Goal–Oriented *A Posteriori* Error Estimation for Multiple Target Functionals

Ralf Hartmann[1,⋆] and Paul Houston[2,⋆⋆]

[1] Institute of Applied Mathematics, University of Heidelberg, Im Neuenheimer Feld 293, D-69120 Heidelberg, Germany *Ralf.Hartmann@iwr.uni-heidelberg.de*

[2] Department of Mathematics & Computer Science, University of Leicester, Leicester LE1 7RH, UK *Paul.Houston@mcs.le.ac.uk*

1 Introduction

In many applications the quantities of interest are a series of target functionals of the solution to the governing system of partial differential equations rather than the solution itself. For example, in the field of aerodynamics, examples include the drag and lift coefficients of an airfoil immersed into a fluid, the pressure difference between the leading and trailing edges of the airfoil and point evaluations of the density or pressure on the profile of the airfoil. While traditionally these quantities are measured in wind tunnel experiments, nowadays these experiments are increasingly replaced by numerical simulations aiming to predict these quantities to a high level of accuracy.

In the series of articles [2, 4, 5, 6, 7], we have developed the theory of goal–oriented *a posteriori* error estimation for discontinuous Galerkin methods applied to inviscid compressible fluid flows; see [10], for related work. On the basis of Type I *a posteriori* bounds, in [2, 4, 5, 6, 7], we considered the design of adaptive finite element algorithms that are capable of generating optimal meshes specifically tailored to the efficient computation of a *single* target functional of practical interest. The purpose of the current article is to extend this earlier work to the case when several target functionals of the solution need to be simultaneously approximated to a given level of accuracy.

2 Duality based *a posteriori* error estimation

Given that X and Y are two Hilbert spaces, $B(\cdot,\cdot)$ is a bounded bilinear functional defined on $X\times Y$ and $\ell(\cdot)$ is a bounded linear functional defined on Y, we suppose that u is the unique solution to the variational problem:

$$\text{find } u\in X \text{ such that } \quad B(u,v)=\ell(v) \quad \forall v\in Y\,. \tag{1}$$

Problem (1) can be thought of as the weak formulation of a linear partial differential equation on X whose unique solution is $u\in X$. In practice (1)

⋆ Supported by the DFG Priority Research Program and the SFB 359 at the IWR, University of Heidelberg.

⋆⋆ Paul Houston acknowledges the financial support of the EPSRC (GR/N24230).

cannot be solved in closed form but needs to be approximated numerically. In order to construct a Galerkin approximation to this problem, we consider a sequence of finite–dimensional spaces $\{X_h\}$ and $\{Y_h\}$, parameterised by $h > 0$; for simplicity we suppose that $X_h \subset X$ and $Y_h \subset Y$ for each h. For 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 then satisfies:

$$\text{find } u_h \in X_h \text{ such that } \quad B(u_h, v_h) = \ell(v_h) \quad \forall v_h \in Y_h \,. \tag{2}$$

Assuming there exists a unique solution u_h to (2), we note that since the discretisation method (2) is fully consistent with the original (primal) problem, the error $u - u_h$ satisfies the following Galerkin orthogonality property

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

It is frequently the case in applications that, instead of accurately computing u_h everywhere within the computational domain, the focus of interest is a derived quantity $J(u_h)$ which should be computed to high accuracy, where $J(\cdot)$ is a given bounded linear functional defined on X. Indeed, it may happen that instead of a single linear functional $J(\cdot)$, several linear functionals $J_i(\cdot)$, $i = 1, \dots, N$, $N \geq 1$, are given, and one needs to compute each $J_i(u_h)$ to within a given user–defined tolerance $\mathtt{TOL_i}$, $i = 1, \dots, N$, respectively. More precisely, we consider the following *measurement problem*: find $J_i(u_h) \in \mathbb{R}$, $i = 1, \dots, N$, such that

$$|J_i(u) - J_i(u_h)| \leq \mathtt{TOL_i}\,, \quad \text{for } i = 1, \dots, N. \tag{4}$$

Our approach to the solution of this problem rests on a duality argument. For simplicity, we first outline the general technique for the case of a single target functional, i.e. $N = 1$, and we write $J(\cdot) \equiv J_1(\cdot)$. The starting point for the *a posteriori* error analysis is to introduce the following *dual* problem:

$$\text{find } z \in Y \text{ such that } \quad B(w, z) = J(w) \quad \forall w \in X \,. \tag{5}$$

By using the linearity of $J(\cdot)$, together with (5), (3) and (1), we get

$$\begin{aligned} J(u) - J(u_h) &= J(u - u_h) = B(u - u_h, z) = B(u - u_h, z - z_h) \\ &= \ell(z - z_h) - B(u_h, z - z_h) \quad \forall z_h \in Y_h \,. \end{aligned} \tag{6}$$

