{\eta}_\kappa|, \tag{24}$$

where $\hat{\eta}_\kappa$ is defined as in (20) with $\mathbf{z}$ replaced by $\hat{\mathbf{z}}$, remains an upper bound on the true error in the target functional $J(\cdot)$. In particular, we shall show that the ratio of the approximate error representation formula

$$\hat{\mathcal{E}}_{\Omega} \equiv \mathcal{E}_{\Omega}(\mathbf{u}_h, \mathbf{u}_h, h, \hat{\mathbf{z}}_h - \mathbf{z}_h) = \sum_{\kappa \in \mathcal{T}_h} \hat{\eta}_\kappa \tag{25}$$

and the true value $J(\mathbf{u}) - J(\mathbf{u}_h)$ is extremely close to one; see [9, 12, 13, 18], for related work.

6. NUMERICAL EXPERIMENTS

In this section we present some numerical examples to highlight the advantages of designing an adaptive finite element algorithm based on the weighted error indicators $|\hat{\eta}_\kappa|$ in comparison with traditional refinement strategies which do not require the solution of the dual problem (18). To this end, we consider the *ad hoc* error indicator

$$\eta_\kappa^{\text{adhoc}} = \|hR_h\|_{L_2(\kappa)} + \|h^{1/2} r_h\|_{L_2(\partial\kappa)} \,,$$

which stems from a Type II *a posteriori* error analysis, cf. [14]. Throughout this section, we employ the local Lax-Friedrichs flux and set $p = 1$ and $\hat{p} = 2$; additionally, for problems with curved boundary Γ, we approximate Γ by a continuous piecewise quadratic polynomial function. Finally, we note that all computations are performed with the fixed fraction mesh refinement algorithm with refinement and derefinement fractions set to 20% and 10%, respectively.

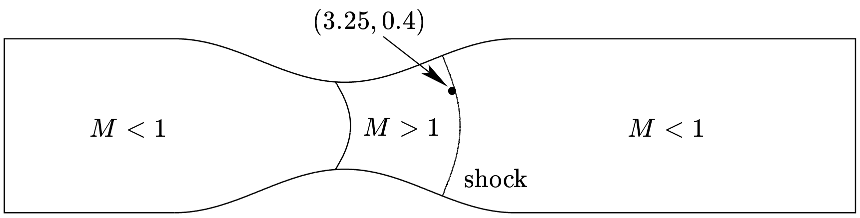


FIG. 2. Geometry for the converging–diverging nozzle problem; here M denotes the Mach number.

6.1. Transonic flow in a converging–diverging nozzle

In this first example we consider the transonic flow in a converging–diverging nozzle. The computational domain is shown in Figure 2; here, the upper and lower walls of the nozzle are given by the function $g^{\pm}$, respectively, where

$$g^{\pm}(x) = \begin{cases} \pm 1, & -2 \leq x \leq 0, \\ \pm(\cos(\pi\, x/2) + 3)/4, & 0 \leq x \leq 4, \\ \pm 1, & 4 \leq x \leq 8. \end{cases}$$

At the (subsonic) inflow, corresponding to $x = -2$, $-1 \leq y \leq 1$, we set the density $\rho = 1$, the pressure $p = 1$ and the Mach number $M = 0.3$; along $x = 8$, $-1 \leq y \leq 1$, we simply prescribe an outflow pressure condition $p = 2/3$, cf. Brooksbanks [3]. The structure of the solution to this problem is as follows: the subsonic flow enters the computational domain from the left–hand side, and accelerates to become supersonic as it enters the converging part of the nozzle; as the nozzle expands, the flow slows to form a shock before becoming subsonic again, cf. Figure 2.

Here, we take the functional of interest to be the value of the density in the supersonic region of the domain, just before the shock forms. More precisely, we choose $J(\cdot)$ to be the linear functional given by

$$J(\mathbf{u}) = \rho(3.25, 0.4);$$

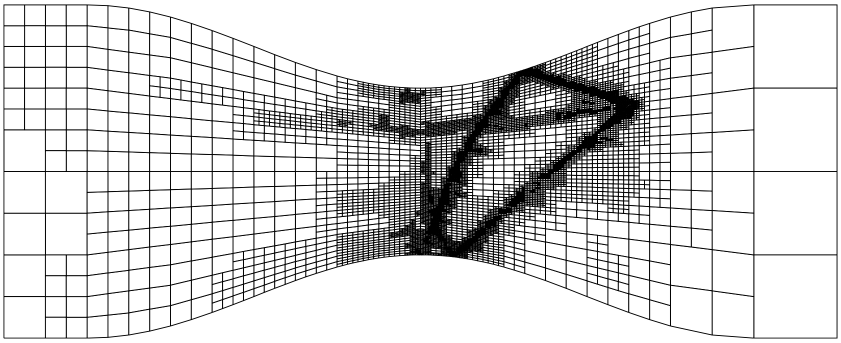
thereby, on the basis of a fine grid computation, the true value of the functional is given by $J(\mathbf{u}) = 0.249695$.

In Table 1, we demonstrate the performance of the adaptive algorithm with $C_\varepsilon = 2/25$ and $\beta = 1/10$. Here, we show the number of elements and degrees of freedom (DOF) in $S_{h,1}$, the true error in the functional $J(\mathbf{u}) - J(\mathbf{u}_h)$, the computed error representation formula (25), the (approximate) *a posteriori* error bound (24) and their respective effectivity indices θ_1 and θ_2. We see that initially on very coarse meshes the quality of the computed error representation formula $\hat{\mathcal{E}}_\Omega$ is very poor, in the sense that $\theta_1 = \hat{\mathcal{E}}_\Omega/(J(\mathbf{u}) - J(\mathbf{u}_h))$ is not close to one; however, as the mesh is refined the effectivity index θ_1 approaches unity. Furthermore, we see that the Type I *a posteriori* error bound $\hat{\mathcal{E}}_{|\Omega|}$ overestimates the true error in the computed functional by about an order of magnitude. We recall that $\mathcal{E}_{|\Omega|}$ was derived from the error representation formula by simply employing the triangle inequality; thereby, it is clear that any further bounding of $\mathcal{E}_{|\Omega|}$ performed *en route* to deriving a Type II *a posteriori* error estimate will lead to a loss of sharpness of the computed error

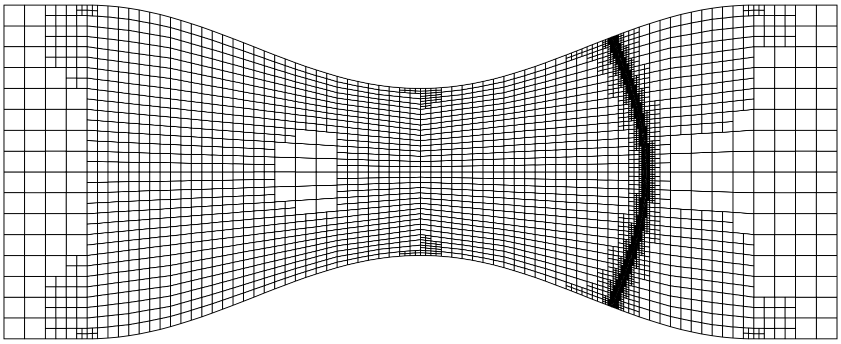
TABLE 1

Adaptive algorithm for the transonic nozzle problem.

