A POSTERIORI CONTROL OF MODELING ERRORS AND DISCRETIZATION ERRORS *

MALTE BRAACK † AND ALEXANDRE ERN ‡

Key words. model adaptivity, a posteriori estimates, finite elements, Galerkin methods, duality

AMS subject classifications. 65C20, 65N15, 65N30, 65N50, 65J15

Abstract. We investigate the concept of dual-weighted residuals for measuring model errors in the numerical solution of nonlinear partial differential equations. The method is first derived in the case where only model errors arise and then extended to handle simultaneously model and discretization errors. We next present an adaptive model/mesh refinement procedure where both sources of errors are equilibrated. Various test cases involving Poisson equations and convection-diffusion-reaction equations with complex diffusion models (highly oscillatory diffusion coefficient, nonlinear turbulent viscosity, multicomponent diffusion matrix) confirm the reliability of the analysis and the efficiency of the proposed methodology.

1. Introduction. In many fields of scientific computing, the underlying equations are well-known but may involve models of different complexity, scales and accuracy. In various cases, the most accurate and validated model can not be chosen in numerical simulations because of the large amount of computational costs. For instance, in the field of combustion and reactive flows, the choice of diffusion models in gas mixtures is not straightforward. Although multicomponent diffusion models are accepted to be accurate [6], simpler and less accurate models, e.g. Fick's law, are widely used in practical two- and three-dimensional simulations. Therefore, it is desirable to apply the complex model just in those regions where necessary; for instance in the flame front where a complex balance of reaction, convection and diffusion phenomena takes place. Another field is the computation of Darcy flows in porous media, where fluctuations on multiple scales of phenomenological parameters such as the hydraulic conductivity lead to hetereogeneous diffusion. The arising question is whether to resolve all characteristic scales or not.

An appealing strategy to achieve a compromise between accuracy of the model and computational costs is to increase adaptively the complexity of the underlying model only whenever useful. To this purpose, it is necessary to derive estimates for the model error and then to implement them in the framework of an adaptive method. Earlier work on model error estimation has been done by Stein [11], Schwab [1] and the group of Oden [9, 10]. Fatone et. al. [7] analyzed recently a method for coupling Ossen and Navier-Stokes equations on the basis of a given domain decomposition. In this work, we propose a mathematical based algorithm to obtain a posteriori model error control for such kinds of model adaptation. Our method is based on the solution of an associated dual problem in order to measure the influence of the model on a user-defined output functional of the numerical solution. The approach is an extension to modeling errors of a posteriori discretization error control by dual weighted residuals [2]. Such an approach is indeed very general, so that different sources of errors with

*This work has been supported by the German Research Association (DFG) through the SFB 359 "Reactive Flow, Diffusion and Transport" at the University of Heidelberg.

†Institut für Angewandte Mathematik, Universität Heidelberg, INF 294, D-69120 Heidelberg, Germany (malte.braack@iwr.uni-heidelberg.de).

‡CERMICS, Ecole nationale des ponts et chaussées, 6 et 8, av. Blaise Pascal, 77455 Marne la Vallée cedex 2, France (ern@cermics.enpc.fr).

respect to an arbitrary functional output can be estimated, not only discretization errors.

A second issue addressed in this work is how to balance local model modification with mesh refinement techniques. To this purpose, we propose an error estimator for measuring model and discretization errors at the same time and discuss its implementation in the framework of an adaptive algorithm that achieves an appropriate equilibrium between model and discretization error. For instance, instead of over-refining the mesh and using a crude model, it is possible to control both sources of error simultaneously, and refine the mesh or change the model locally to higher accuracy.

To start with, we consider a linear equation with linear model and linear output functional. The weak solution u for an accurate model is determined by the partial differential equation in variational formulation in a Hilbert space V:

$$u \in V: \quad a(u,\phi) + d(u,\phi) = (f,\phi) \quad \forall \phi \in V\,. \tag{1.1}$$

Here, $a(u,\phi)$ represents the considered equation and includes a certain (simple) model while $d(u,\phi)$ stands for the part of the model which is expensive to compute, or which may be neglected due to other reasons. Given a functional output $j(u)$, we ask for the influence of neglecting the part $d(u,\phi)$ on $j(u)$. We assume in this section that the functional j is a linear mapping $j: V \to I\!R$ and that the forms $a(\cdot,\cdot)$ and $d(\cdot,\cdot)$ are bilinear.

The reduced system is the equation for an approximate solution u_m:

$$u_m \in V: \quad a(u_m,\phi) = (f,\phi) \quad \forall \phi \in V\,. \tag{1.2}$$

For the model error $e_u = u - u_m$ obviously holds a "perturbed Galerkin orthogonality"

$$a(e_u,\phi) = -d(u,\phi) \quad \forall \phi \in V\,.$$

In order to get control about the error respect to the functional j, we introduce the dual problem

$$z \in V: \quad a(\phi,z) + d(\phi,z) = j(\phi) \quad \forall \phi \in V\,. \tag{1.3}$$

Using the dual solution z gives the error identity

$$j(e_u) \;=\; a(e_u,z) + d(e_u,z) \;=\; -d(u,z) + d(e_u,z) \;=\; -d(u_m,z)\,.$$

Our main implicit assumption concerning computational costs is that the overhead for solving the full variational problem (1.1) rather than the reduced one (1.2) is much higher than a single evaluation of the bilinear form $d(\cdot,\cdot)$. This assumption is reasonable since the variational problems (1.1) and (1.2) are generally solved by iterative methods that involve a substantial number of residual evaluations. We may thus consider that solving the dual problem (1.3) is not pratical, since it involves the (expensive) model $d(\phi,z)$ which was omitted in the primal problem. Therefore, we approximate z by the dual solution z_m of the reduced dual problem:

$$z_m \in V: \quad a(\phi,z_m) = j(\phi) \quad \forall \phi \in V\,.$$

This leads to the representation

$$\begin{aligned} j(e_u) = -d(u_m,z) &= -d(u_m,z_m) - d(u_m,z-z_m) \\ &= -d(u_m,z_m) - \frac{1}{2}\big\{d(u_m,z-z_m) + d(u-u_m,z_m)\big\}\,, \end{aligned}$$

since $d(u_m, z) = d(u, z_m)$. Denoting by $\|\cdot\|$ the norm of V and using a similar notation for linear and bilinear forms on V and $V \times V$ respectively, the first term in the error representation may be estimated as

$$d(u_m, z_m) \leq \|d\| \, \|u_m\| \, \|z_m\| \, .$$

Furthermore, assuming that the bilinear form $a(\cdot, \cdot)$ satisfies a stability property in the form $\exists \alpha > 0$, $\forall u \in V$, $\|u\| \leq \alpha \, \|a(u, \cdot)\|$, we deduce from the "dual perturbed Galerkin orthogonality" $a(z - z_m, \phi) = -d(\phi, z)$, $\forall \phi \in V$, the estimate

$$\|z - z_m\| \leq \alpha \, \|d\| \, \|z\| \, .$$

Therefore, the second term in the error representation may be estimated as

$$d(u_m, z - z_m) \leq \alpha \, \|d\|^2 \, \|u_m\| \, \|z\| \, .$$

As a result, at first order in $\|d\|$, the error $j(e_u)$ may be approximated as

$$j(e_u) \approx -d(u_m, z_m) \, .$$

The outline of the paper is as follows: In Section 2 we derive an a posteriori estimate for the model error in the more general case where $a(\cdot, \cdot)$ and $d(\cdot, \cdot)$ are allowed to be semi-linear and the functional $j(\cdot)$ can be nonlinear. In Section 3 we combine the discretization error and model error. A strategy for balancing local model modification with mesh-size adaptation on the basis of the a posteriori error estimates is presented in Section 4. The numerical results in Section 5 contain three model problems with different diffusion models.