As $v \mapsto \ell(v) - B(u_h, v)$ is a bounded linear functional on Y for each $u_h \in X_h$, there exists $R_p(u_h) \in Y'$ (where Y' denotes the dual space of Y) such that

$$\ell(z - z_h) - B(u_h, z - z_h) = \langle R_p(u_h), z - z_h \rangle \,; \tag{7}$$

here $\langle \cdot, \cdot \rangle$ denotes the duality pairing between Y' and Y. The element $R_p(u_h)$ will be referred to as the *primal residual*. Combining (6) and (7) gives

$$J(u) - J(u_h) = \langle R_p(u_h), z - z_h \rangle \qquad \forall z_h \in Y_h \,, \tag{8}$$

which is referred to as the *primal error representation formula*. On the basis of (8), Type I and Type II *a posteriori* error bounds may be derived, cf. [4].

Let us now consider the extension of the above analysis to the case when $N > 1$. The *direct method* for deriving an error representation formula analogous to (8) for each $J_i(\cdot)$ is to introduce the following N dual problems:

$$\text{find } z_i \in Y \text{ such that } \;\; B(w, z_i) = J_i(w) \quad \forall w \in X \,, \text{ for } i = 1, \dots , N. \tag{9}$$

Thereby, repeating the above argument for each functional, we deduce that

$$J_i(u) - J_i(u_h) = \langle R_p(u_h), z_i - z_h \rangle \quad \forall z_h \in Y_h \,, \text{ for } i = 1, \dots , N. \tag{10}$$

In practice, the dual solutions z_i, $i = 1, \dots , N$, are unknown analytically and must be approximated numerically. Upon discretisation, this amounts to solving N systems of linear equations with the same matrix but N different right–hand side vectors. In view of the error representation formula

$$J_i(u - u_h) = B(u - u_h, z_i) = \langle R_p(u_h), z_i \rangle = \langle R_p(u_h), z_i - z_h \rangle \,, \tag{11}$$

an alternative approach consists of considering the following *error equation*:

$$\text{find } e \in X \text{ such that } \;\; B(e, w) = \langle R_p(u_h), w \rangle \quad \forall w \in Y \,, \tag{12}$$

whose solution is simply the discretisation error $e = u - u_h$. We remark that in the context of duality, (12) may be thought of as the *dual–dual* problem and (11) the *dual/dual–dual* equivalence relating (9) to (12), cf. [8]. Thereby, in practice, instead of solving N dual problems, cf. (9), for $z_i \in Y$ with data $J_i(\cdot)$ and then evaluating $\langle R_p(u_h), z_i - z_h \rangle$ to determine the size of the error in the target functional $J_i(\cdot)$, $i = 1, \dots , N$, one can simply solve the error equation (12) for the error $e \in X$ and evaluate $J_i(e)$, for $i = 1, \dots , N$. When $N \gg 1$ this approach is clearly much more computationally efficient than the direct method. However, a disadvantage of this second approach is that while solving the error equation (12) for e gives information concerning the size of the error in the computed target functionals $J_i(\cdot)$, $i = 1, \dots , N$, it does not provide the necessary local information on each element in the computational mesh to guide adaptive mesh refinement when the desired level of accuracy has not been achieved on the current mesh. On the other hand, computing the solution z_i, $i = 1, \dots , N$, to the N dual problems (9), the error representation formulae (10) provides not only information concerning the size of the error in the computed target functionals, but also upon localisation of the right–hand side of (10), local error indicators which can be employed for adaptive mesh design, cf. [4, 5], for example. In the following section we propose a strategy based on solving only two auxiliary problems (the *error equation* (12) and a *dual* problem subject to appropriate data which stems from a specific combined target functional, cf. (14) below) which provide all the necessary information needed to both estimate the size of the error in the computed target functionals, as well as provide local error indicators that can be used to design adaptively refined computational meshes.

3 *A posteriori* estimation for multiple target functionals

In this section we develop the *a posteriori* error analysis in the case when we wish to control the discretisation error in a number of target functionals $J_i(\cdot)$, $i = 1, \dots, N$, of practical interest. To this end, we weaken the requirement (4), and simply insist that the sum of the relative errors in each of the target functionals $J_i(\cdot)$, $i = 1, \dots, N$, is less than TOL. In practice, since $J_i(u)$, $i = 1, \dots, N$, is unknown, we approximate the sum of the relative errors by

$$\sum_{i=1}^{N} |J_i(u) - J_i(u_h)|/|J_i(u_h)| \,,$$

assuming that $J_i(u_h) \neq 0$, for $i = 1, \dots, N$.