Elements	DOF	$J(\mathbf{u}) - J(\mathbf{u}_h)$	$\sum_\kappa \hat{\eta}_\kappa$	θ_1	$\sum_\kappa \lvert\hat{\eta}_\kappa\rvert$	θ_2
80	1280	-2.357e-01	-1.667e-02	0.07	7.201e-02	0.31
143	2288	-1.997e-01	-5.073e-02	0.25	2.101e-01	1.05
224	3584	-5.983e-02	-7.678e-02	1.28	1.780e-01	2.98
350	5600	3.754e-02	4.667e-02	1.24	9.503e-02	2.53
590	9440	2.954e-02	1.772e-02	0.60	9.194e-02	3.11
926	14816	1.663e-02	1.651e-02	0.99	5.547e-02	3.34
1478	23648	-1.851e-03	-2.128e-03	1.15	1.987e-02	10.73
2462	39392	2.256e-05	1.176e-05	0.52	2.540e-03	112.58
4088	65408	2.466e-05	2.492e-05	1.01	1.070e-03	43.41
6719	107504	1.136e-05	1.258e-05	1.11	6.287e-04	55.36
10805	172880	6.947e-06	7.169e-06	1.03	3.538e-04	50.92
17588	281408	3.102e-06	3.102e-06	1.00	2.112e-04	68.09



(a)



(b)

FIG. 3. Transonic nozzle problem: (a) Mesh constructed using dual error indicator with 10805 elements and 172880 degrees of freedom ($|J(\mathbf{u}) - J(\mathbf{u}_h)| = 6.947 \times 10^{-6}$); (b) Mesh constructed using *ad hoc* error indicator with 21353 elements and 341648 degrees of freedom ($|J(\mathbf{u}) - J(\mathbf{u}_h)| = 2.841 \times 10^{-5}$).

bound in the sense that the size of the resulting effectivity indices will be even larger.

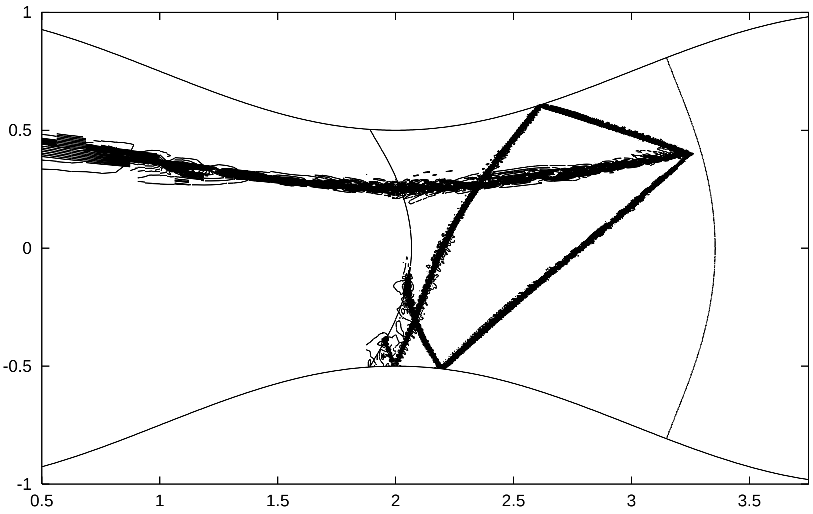


FIG. 4. Transonic nozzle problem: first component $\hat{\mathbf{z}}_1$ of the dual solution for point evaluation of the density.

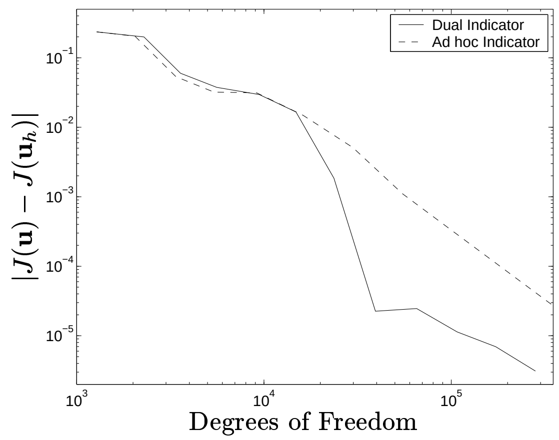


FIG. 5. Transonic nozzle problem. Convergence of $|J(\mathbf{u}) - J(\mathbf{u}_h)|$ using the dual and *ad hoc* error indicators.

In Figure 3 we show the meshes generated using both the weighted error indicator $|\hat{\eta}_\kappa|$ and the *ad hoc* error indicator $\eta_\kappa^{\mathrm{adhoc}}$. From Figure 3(a), we see that there is virtually no refinement in the region of the computational domain where the shock is located when $|\hat{\eta}_\kappa|$ is employed. Indeed, most of the mesh refinement is concentrated in a series of one–dimensional lines upstream of the point of interest. This is due to the presence of the weighting terms involving the difference between the (approximated) dual solution $\hat{\mathbf{z}}$ and $\mathbf{z}_h$ which multiply the computable residual terms involving the numerical solution $\mathbf{u}_h$ in the definition of the local error indicator $|\hat{\eta}_\kappa|$, cf. (20) with $\mathbf{z}$ replaced by $\hat{\mathbf{z}}$. Indeed, from Figure 4, we see that the dual solution

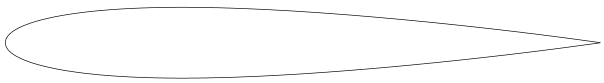


FIG. 6. Profile of the NACA0012 airfoil.

consists of a series of three 'spikes' originating from the point of interest; here, we have only shown $\hat{\mathbf{z}}_1$, though the other components of the dual solution have a similar structure. Within the supersonic region of the computational domain, these three spikes are transported upstream along the three characteristics corresponding to the three eigenvalues $\bar{v}$ and $\bar{v} \pm c$, where $\bar{v} = \sqrt{u^2 + v^2}$ denotes the velocity of the gas and $c = \sqrt{\gamma p/\rho}$ the speed of sound. We see that the middle characteristic $\bar{v}$, corresponding to mass transport, is carried upstream through the subsonic region to the inflow boundary located at $x = -2$. In contrast, the two characteristics corresponding to information transport due to sound waves are first reflected off the walls of the nozzle in the supersonic region, but are then prevented from entering the subsonic region where these characteristics become complex. This corresponds to a change in type in the underlying system of partial differential equations; here the steady compressible Euler equations change from hyperbolic type in the supersonic region of the computational domain to elliptic type in the subsonic region. In contrast, we see that when these weighting terms are neglected, then the mesh produced using the *ad hoc* error indicator is largely concentrated in the vicinity of the shock with some further, almost uniform refinement within the nozzle itself, cf. Figure 3(b).

Finally, in Figure 5, we compare the true error in the computed functional $J(\cdot)$ using the two mesh refinement strategies. Here, we clearly observe the superiority of the weighted *a posteriori* error indicators; on the final mesh the true error in the linear functional is over an order of magnitude smaller than $|J(\mathbf{u}) - J(\mathbf{u}_h)|$ computed on the sequence of meshes produced using $\eta_\kappa^{\mathrm{adhoc}}$. This clearly indicates that a good numerical resolution of the shock in this example is irrelevant for the accurate approximation of the functional of interest.

6.2. Subsonic flow around a NACA0012 airfoil

In this second example, we consider the subsonic flow around a NACA0012 airfoil; here, the upper and lower surfaces of the airfoil geometry are specified by the function $g^{\pm}$, respectively, where

$$g^{\pm}(s) = \pm 5 \times 0.12 \times (0.2969 s^{1/2} - 0.126 s - 0.3516 s^2 + 0.2843 s^3 - 0.1015 s^4),$$

cf. Figure 6. We note that the chord length l of the airfoil is $l \approx 1.00893$; thereby, we write $\hat{g}$ to denote the rescaling of g to yield an airfoil of unit (chord) length. The computational domain Ω is now defined to be an annulus with inner boundary $\hat{g}$ and outer boundary consisting of a circle of radius 10 units. On the outer boundary, we prescribe a Mach 0.5 flow at a zero angle of attack, with farfield density $\rho = 1$ and pressure $p = 1$. The solution to this problem consists of a strictly subsonic flow, symmetric about the x–axis, cf. Figure 7. Given that this flow is smooth, no artificial viscosity is required; thereby, we set $C_\varepsilon = 0$.