2. Nonlinear model error. In this section, we use the technique introduced in [2] for a posteriori error estimation of the discretization error. This will be presented for measuring model errors.

The nonlinear primal problem we are interested in is given by

$$u \in V : \quad a(u)(\phi) + d(u)(\phi) \, = \, (f, \phi) \quad \forall \phi \in V \, . \tag{2.1}$$

The semi-linear forms $a(u)(\cdot)$ and $d(u)(\cdot)$ are linear in the second argument but may be nonlinear in u. The directional derivatives of $a(u)(\cdot)$ and $d(u)(\cdot)$ will be denoted by $a'(u)(\cdot, \cdot)$ and $d'(u)(\cdot, \cdot)$, respectively. The semi-linear form

$$a'(u)(v, \phi) \, = \, \lim_{\epsilon \to 0} \frac{1}{\epsilon} \, \{a(u + \epsilon v)(\phi) - a(u)(\phi)\}$$

is linear in v and ϕ. The second and third directional derivatives are denoted by $a''(u)(\cdot, \cdot, \cdot)$ and $a'''(u)(\cdot, \cdot, \cdot, \cdot)$, respectively. In the general case of a nonlinear output functional $j(u)$, the corresponding dual problem we will use in the analysis is the following:

$$z \in V : \quad a'(u)(\psi, z) + d'(u)(\psi, z) = j'(u)(\psi) \quad \forall \psi \in V \, . \tag{2.2}$$

The primal solution $u_m \in V$ and dual solution $z_m \in V$ of the reduced problems are given by:

$$u_m \in V : \quad a(u_m)(\varphi) = (f, \varphi) \quad \forall \varphi \in V \, , \tag{2.3}$$

$$z_m \in V : \quad a'(u_m)(\psi, z_m) = j'(u_m)(\psi) \quad \forall \psi \in V \, . \tag{2.4}$$

These variational problems will be formulated as optimization problems. The primal and dual solutions will be expressed by the variables $x = (u, z) \in X := V \times V$ and $x_m = (u_m, z_m) \in X$. In the variational space X we consider the functionals

$$L(x) = L_m(x) + \delta L(x)\,, \tag{2.5}$$

$$L_m(x) = j(u) + (f, z) - a(u)(z)\,, \tag{2.6}$$

$$\delta L(x) = -d(u)(z)\,. \tag{2.7}$$

The derivative of L applied to a test function $y = (\varphi, \psi) \in X$ is:

$$\begin{aligned} L'(x)(y) = j'(u)(\varphi) - a'(u)(\varphi, z) - d'(u)(\varphi, z) \\ + (f, \psi) - a(u)(\psi) - d(u)(\psi)\,. \end{aligned}$$

Obviously the original primal and dual problems (2.1) and (2.2) and the reduced primal and dual problems (2.3) and (2.4) consist in finding the stationary points $x = (u, z)$ and $x_m = (u_m, z_m)$ of L and L_m, respectively:

$$x \in X: \quad L'(x)(y) = 0 \quad \forall y \in X\,, \tag{2.8}$$

$$x_m \in X: \quad L_m'(x_m)(y) = 0 \quad \forall y \in X\,. \tag{2.9}$$

Furthermore, the target quantities are given by evaluation of L and L_m at these stationary points:

$$\begin{aligned} j(u) &= L(x)\,, \\ j(u_m) &= L_m(x_m)\,. \end{aligned}$$

We get the following error representation:

THEOREM 2.1. *If the semi-linear forms $a(u)(\cdot)$ and $d(u)(\cdot)$, and the functional $j(u)$ are sufficiently differentiable with respect to u, then it holds*

$$j(u) - j(u_m) = -d(u_m)(z_m) - \frac{1}{2}\{d(u_m)(e_z) + d'(u_m)(e_u, z_m) - R\}\,,$$

where R is cubic in the error $e = \{e_u, e_z\} := \{u - u_m, z - z_m\}$:

$$R := \int_0^1 L'''(x_m + \lambda e)(e, e, e) \cdot \lambda(1 - \lambda)\, d\lambda\,. \tag{2.10}$$

Proof.

$$\begin{aligned} j(u) - j(u_m) &= L(x) - L(x_m) + \delta L(x_m) \\ &= -d(u_m)(z_m) + \int_0^1 L'(x_m + \lambda e)(e)\, d\lambda\,. \end{aligned}$$

Applying the trapezoidal rule, we get

$$\int_0^1 L'(x_m + \lambda e)(e)\, d\lambda = \frac{1}{2}\{L'(x)(e) + L'(x_m)(e) + R\}\,,$$

with remainder term R given by (2.10). Because of (2.8) and (2.9), we have

$$\begin{aligned} L'(x)(e) + L'(x_m)(e) &= L'(x_m)(e) \\ &= L_m'(x_m)(e) + \delta L'(x_m)(e) \\ &= \delta L'(x_m)(e) \\ &= -d(u_m)(e_z) - d'(u_m)(e_u, z_m)\,. \end{aligned}$$

This completes the proof. □

From a numerical viewpoint, it is useful to obtain some a priori estimates of the various terms arising in the error representation obtained in Theorem 2.1. Assuming that the functional L' satisfies a stability property of the form

$$|x_1 - x_2\|_X \leq \alpha \, \|L'(x_1) - L'(x_2)\|_{X'}$$

with $\alpha > 0$, we obtain

$$\|e\|_X = \|x - x_m\|_X \leq \alpha \, \|L'(x_m)\|_{X'} = \alpha \, \|\delta L'(x_m)\|_{X'},$$

thanks to the relations (2.8) and (2.9). Assuming that the perturbation $d(\cdot)(\cdot)$ and its derivatives are sufficiently small, we may write $\|\delta L'(x_m)\|_{X'} \leq c(d) \, \|x_m\|_X$ with a constant $c(d) \ll 1$. Therefore, the error e is first order in $c(d)$ and the contributions in R are thus of third order in $c(d)$. Furthermore, the terms $d(u_m)(e_z)$ and $d'(u_m)(e_u, z_m)$ are quadratic in $c(d)$ since they involve the semi-linear form $d(\cdot)(\cdot)$ and the error e. As a result, the error representation of Theorem 2.1 will be used numerically by neglecting these terms, yielding the simple error estimator η_m:

$$j(u) - j(u_m) \approx \eta_m := -d(u_m)(z_m) \, .$$

In Section 5, we show by numerical experiments that this estimator is reliable and efficient for several test problems. However, if more numerical effort in the error estimate is feasible, in addition one may solve local dual problems involving the accurate model $d(\cdot, \cdot)$.