Let us begin by assuming that the sign of the error in each target functional $J_i(\cdot)$, $i = 1, \dots, N$, is known. For example, in some applications it may be known from either theoretical considerations or numerical experimentation that under mesh refinement the computed quantity of interest $J_i(u_h)$ is always either smaller or greater than the exact value $J_i(u)$, for $i = 1, \dots, N$. Employing this *a priori* knowledge concerning the convergence of the target functionals, we introduce a *combined* target functional

$$J_c(v) = \sum_{i=1}^{N} \omega_i J_i(v) \,, \tag{13}$$

where $\omega_i = s_i/|J_i(u_h)|$ and s_i denotes the expected signs of the errors $J_i(u) - J_i(u_h)$, $i = 1, \dots, N$, respectively. Thereby, we may now proceed as in Section 2 to derive an error representation formula for the error in the combined target functional $J_c(\cdot)$. To this end, we introduce the following dual problem:

$$\text{find } z_c \in Y \text{ such that } \quad B(w, z_c) = J_c(w) \quad \forall w \in X \,. \tag{14}$$

Thus, as before, we now deduce the following error representation formula

$$J_c(u) - J_c(u_h) = \langle R_p(u_h), z_c - z_h \rangle \qquad \forall z_h \in Y_h \,. \tag{15}$$

Equation (15) now provides both global information concerning the size of the error in the combined target functional $J_c(\cdot)$, as well as local information necessary for adaptive mesh refinement.

In general, the signs s_i, $i = 1, \dots, N$, will not be known *a priori*. Thereby, we must first solve the error equation (12) for e and evaluate $s_i = \operatorname{sgn}(J_i(e))$, $i = 1, \dots, N$. Then, the dual problem (14) may be solved computationally using the predicted values of s_i, $i = 1, \dots, N$, and the resulting error representation formula (15) computed. This approach leads to the solution of only two auxiliary problems, in comparison to the N required for the direct method. In the following section we consider the generalisation of this theory to the case when $J(\cdot)$ is nonlinear and $B(\cdot, \cdot)$ is a semilinear form; as a particular example, we consider the discontinuous Galerkin approximation to a first–order system of nonlinear hyperbolic conservation laws.

4 Application to nonlinear hyperbolic conservation laws

Given a bounded open polyhedral domain Ω in $\mathbb{R}^n$, $n \geq 1$, with boundary $\partial\Omega$, we consider the following problem: find $\mathbf{u} : \Omega \to \mathbb{R}^m$, $m \geq 1$, such that

$$\operatorname{div} \mathcal{F}(\mathbf{u}) = 0 \quad \text{in } \Omega\,, \tag{16}$$

subject to appropriate boundary/initial conditions, where $\mathcal{F} : \mathbb{R}^m \to \mathbb{R}^{m\times n}$ is continuously differentiable. To formulate the discontinuous Galerkin finite element method (DGFEM, for short) for (16), we first introduce some notation. Let $\mathcal{T}_h = \{\kappa\}$ be an admissible subdivision of Ω into open element quadrilateral domains κ; here h is a piecewise constant mesh function with $h(\mathbf{x}) = \operatorname{diam}(\kappa)$ when $\mathbf{x}$ is in element κ. For $p \in \mathbb{N}_0$, we define

$$S_{h,p} = \{\mathbf{v} \in [L_2(\Omega)]^m : \mathbf{v}|_\kappa \in [\mathcal{Q}_p(\kappa)]^m \quad \forall \kappa \in \mathcal{T}_h\}\,,$$

where $\mathcal{Q}_p(\kappa)$ denotes the set of tensor product polynomials of degree at most p in each coordinate direction over κ. Given that $\mathbf{v} \in \left[H^1(\kappa)\right]^m$ for each $\kappa \in \mathcal{T}_h$, we denote by $\mathbf{v}^+$ (resp., $\mathbf{v}^-$) the interior (resp., exterior) trace of $\mathbf{v}$ on $\partial\kappa$. The DGFEM for (16) is defined as follows: find $\mathbf{u}_h \in S_{h,p}$ such that

$$\mathcal{N}(\mathbf{u}_h, \mathbf{v}_h) \equiv \sum_{\kappa\in\mathcal{T}_h} \left\{ -\int_\kappa \mathcal{F}(\mathbf{u}_h)\cdot\nabla\mathbf{v}_h \,\mathrm{d}\mathbf{x} + \int_{\partial\kappa} \mathcal{H}(\mathbf{u}_h^+, \mathbf{u}_h^-, \mathbf{n}_\kappa)\,\mathbf{v}_h^+ \,\mathrm{d}s \right. \\ \left. + \int_\kappa \varepsilon\nabla\mathbf{u}_h\cdot\nabla\mathbf{v}_h\,\mathrm{d}\mathbf{x} \right\} = 0 \tag{17}$$