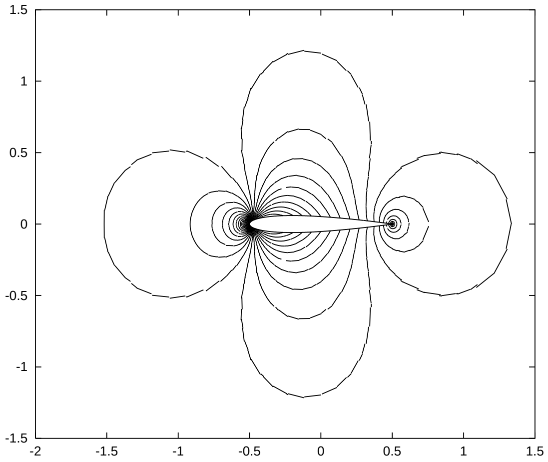


FIG. 7. Pressure isolines for the subsonic NACA0012 flow.

In this section we shall be interested in computing the drag and lift coefficients defined by

$$J_{\mathrm{drag}}(\mathbf{u}) = \frac{2}{\bar{l}\bar{\rho}|\bar{\mathbf{v}}|^2}\int_S (\psi_d \cdot \mathbf{n})\, p \,\mathrm{d}s, \qquad J_{\mathrm{lift}}(\mathbf{u}) = \frac{2}{\bar{l}\bar{\rho}|\bar{\mathbf{v}}|^2}\int_S (\psi_l \cdot \mathbf{n})\, p \,\mathrm{d}s,$$

respectively. Here, S denotes the surface of the airfoil $\hat{g}$, $\bar{l}$ its chord length (equal to one), $\bar{\mathbf{v}}$ and $\bar{\rho}$ denote the reference (or free-stream) velocity and density, respectively, and

$$\psi_d = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad \psi_l = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$

where α denotes the angle of attack. We note that since the pressure p is derived from the conserved variables $(\rho, \rho u, \rho v, \rho E)$ using the equation of state (11), both target functionals $J_{\mathrm{drag}}(\cdot)$ and $J_{\mathrm{lift}}(\cdot)$ are nonlinear. Given that the angle of attack $\alpha = 0$, the true value of both target functionals $J_{\mathrm{drag}}(\cdot)$ and $J_{\mathrm{lift}}(\cdot)$ is zero.

Let us first consider the case of estimating the drag on the surface of the airfoil, i.e. when $J(\cdot) \equiv J_{\mathrm{drag}}(\cdot)$. To this end, in Table 2 we show the performance of our adaptive algorithm; here we see that the quality of the computed error representation formula is extremely good, with $\theta_1 \approx 1$ even on relatively coarse meshes. Furthermore, the Type I *a posteriori* error bound (24) is sharper for this smooth problem; here, $\hat{\mathcal{E}}_{|\Omega|}$ overestimates the true error in the computed functional by about a factor 2–3.

The meshes produced using both our weighted error indicator $|\hat{\eta}_\kappa|$ and the *ad hoc* error indicator $\eta_\kappa^{\mathrm{adhoc}}$ are shown in Figure 8. Here, we see that both meshes are fairly similar in character, in the sense that most of the refinement is concentrated in the neighbourhood of the leading and trailing edges of the airfoil. However, we see that the mesh designed by the weighted error indicator is also refined along the upper and lower surfaces of the airfoil. These extra regions of refinement are introduced as a result of the weighting terms present in the error indicator $|\hat{\eta}_\kappa|$,

TABLE 2
Subsonic NACA0012 flow. Adaptive algorithm for the evaluation of the drag coefficient.

Elements	DOF	$J(\mathbf{u}) - J(\mathbf{u}_h)$	$\sum_\kappa \hat{\eta}_\kappa$	θ_1	$\sum_\kappa \lvert\hat{\eta}_\kappa\rvert$	θ_2
39	624	-1.739e-02	-1.591e-02	0.91	2.906e-02	1.67
66	1056	-6.041e-03	-3.070e-04	0.05	1.934e-02	3.20
111	1776	-6.914e-03	-1.798e-03	0.26	2.491e-02	3.60
177	2832	-5.542e-03	-3.427e-03	0.62	1.893e-02	3.42
279	4464	-3.888e-03	-3.493e-03	0.90	1.018e-02	2.62
420	6720	-1.357e-03	-1.303e-03	0.96	5.157e-03	3.80
678	10848	-8.786e-04	-8.907e-04	1.01	2.417e-03	2.75
1029	16464	-5.926e-04	-6.901e-04	1.16	1.306e-03	2.20
1605	25680	-2.519e-04	-2.756e-04	1.09	5.574e-04	2.21
2508	40128	-1.234e-04	-1.333e-04	1.08	2.956e-04	2.39
3972	63552	-7.969e-05	-8.222e-05	1.03	1.976e-04	2.48
6126	98016	-4.051e-05	-4.036e-05	1.00	1.022e-04	2.52

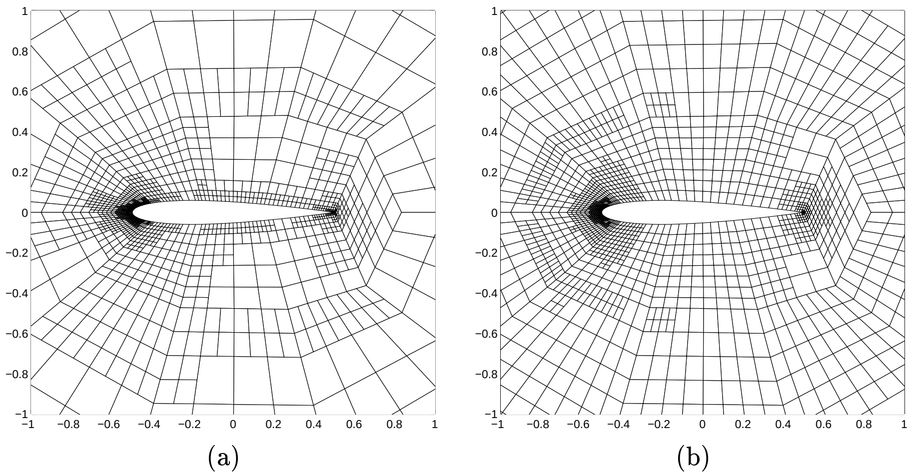


FIG. 8. Evaluation of the drag coefficient for the subsonic NACA0012 flow. (a) Mesh constructed using dual error indicator with 2508 elements and 40128 degrees of freedom ($|J(\mathbf{u}) - J(\mathbf{u}_h)| = 1.234 \times 10^{-4}$); (b) Mesh constructed using *ad hoc* error indicator with 2634 elements and 42144 degrees of freedom ($|J(\mathbf{u}) - J(\mathbf{u}_h)| = 5.856 \times 10^{-4}$).

cf. (20). Indeed, in Figure 9, we see that while there is a singularity in the dual solution at the leading edge of the airfoil, which is transported upstream along the characteristic with eigenvalue $\bar{v}$, $\hat{\mathbf{z}}_1$ also has large gradients around the whole of the geometry. In Figure 10 we compare the true error in the computed functional $J(\cdot)$ using the two mesh refinement strategies. First we note that here the convergence of $|J(\mathbf{u}) - J(\mathbf{u}_h)|$ on the sequence of meshes produced with the weighted error indicator is much ‘smoother’ than on the meshes designed with the *ad hoc* error indicator. Indeed, on finer meshes the error is even monotone allowing extrapolation