3. Combining discretization error and model error. In order to balance model and discretization error, we have to derive also an a posteriori estimator of the discretization error. For this, let $V_h \subset V$ a finite dimensional subspace and $u_{hm} \in V_h$ the discrete solution involving both types of error:

$$u_{hm} \in V_h : \quad a(u_{hm})(\phi) = (f, \phi) \quad \forall \phi \in V_h \, .$$

By $\mathcal{T}_h$ we denote the corresponding triangulation of the computational domain $\Omega \subset \mathbb{R}^d$, with $d = 2, 3$. Possible homogeneous Dirichlet conditions are already included in the choice of the spaces V_h and V. The operators L and L_m are chosen as above in (2.5)-(2.7). The difference is the definition of the discrete solution $x_{hm} = \{u_{hm}, z_{hm}\} \in X_h = V_h \times V_h$:

$$x_{hm} \in X_h : \quad L'_m(x_{hm})(y) = 0 \quad \forall y \in X_h \, ,$$

which corresponds to

$$u_{hm} \in V_h : \quad a(u_{hm})(\varphi) = (f, \varphi) \quad \forall \varphi \in V_h \, , \tag{3.1}$$

$$z_{hm} \in V_h : \quad a'(u_{hm})(\psi, z_{hm}) = j'(u_{hm})(\psi) \quad \forall \psi \in V_h \, . \tag{3.2}$$

For the formulation of the error representation, we use the following notation for the primal and dual residual with respect to the reduced model and for test functions $(\varphi, \psi) \in X$:

$$\rho(u_{hm})(\varphi) := (f, \varphi) - a(u_{hm})(\varphi) \, ,$$
$$\rho^*(u_{hm}, z_{hm})(\psi) := j'(u_{hm})(\psi) - a'(u_{hm})(\psi, z_{hm}) \, .$$

THEOREM 3.1. *Under the same conditions as for Theorem 2.1, we have*

$$\begin{aligned} j(u) - j(u_{hm}) = & -d(u_{hm})(z_{hm}) \\ & + \frac{1}{2}\{\rho(u_{hm})(z - i_h z) + \rho^*(u_{hm}, z_{hm})(u - i_h u)\} \\ & - \frac{1}{2}\{d(u_{hm})(e_z) + d'(u_{hm})(e_u, z_{hm})\} + \frac{1}{2}R\,, \end{aligned}$$

where $e = \{e_u, e_z\} = \{u - u_{hm}, z - z_{hm}\}$, $i_h : V \to V_h$ *is an arbitrary interpolation operator and* R *is given by (2.10).*

Proof. Similar to the previous proof, one gets

$$j(u) - j(u_{hm}) = -d(u_{hm})(z_{hm}) + \frac{1}{2}(L'(x_{hm})(e) + R)\,.$$

Because in the discrete case $L'_m(x_{hm})(y)$ does not vanish for arbitrary test functions $y \in X$, we get

$$\begin{aligned} L'(x_{hm})(e) &= L'_m(x_{hm})(e) + \delta L'(x_{hm})(e) \\ &= \rho(u_{hm})(z - z_{hm}) + \rho^*(u_{hm}, z_{hm})(u - u_{hm}) \\ &\quad - d(u_{hm})(e_z) - d'(u_{hm})(e_u, z_{hm})\,. \end{aligned}$$

Due to (3.1) and (3.2) we may substract arbitrary discrete test functions in the residual terms. Especially, we may use an arbitrary interpolation $i_h u$ of u and $i_h z$ of z:

$$\begin{aligned} 0 = \rho(u_{hm})(z_{hm}) &= \rho(u_{hm})(i_h z)\,, \\ 0 = \rho^*(u_{hm}, z_{hm})(u_{hm}) &= \rho^*(u_{hm}, z_{hm})(i_h u)\,. \end{aligned}$$

This gives the assertion. □

In order to use numerically the error representation derived in Theorem 3.1, we have to approximate various terms. As in the linear case, we will neglect the higher-order terms in e with respect to the model $d(\cdot)(\cdot)$. Furthermore, we have to approximate the interpolation errors $u - i_h u$ and $z - i_h z$. An efficient possibility to do this, is the recovery process of the computed quantities by higher-order polynomials, see [2, 4]. For instance, in the case of triangles ($d = 2$) or tetraedrons ($d = 3$) and when V_h consists of piecewise linear elements, quadratic interpolation may be used. For quadrilaterals and piecewise d-linear elements, the interpolation can be done on d-quadratic elements. To this purpose, we need certain restrictions on the meshes used. We assume that the triangulation $\mathcal{T}_h$ is organized patch-wise: $\mathcal{T}_h$ results from a global refinement of a mesh $\mathcal{T}_{2h}$. Note that $\mathcal{T}_h$ contains in 2D twice as much hanging nodes as $\mathcal{T}_{2h}$. For illustration in the two-dimensional case using quadrilaterals with hanging nodes, see Figure 3.1. The same construction is possible in three dimensions. Let

$$i_{2h}^{(2)} : V_h \to V_{2h}^{(2)}$$

be the quadratic interpolation of piecewise bilinears on $\mathcal{T}_h$ onto biquadratic finite elements on $\mathcal{T}_{2h}$. The interpolation errors will be numerically approximated by

$$\begin{aligned} z - i_h z &\approx i_{2h}^{(2)} z_{hm} - z_{hm}\,, \\ u - i_h u &\approx i_{2h}^{(2)} u_{hm} - u_{hm}\,. \end{aligned}$$

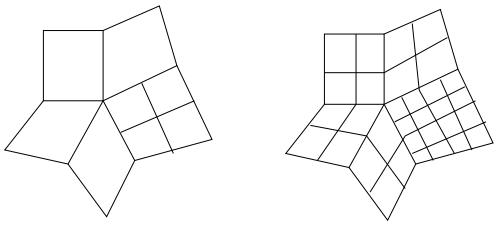

FIG. 3.1. *Possible triangulation for approximation of the interpolation error on quadrilaterals with hanging nodes $\mathcal{T}_{2h}$ (left), $\mathcal{T}_h$ (right).*

Without modeling error, this approximation is usually observed to be accurate enough.

Taking into account that the residuals $\rho(u_{hm})(\phi)$ and $\rho^*(u_{hm}, z_{hm})(\phi)$ with respect to a discrete test function $\phi \in V_h$ vanish, leads to the following estimator consisting of two parts

$$
\begin{aligned}
j(u) - j(u_{hm}) &\approx \eta_h + \eta_m \,, \\
\eta_h &:= \frac{1}{2}\{\rho(u_{hm})(i_{2h}^{(2)} z_{hm}) + \rho^*(u_{hm}, z_{hm})(i_{2h}^{(2)} u_{hm})\} \,, \\
\eta_m &:= -d(u_{hm})(z_{hm}) \,.
\end{aligned} \tag{3.3}
$$

The part η_h of the estimator can be considered as contributions of the discretization and the part η_m measures the influence of the model. For complex models, the evaluation of η_m may be expensive. However, the gain of an adaptive algorithm with local model modification becomes substantial for nonlinear problems, since we do not need to include the (global) detailed model neither in each residual evaluation nor in the Jacobian.