for all $\mathbf{v}_h \in S_{h,p}$, cf. [4, 9], for example. Here, $\mathcal{H}(\cdot,\cdot,\cdot)$ denotes a *numerical flux* function, assumed to be Lipschitz continuous, consistent and conservative. Further, ε is an artificial viscosity defined by $\varepsilon = C_\varepsilon\, h^{2-\beta}\,|\operatorname{div}\mathcal{F}(\mathbf{u}_h)|$, where $C_\varepsilon \geq 0$ and $0 < \beta < 1/2$ are constants. For $\kappa \in \mathcal{T}_h$ whose boundary intersects $\partial\Omega$, we replace $\mathbf{u}_h^-$ by appropriate boundary/initial conditions on $\partial\kappa \cap \partial\Omega$.

In order to apply the theory of duality–based *a posteriori* error estimation, we first need to introduce some notation. We write $\mathcal{M}(\mathbf{u}, \mathbf{u}_h; \cdot, \cdot)$ to denote the mean–value linearisation of the semi–linear form $\mathcal{N}(\cdot,\cdot)$ given by

$$\mathcal{N}(\mathbf{u}, \mathbf{v}) = \mathcal{N}(\mathbf{u}_h, \mathbf{v}) + \mathcal{M}(\mathbf{u}, \mathbf{u}_h; \mathbf{u} - \mathbf{u}_h, \mathbf{v}) \tag{18}$$

for all $\mathbf{v}$ in some suitably chosen space V such that $S_{h,p} \subset V$. With this notation, we note that the Galerkin orthogonality property of the DGFEM (17) may be expressed as follows: $\mathcal{M}(\mathbf{u}, \mathbf{u}_h; \mathbf{u} - \mathbf{u}_h, \mathbf{v}_h) = 0$ for all $\mathbf{v}_h$ in $S_{h,p}$.

Here, we shall be concerned with controlling the error measured in terms of a set of *nonlinear* target functionals $J_i(\cdot)$, $i = 1, \ldots, N$. Assuming that $J_i(\cdot)$, $i = 1, \ldots, N$, are differentiable, we write, for each i, $1 \leq i \leq N$, $\bar{J}_i(\cdot;\cdot)$ to denote the mean value linearisation of $J_i(\cdot)$ defined by

$$J_i(\mathbf{u}) = J_i(\mathbf{u}_h) + \bar{J}_i(\mathbf{u}, \mathbf{u}_h; \mathbf{u} - \mathbf{u}_h)\,. \tag{19}$$

With this notation, we introduce the dual problem: find $\mathbf{z}_c$ such that

$$\mathcal{M}(\mathbf{u},\mathbf{u}_h;\mathbf{w},\mathbf{z}_c)=\bar{J}_c(\mathbf{u},\mathbf{u}_h;\mathbf{w}) \quad \forall \mathbf{w}\in V\,, \tag{20}$$

where $\bar{J}_c(\mathbf{u},\mathbf{u}_h;\mathbf{w}) = \sum_{i=1}^N \omega_i \bar{J}_i(\mathbf{u},\mathbf{u}_h;\mathbf{w})$, $\omega_i = s_i/|J_i(\mathbf{u}_h)|$ and $s_i = \mathrm{sgn}(\bar{J}_i(\mathbf{u},\mathbf{u}_h;\mathbf{u}-\mathbf{u}_h))$, for $i=1,\ldots,N$. Thereby, assuming the well–posedness of this problem, cf. [4], employing the Galerkin orthogonality property, we deduce the following error representation formula:

$$\begin{aligned} J_c(\mathbf{u})-J_c(\mathbf{u}_h) &= \sum_{i=1}^N \omega_i (J_i(\mathbf{u})-J_i(\mathbf{u}_h)) = \sum_{i=1}^N \omega_i \bar{J}_i(\mathbf{u},\mathbf{u}_h;\mathbf{u}-\mathbf{u}_h) \\ &= \mathcal{M}(\mathbf{u},\mathbf{u}_h;\mathbf{u}-\mathbf{u}_h,\mathbf{z}_c) = \mathcal{M}(\mathbf{u},\mathbf{u}_h;\mathbf{u}-\mathbf{u}_h,\mathbf{z}_c-\mathbf{z}_h) \\ &= -\mathcal{N}(\mathbf{u}_h,\mathbf{z}_c-\mathbf{z}_h) \equiv \mathcal{E}_\Omega(\mathbf{u},\mathbf{u}_h,h,\mathbf{z}_c-\mathbf{z}_h) = \sum_{\kappa\in\mathcal{T}_h} \eta_\kappa^{(\mathrm{I})} \end{aligned} \tag{21}$$