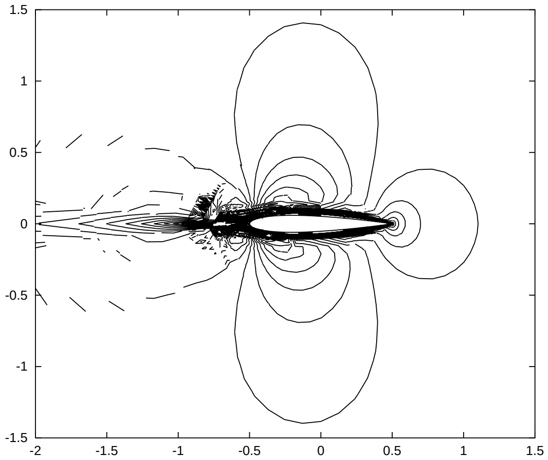


FIG. 9. Subsonic NACA0012 airfoil: first component $\hat{\mathbf{z}}_1$ of the dual solution for the evaluation of the drag coefficient.

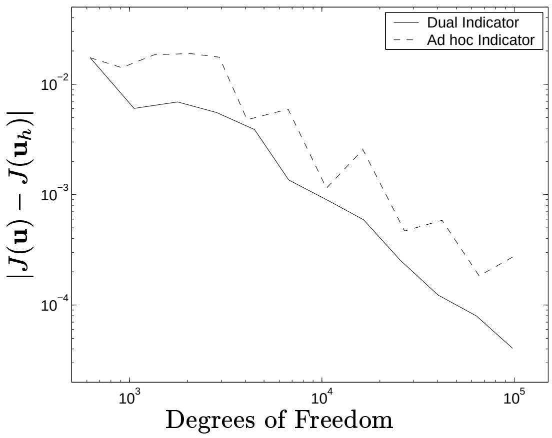


FIG. 10. Evaluation of the drag coefficient for the subsonic NACA0012 flow. Convergence of $|J(\mathbf{u}) - J(\mathbf{u}_h)|$ using the dual and *ad hoc* error indicators.

of the value of the target functional in comparison to the 'zig–zag' convergence of $|J(\mathbf{u}) - J(\mathbf{u}_h)|$ on the meshes designed by $\eta_\kappa^{\mathrm{adhoc}}$. Again, as in the previous example, we clearly see that our weighted *a posteriori* error indicators produce more economical meshes than when the traditional *ad hoc* error indicator is employed, in the sense that $|J(\mathbf{u}) - J(\mathbf{u}_h)|$ is smaller for a given number of degrees of freedom. Indeed, on the final mesh the true error in the computed functional is almost an order of magnitude smaller when the former error indicator is used.

We end this section by considering the case when $J(\cdot) \equiv J_{\mathrm{lift}}(\cdot)$. This is a relatively easy task; on a symmetric grid consisting of only 48 elements, the true error in the functional is $J(\mathbf{u}) - J(\mathbf{u}_h) = 1.332 \times 10^{-9}$, and the computed error representation formula is $\hat{\mathcal{E}}_\Omega = 2.1455 \times 10^{-9}$ giving rise to the effectivity index $\theta_1 = 1.61$.

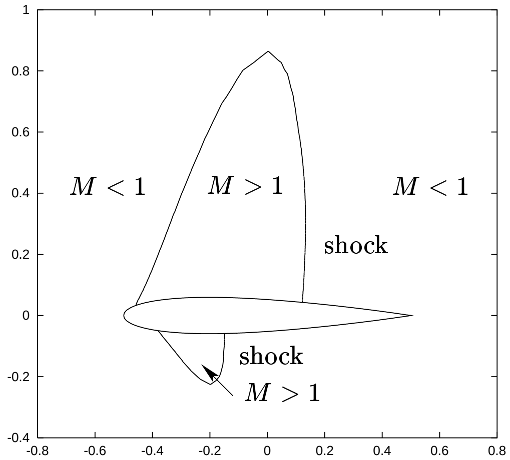


FIG. 11. Structure of the solution to the transonic NACA0012 airfoil problem.

6.3. Transonic flow around a NACA0012 airfoil

In this section, we again compute the flow around a NACA0012 airfoil; here, we now consider a Mach 0.8 flow with an angle of attack $\alpha = 1.25°$, and farfield density $\rho = 1$ and pressure $p = 1$. The structure of the solution in this example is now quite different from Section 6.2. Here, as the impinging subsonic flow passes the airfoil, it first accelerates to become supersonic, before slowing to be subsonic again; this results in the development of shocks on both the upper and lower surfaces of the airfoil, cf. Figure 11. Throughout this section we set artificial viscosity parameters appearing in (16) as follows: $C_\varepsilon = 3/100$ and $\beta = 1/10$.

We first consider the estimation of the drag on the surface of the airfoil; here, on the basis of a fine grid computation, the true value of the functional is given by $J(\mathbf{u}) = 0.0201$. In Table 3 we show the performance of our adaptive algorithm; here we see that while the computed *a posteriori* estimate still remains a reliable upper–bound on the size of the error in the computed functional, the ratio of the approximate error representation formula $\hat{\mathcal{E}}_\Omega$ and $J(\mathbf{u}) - J(\mathbf{u}_h)$ is slightly less than one on all of the meshes employed. This degradation in the quality of the computed error representation formula is attributed to the effects of the linearisation error and the error in the approximation of the Fréchet derivative of the semilinear form $\mathcal{N}(\cdot,\cdot)$ (with respect to its first argument), becoming large in the vicinity of the shocks which are in contact with the airfoil. In this case, the weighting terms involving the difference between the dual solution and its projection/interpolant (cf. (23)) are poorly approximated in the neighbourhood of the shock, where the residual terms R_h and r_h, and the shock capturing term $\varepsilon\nabla\mathbf{u}_h$ are large, thereby leading to a poor approximation of the true error in the target functional $J(\cdot)$. Thus, in order to guarantee the quality of the computed error representation formula, additional computational work is required to reduce the effects of the linearisation error, the error stemming from the approximation of $\mathcal{N}'_{\mathbf{u}}(\cdot, \mathbf{v})$, $\mathbf{v} \in V$ fixed, and the error generated in the numerical approximation of the (approximate) dual problem; this is beyond the scope of the current paper, but will be considered in [11].

TABLE 3
Transonic NACA0012 flow. Adaptive algorithm for the evaluation of the drag coefficient.

