Int. Conference on Boundary and Interior Layers
BAIL 2006
G. Lube, G. Rapin (Eds)


# Anisotropic mesh adaption based on a posteriori estimates and optimisation of node positions

René Schneider[1,a] and Peter K. Jimack[2]

[1] TU Chemnitz, Fak. f. Mathematik, D-09107 Chemnitz, Germany.
[a] rene.schneider@mathematik.tu-chemnitz.de
[2] School of Computing, University of Leeds, Leeds, LS2 9JT, UK.

## 1. Introduction

Boundary or interior layers are usually highly directional solution features. Thus, suitable anisotropic meshes, reflecting the directional features of the solution, provide the basis for the most efficient numerical approximation. Anisotropic mesh design strategies based upon *a priori* analysis have been developed for a variety of PDE problems and discretisations. On the other hand *a posteriori* error estimation techniques have been developed and integrated with mesh refinement strategies, leading to numerical methods which perform extremely well over broad classes of problems, even when no *a priori* analysis is available. One particular advantage of the *a posteriori* approach is that it can yield meshes that efficiently approximate specific functionals of the solution [2]. Most of the common *a posteriori* based algorithms are unable to introduce suitable anisotropy into the mesh however.

We introduce here a new mesh adaptation strategy which allows suitable anisotropy within the mesh. The approach draws upon methods from numerical optimisation in order to modify the node positions of a given (isotropic) mesh such that an *a posteriori* error estimate is minimised. To make this feasible for non-trivial problems the *discrete adjoint* technique [3] is utilised to efficiently evaluate the gradient of the *a posteriori* error estimate. The *Dual Weighed Residual (DWR)* error estimate [2] for the error in a quantity of interest is utilised to allow goal driven adaptivity. This present paper is based upon our previous work [5] to which refer for a more detailed consideration of the approach and for an overview of related previous work.

## 2. The discrete adjoint technique

Consider a scalar-valued function, $I$, of an independent vector variable $s$, such that

$$I(s) := \widetilde{I}(u(s), s), \tag{1}$$

where the vector $u(s)$ is defined implicitly by the (possibly nonlinear) system

$$0 = R(u(s), s). \tag{2}$$

Consider the effect of small perturbations $\delta s$ of $s$ in (1) and (2). Discarding higher order derivative terms, such a perturbation results in perturbations $\delta I$ in $I$ and $\delta R$ in $R$,

$$\delta I = \frac{\partial \widetilde{I}}{\partial u}\delta u + \frac{\partial \widetilde{I}}{\partial s}\delta s, \tag{3}$$

$$0 = \delta R = \frac{\partial R}{\partial u}\delta u + \frac{\partial R}{\partial s}\delta s, \tag{4}$$

where $\delta R = 0$ since (2) must be satisfied. As $\delta R = 0$, we can multiply it by an arbitrary term $\Psi^T$ and subtract it from the right-hand side of (3), giving

$$\delta I = \left(\frac{\partial \widetilde{I}}{\partial u} - \Psi^T \frac{\partial R}{\partial u}\right)\delta u + \left(\frac{\partial \widetilde{I}}{\partial s} - \Psi^T \frac{\partial R}{\partial s}\right)\delta s. \tag{5}$$

This implies that $\delta I$ may be evaluated without calculating $\delta u$ provided

$$\left[\frac{\partial R}{\partial u}\right]^T \Psi = \left[\frac{\partial \widetilde{I}}{\partial u}\right]^T, \tag{6}$$

and if $u(s)$ is well-defined by (2) then Equation (6) uniquely defines $\Psi$ which is of the same dimension as $u$. Equation (6) is known as the adjoint equation and $\Psi$ as the adjoint solution. With this choice of $\Psi$ the total derivative $\mathrm{D}I/\mathrm{D}s$ can be written as

$$\frac{\mathrm{D}I}{\mathrm{D}s} = \frac{\partial \widetilde{I}}{\partial s} - \Psi^T \frac{\partial R}{\partial s}. \tag{7}$$

The importance of this representation is that, once the original equation (2) is solved and $I(s)$ evaluated from (1), $\mathrm{D}I/\mathrm{D}s$ may be evaluated for little more than the cost of a single solve of the linear system (6) and a single matrix-vector product in (7), regardless of the dimension of $s$. This is compared to other methods of evaluating $\mathrm{D}I/\mathrm{D}s$ which typically require the solution of (2) (or a linearised version) per component of $s$.