for all $\mathbf{z}_h$ in the finite element space $S_{h,p}$, where

$$\eta_\kappa^{(\mathrm{I})} = \int_\kappa R_h\,(\mathbf{z}_c-\mathbf{z}_h)\,\mathrm{d}\mathbf{x} + \int_{\partial\kappa} r_h\,(\mathbf{z}_c-\mathbf{z}_h)^+\,\mathrm{d}s - \int_\kappa \varepsilon\nabla\mathbf{u}_h\cdot\nabla(\mathbf{z}_c-\mathbf{z}_h)\,\mathrm{d}\mathbf{x}\,,$$

and $R_h|_\kappa = -\mathrm{div}\mathcal{F}(\mathbf{u}_h)$ and $r_h|_\kappa = \mathcal{F}(\mathbf{u}_h)\cdot\mathbf{n}_\kappa - \mathcal{H}(\mathbf{u}_h^+,\mathbf{u}_h^-,\mathbf{n}_\kappa)$ denote the internal and boundary finite element residuals, respectively, defined on each $\kappa\in\mathcal{T}_h$. Thereby, we deduce the following Type I *a posteriori* error bound.

Theorem 1. *Let $\mathbf{u}$ and $\mathbf{u}_h$ denote the solutions of (16) and (17), respectively, and suppose that the dual problem (20) is well–posed. Then,*

$$|J_c(\mathbf{u})-J_c(\mathbf{u}_h)| \le \mathcal{E}_{|\Omega|}(\mathbf{u},\mathbf{u}_h,h,\mathbf{z}_c-\mathbf{z}_h) \equiv \sum_{\kappa\in\mathcal{T}_h} |\eta_\kappa^{(\mathrm{I})}| \quad \forall \mathbf{z}_h\in S_{h,p}\,. \tag{22}$$

We note that the signs s_i, $i=1,\ldots,N$, appearing in the definition of $J_c(\cdot)$ and $\bar{J}_c(\mathbf{u},\mathbf{u}_h;\cdot)$ may be computed by solving the *error equation*:

$$\text{find } \mathbf{e} \text{ such that } \ \mathcal{M}(\mathbf{u},\mathbf{u}_h;\mathbf{e},\mathbf{w}) = -\mathcal{N}(\mathbf{u}_h,\mathbf{w}) \quad \forall \mathbf{w}\,, \tag{23}$$

where $\mathbf{e}=\mathbf{u}-\mathbf{u}_h$, and evaluating the signs of

$$J_i(\mathbf{u})-J_i(\mathbf{u}_h) = \bar{J}_i(\mathbf{u},\mathbf{u}_h;\mathbf{e}), \quad i=1,\ldots,N. \tag{24}$$

As discussed in [4, 5], the local weighting terms involving the difference between the dual solution $\mathbf{z}_c$ and its projection/interpolant $\mathbf{z}_h$ onto $S_{h,p}$ appearing in the Type I bound (22) provide invaluable information concerning the global transport of the error and the interaction of different components of the error. Thereby, we refrain from eliminating the weighting terms involving the (unknown) dual solution $\mathbf{z}_c$ and approximate $\mathbf{z}_c$ numerically. To this end, we perform the linearisations leading to $\mathcal{M}(\mathbf{u},\mathbf{u}_h;\cdot,\cdot)$ and $\bar{J}_c(\mathbf{u},\mathbf{u}_h;\cdot)$ about $\mathbf{u}_h$ and approximate the dual solution $\mathbf{z}_c$ in (20) by a discontinuous Galerkin approximation $\hat{\mathbf{z}}_c$ computed on the same mesh used for $\mathbf{u}_h$, but with a higher degree polynomial, i.e. $\hat{\mathbf{z}}_c\in S_{h,\hat{p}_1}$ with $\hat{p}_1>p$. Similarly, an approximation $\hat{\mathbf{e}}\in S_{h,\hat{p}_2}$, $\hat{p}_2>p$, to $\mathbf{e}$ is computed in an analogous way by solving a linearised error equation. In the following section we set $p=1$ and $\hat{p}_1=\hat{p}_2=2$.

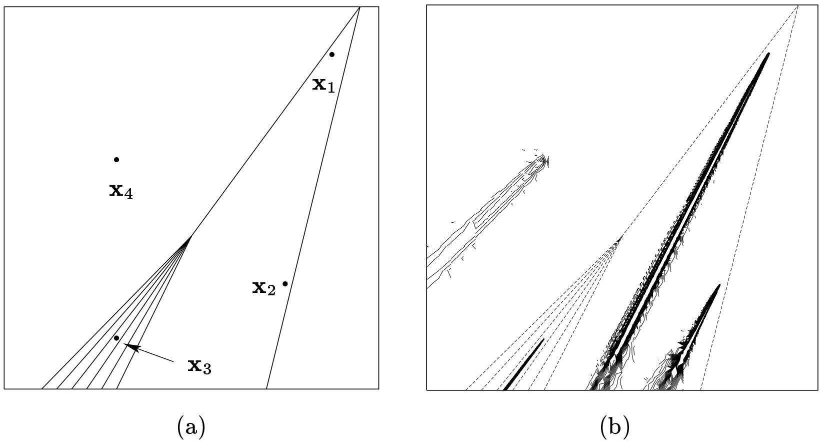


Fig. 1. Burgers' equation: (a) Isolines of the exact (primal) solution u; (b) Isolines of the dual solution z_c