Elements	DOF	$J(\mathbf{u}) - J(\mathbf{u}_h)$	$\sum_\kappa \hat{\eta}_\kappa$	θ_1	$\sum_\kappa \lvert\hat{\eta}_\kappa\rvert$	θ_2
39	624	-4.453e-02	-1.344e-02	0.30	2.585e-02	0.58
57	912	-3.107e-02	-1.353e-02	0.44	3.410e-02	1.10
90	1440	-1.894e-02	-1.243e-02	0.66	3.147e-02	1.66
144	2304	-9.869e-03	-7.086e-03	0.72	2.116e-02	2.14
240	3840	-7.084e-03	-5.430e-03	0.77	1.368e-02	1.93
393	6288	-5.708e-03	-4.282e-03	0.75	8.673e-03	1.52
636	10176	-3.081e-03	-2.554e-03	0.83	6.072e-03	1.97
1056	16896	-1.691e-03	-1.332e-03	0.79	3.193e-03	1.89
1713	27408	-1.022e-03	-7.399e-04	0.72	1.928e-03	1.89
2655	42480	-7.521e-04	-5.171e-04	0.69	1.317e-03	1.75
4233	67728	-3.707e-04	-2.096e-04	0.57	8.210e-04	2.21
6603	105648	-2.257e-04	-1.823e-04	0.81	6.107e-04	2.71

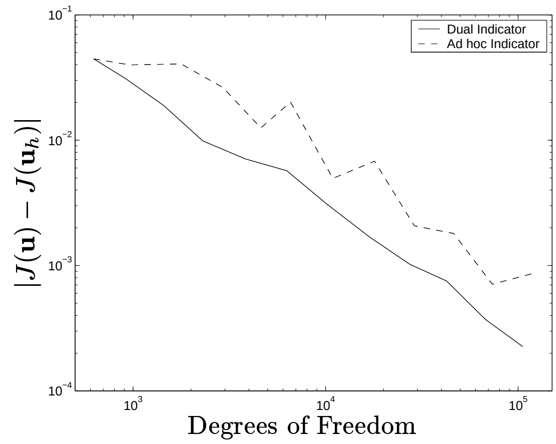


FIG. 12. Evaluation of the drag coefficient for the transonic NACA0012 flow. Convergence of $|J(\mathbf{u}) - J(\mathbf{u}_h)|$ using the dual and *ad hoc* error indicators.

Notwithstanding this reduction in the quality of the computed error representation formula, the adaptively refined meshes generated by employing the weighted error indicator $|\hat{\eta}_\kappa|$ are much more economical than those produced using the traditional *ad hoc* error indicator $\eta_\kappa^{\mathrm{adhoc}}$. Indeed, in Figure 12 we clearly observe the superiority of the former error indicator; on the final mesh the true error in the computed functional is almost an order of magnitude smaller when the weighted error indicator is employed. Again, as in the previous example, the convergence of $|J(\mathbf{u}) - J(\mathbf{u}_h)|$ when the weighted error indicator is employed is monotone in contrast to the 'zig–zag' convergence for the case of the *ad hoc* error indicator.

In Figure 13 we show the meshes produced using both error indicators. Here, we see that the mesh constructed using $\eta_\kappa^{\mathrm{adhoc}}$ is concentrated in the neighbourhood of the leading and trailing edges of the airfoil as well as the shocks on the upper and

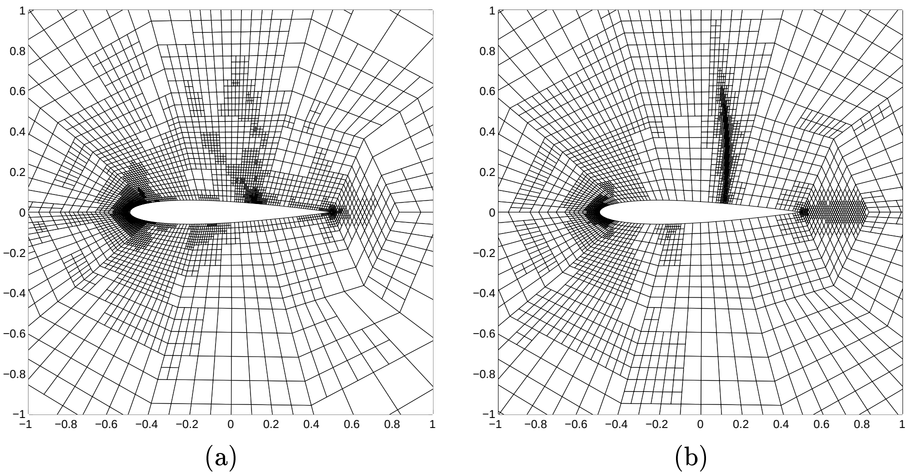


FIG. 13. Evaluation of the drag coefficient for the transonic NACA0012 flow. (a) Mesh constructed using dual error indicator with 6603 elements and 105648 degrees of freedom ($|J(\mathbf{u}) - J(\mathbf{u}_h)| = 2.257 \times 10^{-4}$); (b) Mesh constructed using *ad hoc* error indicator with 7320 elements and 117120 degrees of freedom ($|J(\mathbf{u}) - J(\mathbf{u}_h)| = 8.571 \times 10^{-4}$).

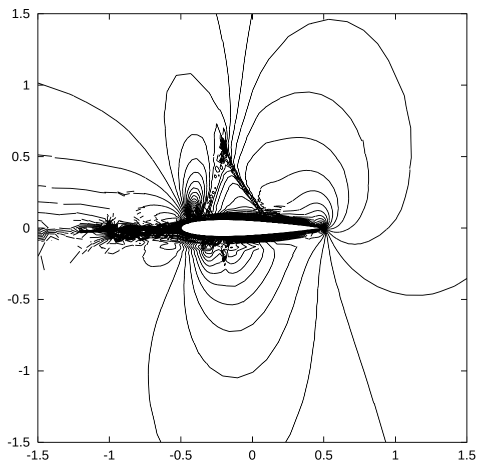


FIG. 14. Transonic NACA0012 airfoil: first component $\hat{\mathbf{z}}_1$ of the dual solution for the evaluation of the drag coefficient.

lower surfaces. The mesh designed using $|\hat{\eta}_\kappa|$ is also highly refined in the vicinity of the leading and trailing edges of the airfoil, due to the presence of singularities in the dual $\hat{\mathbf{z}}$, cf. Figure 14; however, here the shocks are now far less refined. Indeed, on the upper surface of the airfoil, a line of refinement has been introduced in the direction of the $\bar{v} + c$ characteristic emanating from the position of the shock on the airfoil; along this characteristic, the dual solution is singular, cf. Figure 14.

Finally, we end this section by considering the estimation of the lift coefficient; on the basis of fine grid computations, the true value of the functional is given by $J(\mathbf{u}) = 0.3118$. In Table 4 we see again that although the computed *a posteriori*

TABLE 4
Transonic NACA0012 flow. Adaptive algorithm for the evaluation of the lift coefficient.

Elements	DOF	$J(\mathbf{u}) - J(\mathbf{u}_h)$	$\sum_\kappa \hat{\eta}_\kappa$	θ_1	$\sum_\kappa \lvert\hat{\eta}_\kappa\rvert$	θ_2
39	624	1.257e-01	4.476e-02	0.36	3.208e-01	2.55
57	912	1.064e-01	5.433e-02	0.51	2.428e-01	2.28
90	1440	3.292e-03	2.003e-02	6.08	1.902e-01	57.78
141	2256	-1.426e-02	1.378e-02	-0.97	1.123e-01	7.88
222	3552	-4.276e-02	-1.312e-02	0.31	1.248e-01	2.92
387	6192	-1.055e-02	7.767e-03	-0.74	1.043e-01	9.88
624	9984	-2.082e-02	-5.366e-03	0.26	7.072e-02	3.40
990	15840	-1.379e-02	-3.866e-03	0.28	4.928e-02	3.57
1623	25968	-8.779e-03	-2.343e-03	0.27	4.006e-02	4.56
2523	40368	-6.837e-03	-1.497e-03	0.22	2.512e-02	3.67
4023	64368	-4.680e-03	-1.609e-03	0.34	1.871e-02	4.00
6390	102240	-3.063e-03	-1.353e-03	0.44	1.218e-02	3.98
10110	161760	-1.805e-03	-1.146e-03	0.63	9.386e-03	5.20

bound remains a reliable upper-bound on the size of the error in the computed functional, the quality of the error representation formula is generally quite poor. Indeed, we see that the effectivity index θ_1 is less than one for all the meshes employed; we also note that the quality of computed error representation formula for the lift is worse than for the evaluation of the drag coefficient. The structure of the dual solution and thereby the meshes designed using $|\hat{\eta}_\kappa|$ are similar in character to those computed when $J(\cdot) \equiv J_{\mathrm{drag}}(\cdot)$; thereby, for the sake of brevity, these results are omitted.