4. Adaptivity. In order to use the information (3.3) for changing the model locally or to change the mesh size, we have to localize the estimator. Then we design an adaptive process in order to balance the two sources of errors.

4.1. Localization of the estimator. We look at the nodal contributions of $\eta_h + \eta_m$. By n we denote the number of nodes of the triangulation $\mathcal{T}_h$ for the finite element space V_h. Let $U, Z \in \mathbb{R}^n$ be the vector of nodal values of u_{hm} and z_{hm}, respectively. The estimator of the modeling error η_m can be expressed as a scalar product of Z and the vector $\Lambda = \{\Lambda_i\}$ built by the model residuals with respect to the Lagrangian nodal basis $\{\phi_i\} \subset V_h$:

$$
\begin{aligned}
\Lambda_i &= d(u_{hm})(\phi_i) \,, \\
\eta_m &= \langle \Lambda, Z \rangle \,.
\end{aligned}
$$

Here, we use the notation $\langle \cdot, \cdot \rangle$ for the l_2 scalar product in $\mathbb{R}^n$. We localize the modeling error by the upper bound

$$
|\eta_m| \le \sum_{i=1}^{n} |\Lambda_i Z_i| \,.
$$

For the part of the estimator corresponding to the discretization error, we consider $(d-)$ linear finite elements. For each node $\mathcal{N}_i$ of the triangulation of $\mathcal{T}_h$, we have the bilinear nodal function $\phi_i \in V_h$ and a quadratic nodal function $\phi_i^{(2)} := i_{2h}^{(2)} \phi_i \in V_h^{(2)}$.

Let $U \in \mathbb{R}^n$ the vector of nodal values of u_{hm}. Let Ψ_i and Ψ_i^* be the primal and dual residual contributions with respect to the quadratic basis $\{\phi_i^{(2)}\}$:

$$\begin{aligned}
\Psi_i &= \rho(u_{hm})(\phi_i^{(2)})\,, \\
\Psi_i^* &= \rho^*(u_{hm}, z_{hm})(\phi_i^{(2)})\,, \\
\eta_h &= \frac{1}{2}\left\{\langle \Psi, Z\rangle + \langle \Psi^*, U\rangle\right\}\,.
\end{aligned}$$

Direct localization of the terms $\langle \Psi, Z\rangle$ and $\langle \Psi^*, U\rangle$ results in general in a large overestimation of the error due to the oscillatory behaviour of the residual terms. This can be reduced by a filter as described now.

We introduce the nodal interpolation operator $i_{2h}^h : V_h \to V_{2h}$ and the filtering operator $\pi_h : V_h \to V_h$, defined by

$$\pi_h \phi \equiv \phi - i_{2h}^h \phi\,,$$

giving the small-scale linear fluctuations. We denote the nodal vectors of the filtered primal solution $\pi_h u_h$ and dual solution $\pi_h z_h$ by U^π and Z^π, respectively,

$$\pi_h u_{hm} = \sum_{i=1}^n \phi_i U_i^\pi\,, \quad \pi_h z_{hm} = \sum_{i=1}^n \phi_i Z_i^\pi\,.$$

The error estimator may be localized on the basis of the following Proposition.

PROPOSITION 4.1. *The estimator for the discretization error η_h in (3.3) can be computed by the filtered discrete solutions:*

$$\eta_h = \frac{1}{2}\Big\{\langle \Psi, Z^\pi\rangle + \langle \Psi^*, U^\pi\rangle\Big\}\,.$$

Proof. The quadratic interpolation operator $i_{2h}^{(2)}$ is obviously the identity on V_{2h}. This implies

$$i_{2h}^{(2)} i_{2h}^h \phi = i_{2h}^h \phi \quad \forall \phi \in V_h\,.$$

Therefore, it holds for all $\phi \in V_h$,

$$\begin{aligned}
i_{2h}^{(2)}\pi_h\phi - \pi_h\phi &= i_{2h}^{(2)}\phi - i_{2h}^{(2)} i_{2h}^h \phi - (\phi - i_{2h}^h\phi) \\
&= i_{2h}^{(2)}\phi - \phi\,.
\end{aligned}$$

This implies for the functions of the nodal basis $\{\phi_i\}$ of V_h that

$$\phi_i^{(2)} - \phi_i = i_{2h}^{(2)}\pi_h\phi_i - \pi_h\phi_i\,.$$

Therefore, the statement is a direct consequence of the following identities:

$$\begin{aligned}
\rho(u_{hm})(i_{2h}^{(2)} z_{hm}) &= \rho(u_{hm})(i_{2h}^{(2)} z_{hm} - z_{hm}) \\
&= \sum_{i=1}^n \rho(u_{hm})(\phi_i^{(2)} - \phi_i)\cdot Z_i \\
&= \sum_{i=1}^n \rho(u_{hm})(i_{2h}^{(2)}\pi_h\phi_i - \pi_h\phi_i)\cdot Z_i
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{i=1}^{n} \rho(u_{hm})(i_{2h}^{(2)}\phi_i - \phi_i) \cdot Z_i^{\pi} \\
&= \sum_{i=1}^{n} \rho(u_{hm})(\phi_i^{(2)}) Z_i^{\pi} \\
&= \langle \Psi, Z^{\pi} \rangle .
\end{aligned}
$$

□

This gives us the following localized estimator:

PROPOSITION 4.2. *With the computable quantities on each node*

$$
\begin{aligned}
\eta_{h,i} &:= \frac{1}{2} |\Psi_i Z_i^{\pi} + \Psi_i^* U_i^{\pi}| , \\
\eta_{m,i} &:= |\Lambda_i Z_i| ,
\end{aligned}
$$

we have the following upper bound for the estimator

$$
|\eta_h + \eta_m| \leq \sum_{i=1}^{n} (\eta_{h,i} + \eta_{m,i}) .
$$

To sum up, the computational extra work for evaluating the estimator and getting the nodal contributions consists of the following steps:

- Solving the (linear) dual problem for getting z_{hm}. There is no need to assemble a new Jacobian. Instead, the matrix corresponding to the primal problem can be simply transposed.
- Evaluation of the model with respect to the linear nodal basis $d(u_{hm})(\phi_i)$ for getting Λ. This can be an expensive step for complex models. Taking the scalar product with the vector Z gives the model estimator η_m.
- Evaluation of the primal and dual residuals Φ and Φ^* with respect to the quadratic test functions and taking the scalar product with Z^{π} and U^{π}, respectively, for getting η_h.

4.2. Balancing model and mesh size adaptation. On the basis of the local indicators $\{\eta_{h,i}, \eta_{m,i}\}$, an adaptive process can be designed. For refining the mesh, standard algorithms can be used. We refer to [] for an overwiev of standard procedures. Since the values are nodal-based, the information has to be shifted to the cells. This can be done in a straightforward way or by applying more sophisticated algorithms such that more smoothing is achieved.