5 Numerical experiment

In this section we present a numerical example, based on [1], to highlight the key aspects of the proposed approach for the *a posteriori* error estimation of multiple target functionals and the subsequent design of adaptively refined computational meshes specifically tailored to the efficient computation of the selected functionals of interest. In particular, we highlight the following issues:

- We shall show that $\bar{J}_i(\mathbf{u}, \mathbf{u}_h; \hat{\mathbf{e}})$, $i = 1, \ldots, N$, evaluated by approximating (23) provides asymptotically sharp approximations to the true errors $J_i(\mathbf{u}) - J_i(\mathbf{u}_h)$, and thereby lead to the correct prediction of the signs s_i.
- The approximate error representation formula computed by replacing $\mathbf{z}_c$ in (21) by $\hat{\mathbf{z}}_c$, provides an asymptotic estimate of the exact error in $J_c(\cdot)$.
- An adaptive refinement algorithm designed using the *Type I* indicator $|\hat{\eta}_\kappa^{(\mathrm{I})}|$, cf. (22) with $\mathbf{z}_c$ replaced by $\hat{\mathbf{z}}_c$, results in meshes that are efficient for computing all the target functionals under consideration. Moreover, we demonstrate the superiority of this approach over a refinement algorithm based on the *Type II* indicator $\eta_\kappa^{(\mathrm{II})} = \|hR_h\|_{L_2(\kappa)} + \|h^{1/2} r_h\|_{L_2(\partial\kappa)}$, cf. [4].

5.1 Burgers' equation

Here, we consider the initial value problem for the inviscid (scalar) Burgers' equation $\partial_t u + \partial_x (u^2/2) = 0$ for $\mathbf{u} \equiv u$, subject to the initial condition

$$u(x,0) \equiv u_0(x) = \begin{cases} 1, & \text{for } x \leq 0.1, \\ -2.5x + 1.25, & \text{for } 0.1 < x \leq 0.3, \\ 0.5, & \text{for } 0.3 < x \leq 0.7, \\ 0, & \text{for } x > 0.7; \end{cases}$$

Table 1. History of the true errors $J_i(e)$, $i = 1, \ldots, 4$, and their estimates $J_i(\hat{e})$ on locally refined meshes of 256 elements (Mesh 1) to 15202 elements (Mesh 9)

Mesh	$J_1(e)$	$J_1(\hat{e})$	$J_2(e)$	$J_2(\hat{e})$	$J_3(e)$	$J_3(\hat{e})$	$J_4(e)$	$J_4(\hat{e})$
1	-1.45e-01	-4.06e-01	7.32e-02	2.65e-02	-2.03e-02	-2.21e-02	-7.26e-06	-7.32e-06
2	-9.96e-03	-6.89e-02	4.83e-03	9.90e-03	-2.11e-03	-2.63e-03	-4.49e-06	-4.51e-06
3	4.96e-02	4.21e-02	-1.67e-02	-1.02e-02	1.50e-04	1.30e-04	-5.13e-07	-5.11e-07
4	-6.51e-03	-8.46e-03	4.53e-04	5.29e-04	4.75e-04	4.90e-04	-5.14e-07	-5.11e-07
5	-2.95e-04	-3.11e-04	-2.65e-05	-2.66e-05	8.45e-05	8.45e-05	-5.67e-07	-5.65e-07
6	-5.83e-06	-6.01e-06	-1.00e-05	-1.01e-05	6.61e-06	6.62e-06	-1.58e-07	-1.58e-07
7	1.63e-06	1.63e-06	4.60e-07	4.60e-07	-5.90e-07	-5.89e-07	-1.58e-07	-1.58e-07
8	4.43e-08	4.41e-08	1.61e-07	1.61e-07	5.33e-07	5.34e-07	9.99e-10	9.98e-10
9	-1.61e-08	-1.62e-08	-3.77e-08	-3.77e-08	1.26e-07	1.26e-07	9.99e-10	9.98e-10