6.4. Supersonic flow around a BAC3-11 airfoil

In this final example we study a supersonic flow around a BAC3-11 airfoil; this unsymmetric airfoil, see Figure 15, was originally specified in the AGARD report AR-303. Here, we consider a Mach 1.2 flow at an angle of attack $\alpha = 5°$, with inflow density $\rho = 1$ and pressure $p = 1$. Throughout this section, we set the shock capturing parameters C_ε and β, appearing in (16), equal to 1/50 and 1/10, respectively.

The structure of the solution to this problem includes two shocks: one located in front of the leading edge of the airfoil and one originating from the trailing edge; see Figure 16(a) and also Figure 17(b) which shows a mesh that is refined at the position of the two shocks. Here, Figure 16(a) shows the Mach 1 isolines of the solution; the Mach $M = 1$ isoline to the left of the airfoil indicates the position of the first shock. The $M = 1$ isolines that originate from the upper and lower surfaces of the airfoil represent the transonic lines of the flow. The flow left of the first shock is supersonic; it is simply the $M = 1.2$ flow prescribed on the inflow boundary of the computational domain. The flow in between the shock and the transonic lines is subsonic; we note that the leading edge of the airfoil is located within this subsonic part of the flow. Finally, the flow behind the transonic lines is supersonic again.

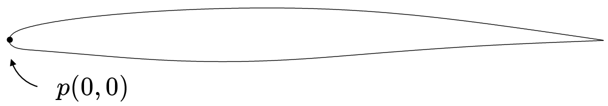


FIG. 15. Profile of the BAC3-11 airfoil. Quantity of interest: pressure p at leading edge.

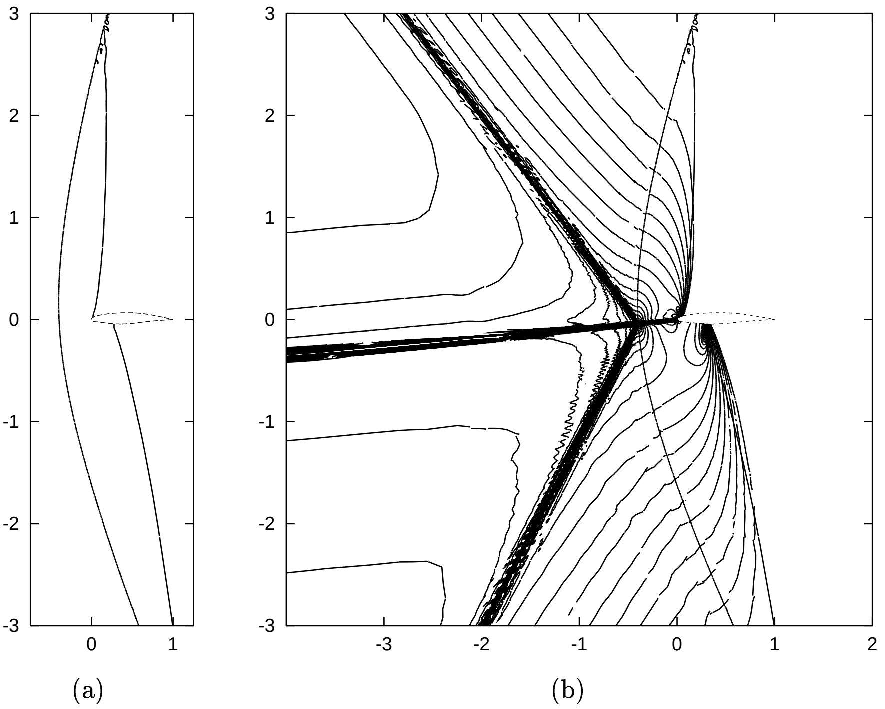


FIG. 16. Supersonic BAC3-11 flow. (a) Mach 1 isolines of the primal solution; (b) $\hat{\mathbf{z}}_1$ isolines of dual solution and the Mach 1 isolines of the primal solution

In this example we take the functional of interest to be the value of the pressure at the leading edge, i.e.

$$J(\mathbf{u}) = p(0,0),$$

cf. Figure 15. We note that, as in the case of estimating the drag and lift coefficients of a body immersed in an inviscid fluid, cf. Sections 6.2 & 6.3, this functional is nonlinear as it depends on the pressure. A computation on a fine grid gives a reference value of $J(\mathbf{u}) = 2.393$.

The structure of the dual solution $\hat{\mathbf{z}}$ corresponding to this point evaluation is displayed in Figure 16(b). This figure illustrates some principles of information transport in supersonic as well as in subsonic flow regions as we have already seen in Sections 6.1–6.3. To the right–hand side of the transonic lines the dual solution is zero as no information, neither by material transport nor even by information

TABLE 5
Supersonic BAC3-11 flow. Adaptive algorithm for the point evaluation of the pressure at the leading edge.

Elements	DOF	$J(\mathbf{u}) - J(\mathbf{u}_h)$	$\sum_\kappa \hat{\eta}_\kappa$	θ_1	$\sum_\kappa \|\hat{\eta}_\kappa\|$	θ_2
39	624	3.188e-01	2.773e-01	0.87	4.288e-01	1.35
63	1008	2.313e-01	-1.501e-02	-0.06	2.003e-01	0.87
114	1824	2.069e-01	7.275e-02	0.35	3.498e-01	1.69
192	3072	7.398e-02	5.404e-02	0.73	2.680e-01	3.62
348	5568	6.425e-02	2.695e-02	0.42	2.120e-01	3.30
609	9744	2.876e-02	1.389e-02	0.48	1.839e-01	6.39
1065	17040	5.066e-03	7.602e-03	1.50	1.171e-01	23.11
1803	28848	3.042e-03	2.868e-03	0.94	1.028e-01	33.78
3045	48720	1.561e-03	2.801e-03	1.79	1.067e-01	68.39
5643	90288	5.790e-04	5.790e-04	1.00	5.551e-02	95.88

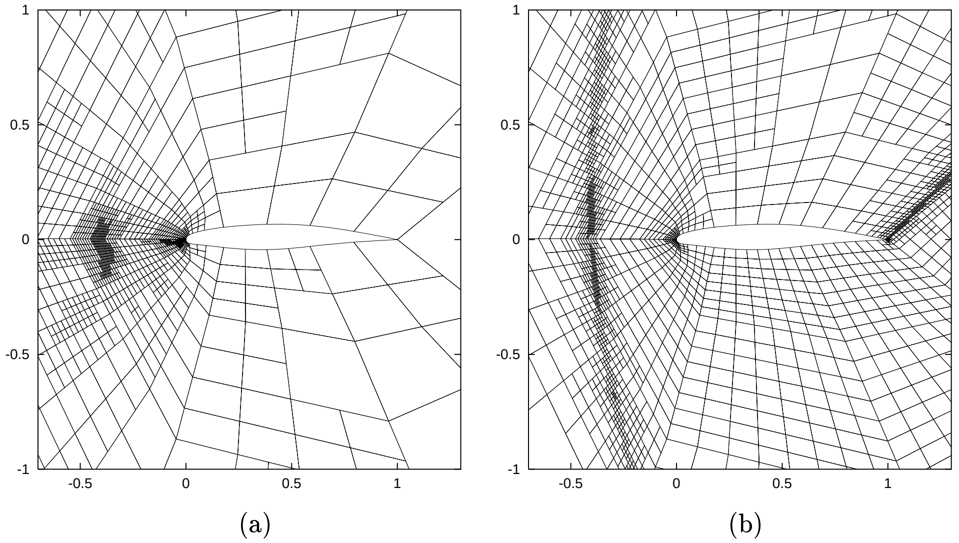


FIG. 17. Supersonic BAC3-11 flow. (a) Mesh constructed using dual error indicator with 1803 elements and 28848 degrees of freedom ($|J(\mathbf{u}) - J(\mathbf{u}_h)| = 3.042 \times 10^{-3}$); (b) Mesh constructed using *ad hoc* error indicator with 13719 elements and 219504 degrees of freedom ($|J(\mathbf{u}) - J(\mathbf{u}_h)| = 3.542 \times 10^{-2}$).