The strategy for "refining the model" strongly depends on the problem. The considered semi-linear forms $a(\cdot)(\cdot)$ and $d(\cdot)(\cdot)$ may change in each adaptive step, so that portions of $d(\cdot)(\cdot)$ are successively (and locally) shifted to $a(\cdot)(\cdot)$. In step i, the equations to be solved are of the form:

$$
\begin{aligned}
u_i \in V_i : \quad & a_i(u_i)(\phi) = (f, \phi) \quad \forall \phi \in V_i , \\
z_i \in V_i : \quad & a_i'(u_i)(\phi, z_i) = j(\phi) \quad \forall \phi \in V_i .
\end{aligned}
$$

The discrete subspace $V_i \subset V$ depends on i because of mesh refinement, whereas the semi-linear form $a_i(\cdot)(\cdot)$ depends on i because of model modification. The neglected part of the model in iteration i is

$$
d_i(u_i)(\phi) = a(u_i)(\phi) - a_i(u_i)(\phi) + d(u_i)(\phi) .
$$

In order to balance both sources of errors, it seems reasonable to work with modified error indicators for the refinement of the mesh and the model:

$$\widetilde{\eta}_{m,i} = \begin{cases} \eta_{m,i} & \text{if } \eta_{m,i} \geq \alpha \eta_{h,i} \\ 0 & \text{otherwise} \end{cases}$$

$$\widetilde{\eta}_{h,i} = \begin{cases} \eta_{h,i} & \text{if } \eta_{h,i} \geq \alpha \eta_{m,i} \\ 0 & \text{otherwise} \end{cases}$$

The parameter α is set to 0.2 in the numerical results presented in the next section. The mesh adaptation strategy we used is based on an optimization strategy, see [3, 5] for details. The change from the crude to the detailed model is done using an "error balancing" technique. Cells are marked whenever

$$\widetilde{\eta}_{m,i} > \frac{1}{n} TOL \,,$$

with a tolerance based on the l_1-norm of the indicators, $TOL = 0.5 \|\widetilde{\eta}_m\|_1$. The detailed model is then used on marked cells in step $i+1$.

5. Numerical results. We consider two examples of Poisson-type problems and one convection-diffusion-reaction problem. The diffusion coefficients may depend in different ways on the solution. In order to decouple the error contributions of the model and the discretization, the first example includes only the modeling error. Examples 2 and 3 include both sources of errors. These examples do not really involve very complex models, because they are not chosen for illustration of CPU time savings when model adaptivity is used. The scope of the present work is to validate the accuracy of the estimator and to illustrate the mechanism of error balancing. The application to large-scale problems including very complicated and expensive models will be addressed in a forthcoming work.

5.1. Highly oscillatory diffusion coefficient. As a first example, we consider the (discrete) Poisson problem in a L-shaped domain $\Omega = [(-1,1) \times (0,1))] \cup [(0,1) \times (-1,1)] \subset I\!R^2$:

$$-\mathrm{div}\,(\mu \nabla u) = f \,, \tag{5.1}$$

with homogeneous Dirichlet conditions, $u \equiv 0$ on $\Gamma_1 \subset \partial\Omega$ (vertical boundaries of Ω), homogeneous Neumann conditions on $\Gamma_2 = \partial\Omega \setminus \Gamma_1$ (horizontal boundaries of Ω), and constant right-hand side, $f \equiv 1$. The diffusion coefficient μ oscillates in space,

$$\mu(x,y) = 1 + \beta e^{-\alpha\gamma} \,, \quad \gamma = \min\{\cos(\pi k x), \cos(\pi k y)\} \,,$$

whereas for the simplest model, the diffusion coefficient is constant, $\mu = 1$. The parameter β is the amplification factor, α localizes the perturbation to a narrow stripe and k is the frequency in x and y direction. The parameters are set to $\alpha = 700$, $\beta = 100$ and $k = 0.5$. Physically, this problem may be viewed as a model for a Darcy flow in a porous medium with cracks located along the lines $\{x = \pm\frac{1}{2}\}$ and $\{y = \pm\frac{1}{2}\}$ where the diffusion coefficient (the hydraulic conductivity) is very high. The crude model consists in neglecting the presence of the cracks.

For this first test problem, we focus on the model error on a fixed mesh with 12,545 nodes, so that we compare u_{hm} with the numerical solution u_h computed with the exact model. This latter solution is presented in Figure 5.1. The perturbation in the diffusion coefficient results in small oscillations in $x-$ and $y-$direction.

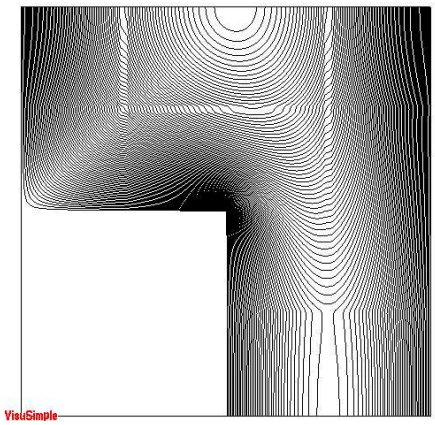

FIG. 5.1. *Solution of the model problem with highly oscillatory diffusion coefficient.*

The output functionals we investigate are the mean value over Ω and a point value at $x_0 = (0, 0.5)$:

$$j_1(u) = \int_\Omega u \, dx \,,$$
$$j_2(u) = u(x_0) \,.$$

We start with the constant diffusion model over the whole domain. After computing the solution u_{hm}, we evaluate the estimator η_m for $j(u_h) - j(u_{hm})$, and decide on the basis of that information which cells we have to mark. In the next iteration, we apply the exact model $\mu(x)$ only on the marked cells. Because of the sharp variations in the diffusion coefficient, we use an expensive quadrature rule on the marked cells: 100-point Gauss (10×10). On the unmarked cells, a composite lower-order quadrature rule (4-point Gauss) is sufficient because of constant diffusion.

iter	fraction of exact model	η_m	$j_1(u_h - u_{hm})$	I_{eff}
1	0.	-5.369e-02	-3.787e-02	1.41
2	0.1112	-1.601e-02	-9.364e-03	1.71
3	0.1163	-1.599e-10	-1.596e-10	1.00
4	0.6762	-9.574e-11	-9.573e-11	1.00

TABLE 5.1
Model error estimator and efficiencies for mean value functional j_1.

iter	fraction of exact model	η_m	$j_2(u_h - u_{hm})$	I_{eff}
1	0.	-4.0498e-02	-2.5927e-02	1.56
2	0.0806	-7.6145e-03	-5.1713e-03	1.47
3	0.1180	-7.3481e-04	-4.2331e-04	1.73
4	0.1187	-9.4160e-11	-9.4028e-11	1.00
5	0.4887	-4.5844e-11	-4.5712e-11	1.00

TABLE 5.2
Model error estimator and efficiencies for point functional j_2.

Tables 5.1 and 5.2 display our numerical results. The second column contains the fraction of marked cells. For both functionals, with less than 12% cells using the

exact diffusion model, the solution u_{hm} is equal to u_h up to machine accuracy. The effectivity index

$$I_{eff} = \frac{\eta_m}{j(u_h - u_{hm})}$$

is asymptotically exact. The estimator gives us the true prediction: using the exact model on more cells than in the third iteration for j_1 (and fourth iteration for j_2) does not pay off. We need to use the exact model and the expensive quadrature rule only on a small fraction of all cells. The areas where the exact model is used are shown in Figure 5.2 by the white parts. As expected, they are mainly localized at the cracks.