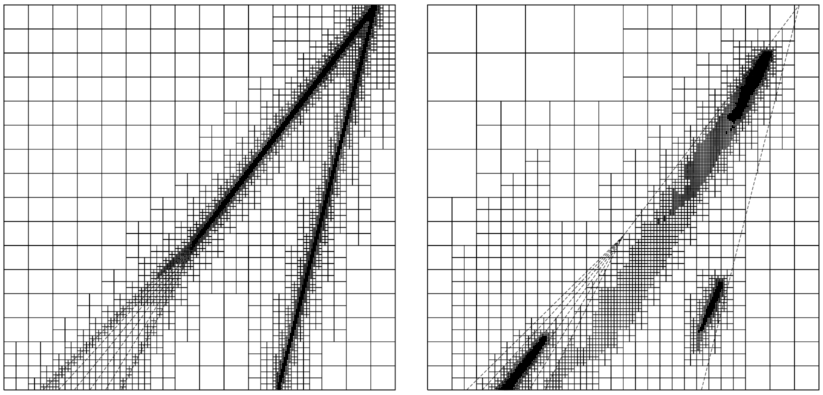
Table 2. Adaptive algorithm for the combined target functional $J_c(\cdot)$

Mesh	# Ele	$J_c(e)$	$\hat{\mathcal{E}}_\Omega = \sum_\kappa \hat{\eta}_\kappa^{(\mathrm{I})}$	$\hat{\mathcal{E}}_\Omega / J_c(e)$	$\hat{\mathcal{E}}_{\lvert\Omega\rvert} = \sum_\kappa \lvert\hat{\eta}_\kappa^{(\mathrm{I})}\rvert$	$\hat{\mathcal{E}}_{\lvert\Omega\rvert} / J_c(e)$
1	256	4.642e-01	7.204e-01	1.55	9.140e-01	1.97
2	415	3.239e-02	1.586e-01	4.90	2.925e-01	9.03
3	712	1.327e-01	1.135e-01	0.86	1.967e-01	1.48
4	1192	1.456e-02	1.842e-02	1.27	5.975e-02	4.10
5	1981	7.563e-04	7.882e-04	1.04	5.951e-03	7.87
6	3316	4.066e-05	4.113e-05	1.01	5.953e-04	14.64
7	5608	5.121e-06	5.119e-06	1.00	1.646e-04	32.14
8	9175	1.124e-06	1.124e-06	1.00	6.345e-05	56.47
9	15202	2.771e-07	2.774e-07	1.00	2.573e-05	92.87

cf. [2, 3]. Fig. 1(a) shows the structure of the solution u: as time increases the slope of the linear declined part of the solution increases until a shock develops that moves to the right. This shock eventually merges with a second right moving shock that originates from the discontinuity in the initial data.

In this example, we shall be interested in the solution evaluated at the series of points $\mathbf{x}_1 = (0.875, 0.875)$, $\mathbf{x}_2 = (0.75, 0.275)$, $\mathbf{x}_3 = (0.3, 2/15)$ and $\mathbf{x}_4 = (0.3, 0.6)$; cf. Fig. 1(a). Defining the (linear) target functionals by $J_i(u) = u(\mathbf{x}_i)$, $i = 1, \ldots, 4$, we note that their true values are given by $J_1(u) = J_2(u) = 0.5$, $J_3(u) = 0.75$ and $J_4(u) = 1$. The dual solution z_c, cf. (20), is depicted in Fig. 1(b); here, we see that z_c consists of a series of 'spikes' or delta functions originating from each of the points of interest which are transported backwards along the single characteristic passing through each $\mathbf{x}_i$, $i = 1, \ldots, 4$, to the inflow boundary.

In Tables 1 and 2, we demonstrate the performance of the proposed adaptive algorithm with $C_\varepsilon = 1/4$ and $\beta = 1/10$, cf. [4]. From Table 1 we see that the values of $\bar{J}_i(u, u_h; \hat{e}) \equiv J_i(\hat{e})$ (since each $J_i(\cdot)$ is linear), $i = 1, \ldots, 4$, evaluated by approximating the error equation (23) provide asymptotically



(a) (b)

Fig. 2. (a) Mesh constructed using the Type II error indicator $\eta_\kappa^{(\mathrm{II})}$ with 17242 elements ($|J_c(u) - J_c(u_h)| = 2.937 \times 10^{-4}$); (b) Mesh constructed using the Type I indicator $|\hat{\eta}_\kappa^{(\mathrm{I})}|$ with 15202 elements ($|J_c(u) - J_c(u_h)| = 2.771 \times 10^{-7}$)