transport due to sound waves, can enter the subsonic region from the supersonic one. Within the whole subsonic region the dual solution is non-zero corresponding to the fact that sound waves can reach the point of evaluation from any point in the subsonic area and that all numerical errors which occur within this subsonic region can (even though possibly to a small portion) affect the value of the solution at the point of evaluation. Nevertheless, the dual solution in the subsonic region is concentrated in a thin spike that is transported upstream from the point of evaluation in direction of the flow. This spike corresponds to the path of material transport and represents the main path of information transport. To the left of the

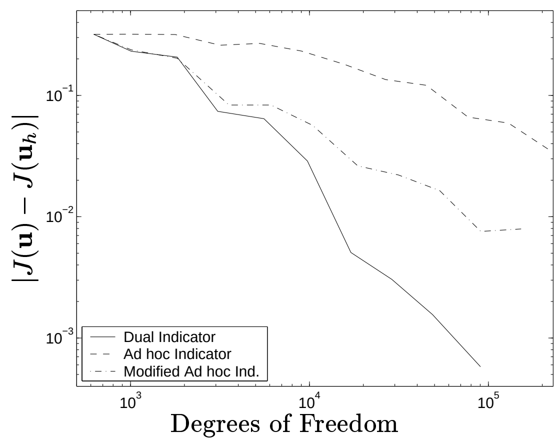
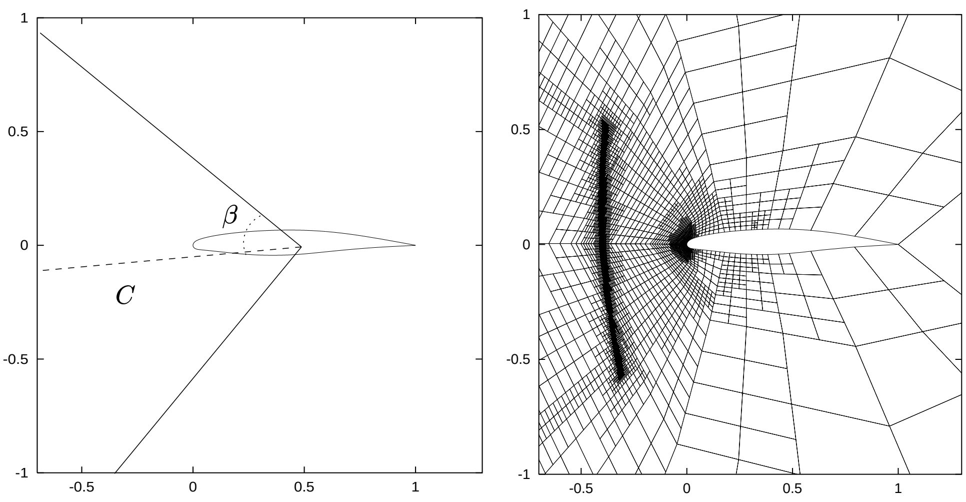


FIG. 18. Supersonic BAC3-11 flow. Convergence of $|J(\mathbf{u}) - J(\mathbf{u}_h)|$ using the dual and *ad hoc* error indicators.

airfoil, this spike crosses the shock and splits into three spikes while entering the supersonic region left of the shock. These spikes are transported upstream along the characteristics corresponding to the three eigenvalues $\bar{v}$ and $\bar{v} \pm c$. We recall that the characteristic corresponding to $\bar{v}$ represents the path of material transport, that in this example is given by the line inclined at 5 degrees, whereas the characteristics corresponding to $\bar{v} \pm c$ represent the paths of information transport due to sound waves.

In Table 5 we show the performance of our adaptive algorithm. Here, we see that on very coarse meshes the quality of the computed error representation formula $\hat{\mathcal{E}}_\Omega$ is quite poor, in the sense that the effectivity index θ_1 is not close to one. However, ignoring the first few refinement steps, we see that θ_1 lies in the interval $(0.94, 1.79)$. Moreover, we observe that the Type I *a posteriori* estimate provides a reliable upper bound on the size of the error in the computed functional $J(\cdot)$.

In Figure 17 we show the meshes produced using the weighted and the *ad hoc* error indicators. Here, we see that the mesh constructed using $\eta_\kappa^{\mathrm{adhoc}}$ is concentrated in the neighbourhood of the two shocks. In contrast, the mesh produced using the weighted error indicator $|\hat{\eta}_\kappa|$ only refines the mesh in the vicinity of the point of evaluation and the part of the shock where the spike of the dual solution, i.e. where the main part of information, crosses the shock. The other parts of the shock are not resolved, as the numerical error in these regions only has a small affect on the accuracy of the solution at the point of evaluation. Also there is no refinement in the vicinity of the shock emanating from the trailing edge of the airfoil; thereby, this shock is not well resolved at all. Nevertheless the solution at the leading edge of the airfoil is not affected by this as no information is transported upstream from the trailing edge, located in a supersonic part of the flow, to the leading edge, located in the subsonic region. As in the previous examples, we see that the adaptively refined meshes generated by employing the weighted error indicator $|\hat{\eta}_\kappa|$ are much more economical than those produced using the traditional *ad hoc* error indicator $\eta_\kappa^{\mathrm{adhoc}}$. Indeed, in Figure 18 we clearly observe the superiority of the former error



(a) (b)

FIG. 19. Supersonic BAC3-11 flow. (a) Cone C: domain where the modified *ad hoc* error indicator is active; (b) Mesh constructed using the modified *ad hoc* error indicator with 9516 elements and 152256 degrees of freedom ($|J(\mathbf{u}) - J(\mathbf{u}_h)| = 7.924 \times 10^{-3}$).

indicator; on the final mesh the true error in the computed functional is over two orders of magnitude smaller when the weighted error indicator is employed.

Motivated by the structure of the mesh generated by the weighted error indicator, here we also consider the performance of an alternative *ad hoc* error indicator based on a modification of η^{adhoc}, whereby only elements in a neighbourhood of a region upstream of the point of interest are marked for refinement. More precisely, we write C to denote the cone depicted in Figure 19(a) with apex half angle β, located in the centre of the airfoil with symmetry axes inclined at $\alpha = 5°$ according to the direction of the inflow. We now define the modified *ad hoc* indicator $\eta_\kappa^{\mathrm{adhoc,C}}$ as follows:

$$\eta_\kappa^{\mathrm{adhoc,C}} = \begin{cases} \eta_\kappa^{\mathrm{adhoc}}, & \text{if centroid}(\kappa) \in C, \\ 0, & \text{otherwise.} \end{cases}$$

This modification takes into account that we are not interested in the flow field in the whole domain, but only in the point value of the pressure at the leading edge. Thereby, adaptive mesh refinement is inhibited in the region downstream of the airfoil including the neighbourhood of the shock emanating from the trailing edge. Furthermore, refinement of the shock in front of the leading edge of the airfoil is prevented in regions that are placed too far above or below the airfoil since a low resolution of this shock in these areas is believed to not significantly degrade the accuracy of the pressure value at the leading edge, cf. Figure 17(a). In Figure 19(b) we show the mesh produced by employing $\eta_\kappa^{\mathrm{adhoc,C}}$ with $\beta = 45°$.