## 3. The Dual Weighed Residual method

The Dual Weighed Residual (DWR) method for estimating the error in a functional, that depends upon the computed finite element (FE) solution, is introduced here for the case of a linear functional and a linear PDE. Further details may be found in [2] and the references therein. Consider a PDE in weak form (8) with unique solution $u \in \mathbb{H}^1_{g_D,\Gamma_D}$ and the functional $J(u)$ as defined by (9)

$$a(u,v) = b(v) \qquad \forall v \in \mathbb{H}^1_{0,\Gamma_D}, \tag{8}$$

$$J(u) := \int_\Omega gu \, \mathrm{d}\Omega, \tag{9}$$

where $g$ is a kernel function. Here $\mathbb{H}^1_{f,\Gamma_D}$ denotes $\left\{u \in \mathbb{H}^1(\Omega) : \quad u = f \text{ on } \Gamma_D\right\}$, where $\Gamma_D$ is the (non-empty) Dirichlet part of the boundary of the domain $\Omega$. Let $u_h$ be defined as the solution of the FE discretisation of the weak form (8), hence $u_h \in \mathbb{V}_{g_D,h}$ such that

$$a(u_h, v_h) = b(v_h) \qquad \forall v_h \in \mathbb{V}_{0,h}, \tag{10}$$

where $\mathbb{V}_{f,h}$ denotes the FE function space $\mathbb{V}_{f,h} \subset \mathbb{H}^1_{f,\Gamma_D}$. Utilising the solution $z$ of the dual problem: find $z \in \mathbb{H}^1_{0,\Gamma_D}$ for which

$$a(\varphi, z) = J(\varphi) \qquad \forall \varphi \in \mathbb{H}^1_{0,\Gamma_D}, \tag{11}$$

one can easily derive the representation of the error in the approximation of functional $J$,

$$J(e) := J(u) - J(u_h) = b(z) - a(u_h, z). \tag{12}$$

In order to define a computable error estimate the dual solution $z$ has to be approximated by a computable quantity. Note that the use of any approximation $z_h \in \mathbb{V}_{0,h}$ for $z$ in (12) will not provide a useful approximation due to Galerkin orthogonality. Thus an approximation from a richer function space than $\mathbb{V}_{0,h}$ has to be used. For simplicity and reliability here we utilise an approximation $z_{\mathrm{app}}$ derived by solving the dual FE problem on a uniformly refined mesh ($z_{\mathrm{app}} \in \mathbb{V}_{0,h/2}$). The resulting error estimate is defined as

$$J_{\mathrm{est}} := b(z_{\mathrm{app}}) - a(u_h, z_{\mathrm{app}}). \tag{13}$$

Error estimates such as the DWR estimate are commonly used to guide local mesh refinement in regions which contribute most to the estimated error. This may be achieved, for example, by evaluating the right-hand side of (13) separately on each element and then refining those elements with the largest contributions. By *isotropic local mesh refinement* we denote the common approach of successively marking the unmarked element with the largest absolute error contribution until the sum of error contributions of the marked elements exceeds a given threshold ($\gamma = 0.3$, say) of the total error, and then subdividing each marked element into four elements of equal size. For a discussion of the treatment of hanging nodes, which come about when an element is refined but a neighbouring element is not, we refer to [5].

## 4. The proposed approach

The approach discussed in this section may be summarised as seeking to move the nodes of an existing mesh in order to reduce the estimated error (and therefore, one hopes, the actual error) in the quantity of interest, while maintaining the same mesh connectivity. Of course the use of node movement alone can only redistribute a constant number of degrees of freedom and so for most practical problems it will be necessary to combine this with isotropic local mesh refinement in some way. Such a hybrid approach of node movement and isotropic local refinement will not generally improve the asymptotic convergence properties of the underlying FE discretisation in terms of $h$, however realistic goals are to yield a better constant for this asymptotic behaviour and/or to reach the asymptotic regime more quickly (i.e. with fewer degrees of freedom). The remainder of this section describes a simple adaptive refinement algorithm comprising of two stages. First, a relatively coarse initial mesh has the positions of its nodes improved by applying a number of steps of an optimisation algorithm. The resulting mesh is then assumed to be a good staring point for adaptive isotropic refinement, the second stage of the algorithm. This is applied repeatedly until the error estimate indicates a satisfactory accuracy. Figure 1 gives a schematic overview of this approach.