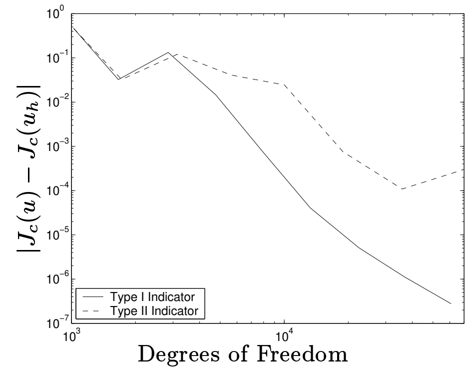


Fig. 3. Convergence of $|J_c(u) - J_c(u_h)|$ using the Type I and Type II indicators

sharp approximations to the exact errors $J_i(e)$, and consequently the correct signs s_i, $i = 1, \ldots, 4$, needed for the computation of the combined target functional $J_c(\cdot)$. Furthermore, in Table 2 we see that the quality of the computed error representation formula $\hat{\mathcal{E}}_\Omega$ is extremely good, in the sense that the effectivity index $\hat{\mathcal{E}}_\Omega/(J_c(u) - J_c(u_h))$ is very close to one, even on relatively coarse meshes. On the other hand, we observe that the Type I *a posteriori* error bound is not sharp, in the sense that the second effectivity index $\hat{\mathcal{E}}_{|\Omega|}/|J_c(u) - J_c(u_h)|$ grows as the mesh is refined.

In Figs. 2(a) and 2(b) we show the meshes produced using the Type II and Type I indicators, respectively. From Fig. 2(a), we see that the mesh produced using $\eta_\kappa^{(\mathrm{II})}$ is concentrated in the region of the computational domain where the shocks are located, as well as in the vicinity of the top and bottom parts of the linear declined part of the solution. In contrast, from Fig. 2(b) we see that there is virtually no refinement in the regions of the computational domain where the shocks are located when $|\hat{\eta}_\kappa^{(\mathrm{I})}|$ is employed. Indeed, most of the refinement takes place upstream of each of the points of interest; this is due to the structure of the dual solution, cf. Fig. 1(b) above. We note that there is much less refinement upstream of the point $\mathbf{x}_4$ than for the other points, since $\mathbf{x}_4$ is located within a region of the computational domain where the primal solution is very smooth; hence, the primal residuals and the resulting local Type I error indicators are very small in this portion of the computational domain. For the same reason only a portion of the region upstream of point $\mathbf{x}_2$ has been refined by the adaptive algorithm.

Finally, in Fig. 3 we compare the true error in the computed functional $J_c(\cdot)$ using the two mesh refinement strategies. Here, we clearly observe the superiority of the Type I *a posteriori* error indicator; on the final mesh the true error in the combined functional is over three orders of magnitude smaller than $|J_c(e)|$ computed on the sequence of meshes produced using $\eta_\kappa^{(\mathrm{II})}$.

References

1. Bangerth, W., Hartmann, R., and Kanschat, G.: `deal.II` *Diff. Equ. Analysis Library, Tech. Rep.* IWR, University of Heidelberg. `http://www.dealii.org`.
2. Hartmann, R. (2001): Adaptive FE Methods for Conservation Equations. In: Freistühler, H., and Warnecke, G. (eds) Hyperbolic Problems: theory, numerics, applications: eighth international conference in Magdeburg, 2000. International series of numerical mathematics **141**:495–503. Birkhäuser, Basel.
3. Hartmann, R. (2002): Adaptive Finite Element Methods for the Compressible Euler Equations, PhD thesis, University of Heidelberg.
4. Hartmann, R., and Houston, P. (2002): Adaptive discontinuous Galerkin finite element methods for nonlinear hyperbolic conservation laws. SIAM J. Sci. Comp. (to appear).
5. Hartmann, R., and Houston, P. (2001): Adaptive discontinuous Galerkin finite element methods for the compressible Euler equations. Preprint 2001-42, (SFB 359), IWR Heidelberg, (submitted for publication).
6. Hartmann, R., and Houston, P. (2002): Goal–oriented a posteriori error estimation for compressible fluid flows. In *Proceedings of ENUMATH 2001*, submitted.
7. Houston, P., Hartmann, R., and Süli, E. (2001): Adaptive discontinuous Galerkin finite element methods for compressible fluid flows. In: Baines, M. (ed) Numerical methods for Fluid Dynamics VII, ICFD, pages 347–353.
8. Houston, P., Senior, B., and Süli, E. (2001): *hp*–Discontinuous Galerkin finite element methods for hyperbolic problems: error analysis and adaptivity. Int. J. Numer. Meth. Fluids (to appear).
9. Jaffre, J., Johnson, C., and Szepessy, A. (1995): Convergence of the discontinuous Galerkin finite element method for hyperbolic conservation laws. Math. Models Methods Appl. Sci., **5**(3):367–386.
10. Larson, M. and Barth, T.J. (2000): In: Cockburn, B., Karniadakis, G.E., Shu C.-W. (eds) Discontinuous Galerkin Finite Element Methods. Lecture Notes in Computational Science and Engineering, **11**:363–368. Springer–Verlag.