Finally, in Figure 18 we see that the modified *ad hoc* indicator produces meshes that are much more efficient for computing the value of the pressure at the leading edge of the airfoil in comparison to the (unmodified) *ad hoc* indicator $\eta_\kappa^{\mathrm{adhoc}}$. Nevertheless the meshes produced employing the weighted indicators are even more efficient than those designed by $\eta_\kappa^{\mathrm{adhoc,C}}$; on the final mesh the true error in the computed functional is over an order of magnitude smaller when the weighted error indicator is employed. We note that the chosen shape and size of the subdomain C and the resulting modified indicator only represents an 'attempt' to find a reasonable modification of the *ad hoc* indicator $\eta_\kappa^{\mathrm{adhoc}}$ that is capable of efficiently computing the pressure at the leading edge of the airfoil and to provide a 'fair' comparison with the goal–oriented weighted indicator $|\hat{\eta}_\kappa|$. Indeed, the value of the angle β may be chosen differently, though *a priori* it is unclear which parts of the shock in front of the leading edge of the airfoil will influence the target functional. The angle β should not be chosen too small as otherwise the lack of resolution of the shock in front of the leading edge of the airfoil will impact on the computed value of the pressure at the point of interest; on the other hand choosing β too large may lead to over–refinement. In contrast, the weighted error indicator provides all the necessary information in order to decide which regions of the shock should be refined, and by what extent.

7. CONCLUDING REMARKS

In this article we have applied the recently developed approach of weighted *a posteriori* error estimation for discontinuous Galerkin finite element methods to the discretisation of physically relevant problems arising in inviscid compressible fluid flows governed by the Euler equations of gas dynamics. Indeed, here weighted or Type I (cf. [14]) *a posteriori* error bounds have been employed for the design of economical adaptive finite element algorithms capable of delivering sharp error estimation for general linear and nonlinear target functionals of the solution. Numerical experiments have been presented to illustrate both the quality of the approximate error representation formula and the resulting Type I *a posteriori* error bound, when the (approximate) dual problem is approximated numerically. In all cases, our weighted *a posteriori* estimates provide sharp upper bounds on the size of the error in the computed target functional of physical interest. Moreover, comparisons with traditional *ad hoc* error indicators which stem from Type II *a posteriori* error bounds have clearly demonstrated the superiority of exploiting weighted *a posteriori* error indicators to guide adaptive mesh refinement; in all examples, the error in the computed target functional is significantly smaller for a given number of degrees of freedom when the proposed error indicator is employed.

ACKNOWLEDGMENT

Ralf Hartmann acknowledges the financial support of the DFG Priority Research Program and the SFB 359 at the IWR, University of Heidelberg. Paul Houston acknowledges the financial support of the EPSRC (Grant GR/N24230). Part of this work was performed using the University of Leicester Mathematical Modelling Centre's supercomputer which was purchased through the EPSRC strategic equipment initiative.

REFERENCES

1. F. Bassi and S. Rebay, *High–order accurate discontinuous finite element solution of the 2D Euler equations.* J. Comput. Phys. 138:251–285, 1997.
2. R. Becker and R. Rannacher, *A feed–back approach to error control in finite element methods: Basic analysis and examples.* East–West J. Num. Math., 4:237–264, 1996.
3. E. Brooksbanks, *A Numerical Investigation of Time Resolved Flows Around Turbine Blades.* Ph.D. thesis, University of Leicester (submitted).

4. B. Cockburn, S. Hou, and C. Shu, *TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: The multidimensional case.* Math. Comp., 54:545–581, 1990.

5. B. Cockburn, G.E. Karniadakis and C.-W. Shu, *The development of discontinuous Galerkin methods.* In: B. Cockburn, G. Karniadakis, and C.-W. Shu (Eds.) Discontinuous Galerkin Finite Element Methods. Lecture Notes in Computational Science and Engineering, Volume 11. Springer-Verlag 2000.

6. B. Cockburn and C.-W. Shu, *The Runge–Kutta discontinuous Galerkin finite element method for conservation laws V: Multidimensional systems.* J. Comput. Phys. 141:199–224, 1998.

7. K. Eriksson, D. Estep, P. Hansbo, and C. Johnson, *Introduction to adaptive methods for differential equations.* Acta Numerica, 105–158, 1995.

8. P. Hansbo and C. Johnson, *Streamline diffusion finite element methods for fluid flow.* von Karman Institute Lectures, 1995.

9. R. Hartmann, *Adaptive FE–methods for conservation equations.* In: G. Warnecke and H. Freistühler (Eds.) Proc. Eighth International Conference on Hyperbolic Problems. Theory, Numerics, Applications (HYP2000). Birkhäuser, Basel. (to appear).

10. R. Hartmann and P. Houston, *Adaptive Discontinuous Galerkin Finite Element Methods for Nonlinear Hyperbolic Conservation Laws.* Leicester University Technical Report 2001/20, 2001. (Submitted for publication). Available from the URL **http://www.mcs.le.ac.uk/~phouston/**.

11. R. Hartmann and P. Houston, *A posteriori error estimation for nonlinear hyperbolic conservation laws involving discontinuous solutions.* In preparation.

12. P. Houston, R. Hartmann and E. Süli, *Adaptive Discontinuous Galerkin Finite Element Methods for Compressible Fluid Flows.* In M. Baines, editor, *Numerical Methods for Fluid Dynamics VII*, ICFD, 347–353, 2001.

13. P. Houston, R. Rannacher and E. Süli, *A posteriori error analysis for stabilised finite element approximations of transport problems.* Comput. Meth. Appl. Mech. Engrg., 190(11-12):1483–1508, 2000.

14. P. Houston and E. Süli, *hp–Adaptive discontinuous Galerkin finite element methods for first–order hyperbolic problems.* SIAM J. Sci. Comp. (to appear).

15. J. Jaffre, C. Johnson and A. Szepessy, *Convergence of the discontinuous Galerkin finite element method for hyperbolic conservation laws.* Math. Models Methods Appl. Sci. 5(3):367–386, 1995.

16. C. Johnson and A. Szepessy, *Adaptive finite element methods for conservation laws.* Commun. Pure Appl. Math. 48:199–243, 1995.

17. D. Kröner, *Numerical Schemes for Conservation Laws.* Wiley-Teubner, 1997.

18. M.G. Larson and T.J. Barth, *A posteriori error estimation for discontinuous Galerkin approximations of hyperbolic systems.* In: B. Cockburn, G. Karniadakis, and C.-W. Shu, editors, *Discontinuous Galerkin Finite Element Methods.* Lecture Notes in Computational Science and Engineering, Volume 11. Springer–Verlag 2000.

19. R. Rannacher, *Adaptive finite element methods*, in H. Bulgak and C. Zenger, eds., Proc. NATO-Summer School *Error Control and Adaptivity in Scientific Computing*, Kluwer, Academic Publishers, 247–278, 1998.

20. E. Süli, *A posteriori error analysis and adaptivity for finite element approximations of hyperbolic problems.* In: D. Kröner, M. Ohlberger and C. Rohde (Eds.) An Introduction to Recent Developments in Theory and Numerics for Conservation Laws. Lecture Notes in Computational Science and Engineering. Volume 5, pages 123–194. Springer-Verlag, 1998.

21. E. Toro, *Riemann solvers and numerical methods for fluid dynamics.* Springer–Verlag, 1997.