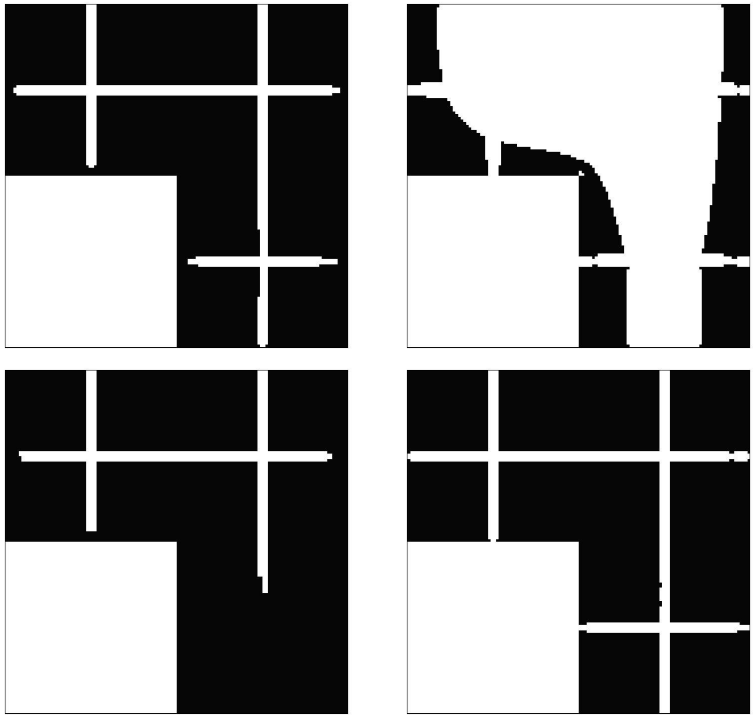

FIG. 5.2. *Areas with exact model $\mu(x)$ (light area) and crude model with constant diffusion (black area). Top line: iteration 2 and 4 of adaptive process for the mean functional j_1. Bottom: iteration 2 and 4 for the point functional j_2.*

5.2. Nonlinear perturbation of the diffusion coefficient. We consider the Poisson problem (5.1) posed on the unit 2D square $\Omega = (0,1)^2$ with homogeneous Neumann condition at $\Gamma_N = (\{0\} \times [0,1)) \cup ([0,1) \times \{0\})$, homogeneous Dirichlet conditions at $\partial\Omega \setminus \Gamma_N$, the right-hand side $f \equiv 10^3$ and with μ depending nonlinearly on u:

$$\mu(x) = 1 + \alpha \|\nabla u(x)\| \,.$$

This type of nonlinear diffusion coefficient arises in turbulence modeling (Smagorinsky model). While the laminar viscosity is equal to 1, the additional nonlinear part is a

turbulent viscosity. The part we want to neglect in a controlled way is exactly this part:

$$d(u)(\phi) \;=\; (\alpha\|\nabla u\|\nabla u, \nabla\phi)\,.$$

During the refinement of the model, we split the cells of the triangulation in those where only the laminar viscosity is used, $\mathcal{T}_{1h}$, and those where the sum of laminar and turbulent viscosity is used. In iteration i we solve the Poisson problem with approximate diffusion coefficient μ_i given by

$$\mu_i(x) = \begin{cases} 1\,, & \text{if } x \in \mathcal{T}_{1h}\,, \\ \mu(x)\,, & \text{otherwise.} \end{cases}$$

The adaptive procedure is started with the laminar viscosity over the whole domain. The output functional we choose is the mean value over a rectangular subdomain $\Omega_0 = [0, 0.5] \times [0.5, 1] \subset \Omega$:

$$j(u) = \int_{\Omega_0} u\,dx\,.$$

We investigate two test cases: one with small turbulent viscosity ($\alpha = 0.001$) and one with large ($\alpha = 0.1$).

#nodes	exact model	η_h	η_m	η	$j(u-u_{mh})$	I_{eff}
81	0.000	2.258e-01	-1.459e-02	2.112e-01	2.118e-01	1.00
289	0.000	5.662e-02	-1.476e-02	4.186e-02	4.187e-02	1.00
1,005	0.084	1.422e-02	-1.294e-02	1.282e-03	1.335e-03	0.96
3,321	0.348	3.522e-03	-8.801e-03	-5.279e-03	-5.253e-03	1.00
11,085	0.703	8.612e-04	-3.808e-03	-2.947e-03	-2.941e-03	1.00
32,703	0.954	2.347e-04	-1.115e-03	-8.807e-04	-8.787e-04	1.00
125,861	0.995	5.897e-05	-5.477e-05	4.206e-06	4.299e-06	0.98
495,031	0.999	1.474e-05	-4.053e-06	1.069e-05	1.069e-05	1.00

TABLE 5.3
Error estimators and efficiencies for nonlinear turbulent viscosity, $\alpha = 0.001$.

#nodes	exact model	η_h	η_m	η	$j(u-u_{mh})$	I_{eff}
81	0.000	2.259e-01	-1.459e-00	-1.233e-00	-1.125e-00	1.10
81	0.609	1.989e-01	-3.675e-01	-1.686e-01	-1.281e-01	1.32
201	0.855	4.862e-02	-4.247e-01	-3.761e-01	-3.330e-01	1.13
745	0.903	-1.506e-03	-3.319e-01	-3.334e-01	-2.851e-01	1.17
2,793	0.997	3.291e-03	-5.794e-04	2.712e-03	2.639e-03	1.03
10,669	0.998	8.187e-04	-1.542e-04	6.645e-04	6.454e-04	1.03
41,375	0.992	-9.095e-04	-3.728e-02	-3.819e-02	-3.317e-02	1.15
162,663	0.999	5.028e-05	-7.708e-05	-2.680e-05	-2.278e-05	1.18

TABLE 5.4
Error estimators and efficiencies for nonlinear turbulent viscosity, $\alpha = 0.1$.

The results for $\alpha = 0.001$ are listed in Table 5.3. Each line corresponds to a step in the adaptive model/mesh refinement procedure. The number of nodes is given in

the first column while the percentage of cells where the accurate model is used is given in the second column. The adaptive algorithm achieves a very good balance of error contributions from the discretization and the model. Although the two parts of the error cancel in part, the estimator is a very good approximation of the true error. This can be seen by looking at the efficiency index

$$I_{eff} = \frac{\eta}{j(u - u_{hm})}$$

in the last column. We also notice that the discretization error is dominant on coarse meshes so that the turbulent viscosity model starts being used only when local mesh refinement has brought the discretization error to levels comparable with the model error.

The results for $\alpha = 0.1$ shown in Table 5.4 confirm that our error estimator yields a very accurate prediction of the overall error. In the case $\alpha = 0.1$ however, even on very coarse meshes, the influence of the turbulent viscosity is larger. This results in a large modeling error compared to the discretization error. The first step in the adaptive model/mesh refinement procedure only refines the model. As a result, most of the cells involve the exact model even on relatively coarse meshes.

Some obtained meshes and the corresponding areas of crude and exact models are shown in Figure 5.3 for the case $\alpha = 0.001$. The black areas indicate the regions where the laminar viscosity is used. The mesh is locally refined in the neighbourhood of Ω_0. Similar results are obtained in the case $\alpha = 0.1$.

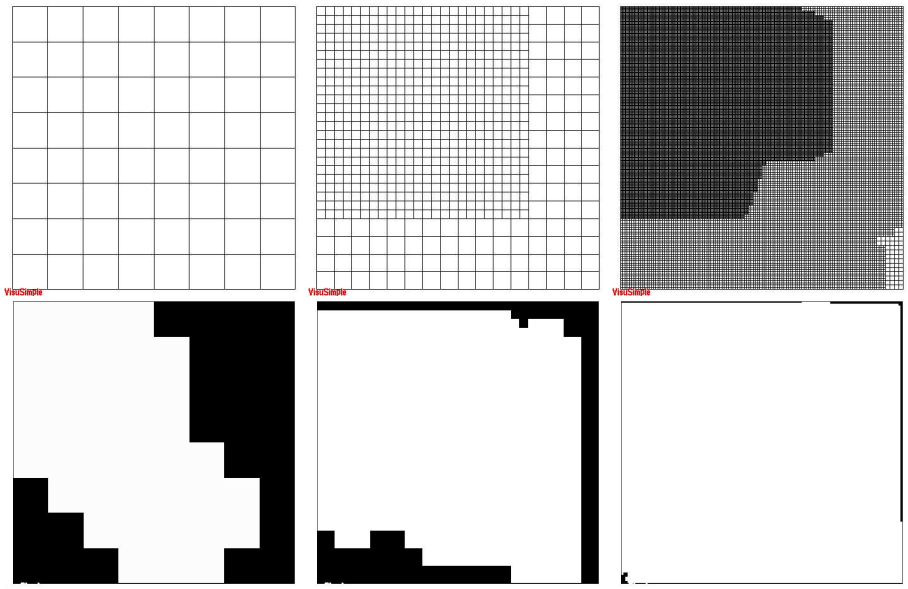

FIG. 5.3. *Obtained meshes (upper part) for $\alpha = 0.1$ with 81, 745 and 41,375 points; and corresponding models (lower part): exact model $\mu(x)$ (light area), laminar viscosity only (black area).*

5.3. System of convection-diffusion-reaction equations. The considered equations describe a reacting mixing layer of three species S_i, $i = 1, 2, 3$, diluted in an inert carrier gas

$$\beta \cdot \nabla u - \operatorname{div}(D \nabla u) = f(u)\,,$$

where $u = (u_1, u_2, u_3)^T$ are their concentrations, and $D = (D_{ij})$ is the flux diffusion matrix taken in the form

$$D = \lambda \begin{pmatrix} d_1 & 0 & 0 \\ 0 & d_2 & 0 \\ 0 & 0 & d_3 \end{pmatrix} + \lambda \begin{pmatrix} d_{11}u_1 & d_{12}u_1 & d_{13}u_1 \\ d_{21}u_2 & d_{22}u_2 & d_{23}u_2 \\ d_{31}u_3 & d_{32}u_3 & d_{33}u_3 \end{pmatrix},$$

with $\lambda = (2 - u_1 - u_2 - u_3)^{-1}$ and fixed parameters $d_{ij} = \sqrt{d_i d_j}$, $d_1 = 10^{-4}$ and $d_2 = d_3 = 10^{-2}$. Note that $D' = (D'_{ij})$ with $D'_{ij} = D_{ij}/u_i$ is the diffusion matrix and is symmetric positive definite. The advection is constant and divergence free, $\beta = (1, 0)^T$. The notation $\beta \cdot \nabla u$ stands for the vector with components $\beta \cdot \nabla u_i$, $i = 1, 2, 3$. The reaction term reads $f(u) = (0, -10^3, 10^3)^T u_1 u_2$ and describes the reaction

$$S_1 + S_2 \to S_1 + S_3 \,.$$

The species S_2 is consumed, S_3 is produced and S_1 is a so-called third-body. The computational domain is $\Omega = [0, 1] \times [0, 0.2]$. The boundary conditions are homogeneous Neumann for all components on all parts of $\partial\Omega$, except for $x = 0$, where Dirichlet conditions are imposed:

$$u(0, y) = \begin{cases} (0.1, 0, 0)^T & \text{if } y \geq 0.5 \,, \\ (0, 0.1, 0)^T & \text{if } y < 0.5 \,. \end{cases}$$

This model problem describes the mixing of two diluted species S_1 and S_2 with equal molecular weight and with a realistic diffusion matrix. The diffusion coefficients d_i represent the diffusion of species i in the mixture. The non-diagonal diffusion of this model introduces a coupling between gradients of the species. In practice, this yields additional couplings in the Jacobian matrix. For large reaction systems with many chemical species involved, this may become prohibitive. Moreover, multicomponent diffusion matrices are usually given only implicitly from the solution of a constrained singular system and their computation is numerically expensive, see [6, 8]. Therefore, instead of using the flux diffusion matrix D, we prefer to use a simpler model consisting of a diagonal flux diffusion matrix

$$\widetilde{D} = \lambda \cdot \mathrm{diag}(d_i) \,.$$

This matrix is slightly nonlinear due to the $u-$dependence of λ. The corresponding semi-linear forms read for test functions ϕ:

$$\begin{aligned} a(u)(\phi) &= (\beta \cdot \nabla u, \phi) + (\widetilde{D}\nabla u, \nabla\phi) - (f(u), \phi) \,, \\ d(u)(\phi) &= ((D - \widetilde{D})\nabla u, \nabla\phi) \,. \end{aligned}$$

The quantity of interest is chosen as the mean value of the product

$$j(u) = \int_\Omega u_3 dx \,.$$

The species concentrations u_i are shown in Figure 5.4. The third-body S_1 diffuses very slowly, S_2 diffuses faster and is consumed, S_3 is produced in the reaction layer and diffuses fast.

Table 5.5 presents the discretization and model errors. Because on coarse meshes the discretization error dominates, the mesh is refined in the mixing layer keeping the

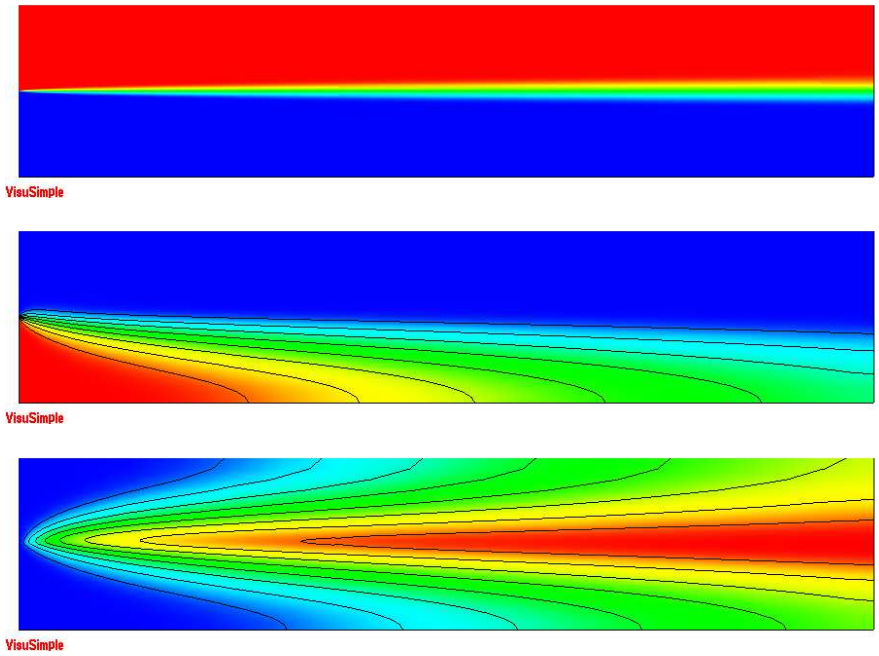

FIG. 5.4. *Solution of the reacting mixing layer problem, species S_1 (top), S_2 (middle), S_3 (bottom).*

#cells	exact model	η_h	η_m	η	$j(u-u_{mh})$	I_{eff}
289	0	9.470e-04	-5.199e-05	8.950e-04	6.121e-04	1.46
537	0	3.479e-04	-5.756e-05	2.904e-04	2.591e-04	1.12
1001	0	1.452e-04	-5.942e-05	8.584e-05	1.029e-04	0.83
1425	0.344	6.940e-05	-1.369e-05	5.570e-05	7.741e-05	0.72
2863	0.303	3.269e-05	-1.585e-05	1.683e-05	2.772e-05	0.61
3615	0.369	1.690e-05	-1.455e-06	1.545e-05	2.231e-05	0.69
7855	0.361	8.087e-06	-2.677e-06	5.409e-06	5.047e-06	1.07
9217	0.434	4.609e-06	3.889e-06	8.499e-06	4.905e-06	1.73

TABLE 5.5
Error estimators and efficiencies for simplified diffusion matrix

crude model in the whole domain. When both types of error are balanced, the model is adapted locally to the full flux diffusion matrix on up to 43.4% of the cells. The error $j(u-u_{mh})$ is still well represented by the estimator η.

Some meshes and the areas where both types of flux diffusion matrices are used are presented in Figure 5.5. The detailed model is only used in the lower part of Ω where gradients in u_2 arise (light area).

6. Conclusions and perspectives. We derived an estimator for measuring simultaneously two types of errors, modeling and discretization errors, with respect to nearly arbitrary output functionals. The approach is formulated for stationary nonlinear partial differential equations involving complex models. By localization of the estimator, we derived local error indicators which allow for local mesh refinement

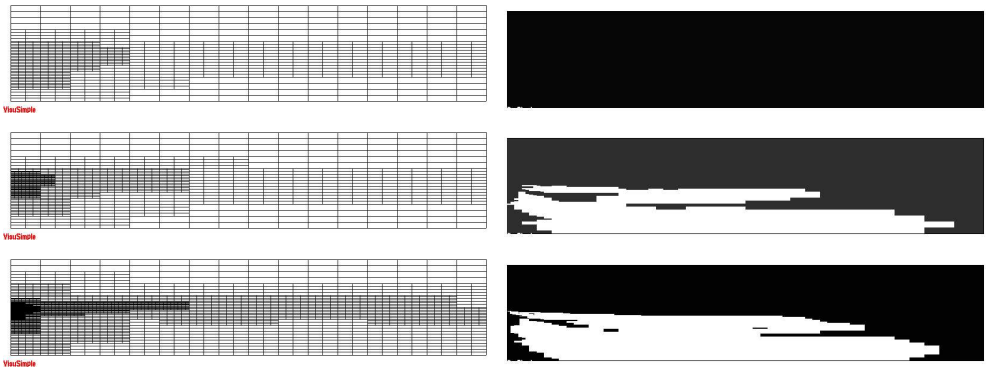

FIG. 5.5. *Meshes (left) and corresponding model zones (right) for the reacting mixing-layer problem. Light areas indicate regions where the detailed model is used.*

and local model modification. The extra costs for measuring the model error are basically the evaluation of residuals with respect to the detailed model.

We analyzed three numerical examples involving linear and nonlinear problems, scalar equations and systems, and where always two types of diffusion models entered: a cheap but inaccurate and a more complex diffusion model. In all cases, the estimator is reliable and efficient. By local modification of the model and the mesh size, both sources of errors are balanced. In future work, the present framework will be the basis for handling large-scale complex models where substantial CPU time savings are expected. A typical setting is the application to Soret- and Dufour effects which are important on small scales (e.g., in flame fronts).

REFERENCES

[1] R. L. ACTIS, B. A. SZABO, AND C. SCHWAB, *Hierarchic models for laminated plates and shells*, Comput. Methods Appl. Mech. Engrg., 172 (1999), pp. 79–107.

[2] R. BECKER AND R. RANNACHER, *An optimal control approach to a-posteriori error estimation in finite element methods*, in Acta Numerica 2001, A. Iserles, ed., Cambridge University Press, 2001.

[3] M. BRAACK, *An Adaptive Finite Element Method for Reactive Flow Problems*, Dissertation, Universität Heidelberg, 1998.

[4] M. BRAACK, R. BECKER, AND R. RANNACHER, *The dual-weighted residual method applied to three-dimensional flow problems.* submitted to AMIF 2002 Proc., Computer & Fluids, 2002.

[5] M. BRAACK AND R. RANNACHER, *Adaptive finite element methods for low-Mach-number flows with chemical reactions*, in 30th Computational Fluid Dynamics, H. Deconinck, ed., vol. 1999-03 of Lecture Series, von Karman Institute for Fluid Dynamics, 1999, pp. 1–93.

[6] A. ERN AND V. GIOVANGIGLI, *Multicomponent Transport Algorithms*, Lecture Notes in Physics, m24, Springer, 1994.

[7] L. FATONE, P. GERVASIO, AND A. QUARTERONI, *Multimodels for incompressible flows: iterative solutions for the navier-stokes/oseen coupling*, M2AN, Math. Model. Numer. Anal., 35 (2001), pp. 549–574.

[8] V. GIOVANGIGLI, *Multicomponent Flow Modeling*, Birkhäuser, Boston, 1999.

[9] J. T. ODEN AND K. VEMAGANTI, *Estimation of local modeling error and goal-oriented modeling of hetereogeneous materials; Part I: error estimates and adaptive algorithms*, J. Comp. Phys., 164 (2000).

[10] ———, *Estimation of local modeling error and goal-oriented modeling of hetereogeneous materials; Part II: A computational environment for adaptive modeling of heteregeneous elastic*

solids, Comput. Methods Appl. Mech. Engrg., 190 (2001).
[11] S. Stein, E.; Ohnimus, *Anisotropic discretization- and model-error estimation in solid mechanics by local neumann problems*, Comput. Methods Appl. Mech. Engrg., 176 (1999), pp. 363–385.