As the node relocation by the optimisation is only performed on a relatively coarse mesh, the computational cost of this step is small compared to the potential benefit. However, the performance of this approach is clearly limited by the reliability of the approximation on the coarse mesh which could lead to a sub-optimal distribution of the node points at the initial stage, which would be impossible recover from in the later stages. Also the maximum aspect ratio is determined by the connectivity of the initial mesh and the constraints that are imposed. An alternative approach is also considered in [5], circumventing some of these difficulties, albeit at higher computational cost.

For a mesh of fixed size the smallest overall error is achieved by equidistribution of element-wise error indicators (e.g. [2, Section 4.2]). We attempt this equidistribution by solving a least squares optimisation problem. For this purpose we define the cost functional as

$$\eta^2 := \sum_{T \in \mathcal{T}} |J_{est,T}|^2, \tag{14}$$

where $J_{est,T}$ represents the right hand side of (13) evaluated on element $T$, of triangulation $\mathcal{T}$, only. Essential for the optimisation approach taken in this work is that this choice of performance function is differentiable with respect to the node positions in the mesh and that these derivatives are easily computable. Naturally one must also impose certain constraints on the node movement in order to ensure that the mesh remains suitable for finite element calculations. However, restrictions are also necessary to ensure that the error estimate itself remains a reliable approximation of the error. These constraints are summarised in the following statement of the optimisation problem.

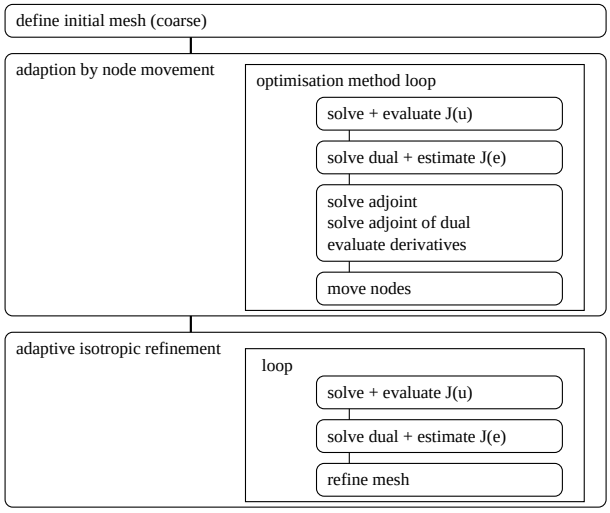


Figure 1: Combined adaption strategy `opt-adapt`

**Problem 1:**

Minimise $\eta^2(s)$, with respect to the node positions $s$, subject to:

1. the mesh approximates the boundaries of the domain,
2. the mesh is non-self-overlapping (NSO),
3. interior angles of the triangles staying bounded well bellow $\pi$, and
4. the aspect ratio of triangles varies smoothly (i.e. changes in the aspect ratio of neighbouring elements must be bounded).

Constraints 1, 2 and 3 are standard geometric restrictions. Note that bounding the angles from below is not appropriate since this prevents the possibility of strongly anisotropic meshes. Condition 4 is introduced as a safeguard to ensure the reliability of the DWR error estimate. The realisation of these constraints and further details are discussed in [5]. It should be noted that Problem 1 is highly nonlinear and its solution is extremely challenging for all but the most trivial finite element meshes. For a practical adaptive algorithm it is therefore essential to limit the computational effort that goes into attempting to find a solution, if one even exists.

Problem 1 comprises a nonlinear optimisation problem with nonlinear inequality constraints. For this class of problem sequential quadratic programming (SQP, e.g [4]) is established as a reliable and efficient technique. Important for the efficient implementation of this optimisation algorithm in our context is that the required quadratic models of the performance function can be obtained by combining the gradients at a sequence of iterates of the SQP algorithm using the BFGS update formula. Thus, only the performance function $\eta^2$ and its gradient with respect to the node positions are required for the computations. The discrete adjoint technique, outlined in Subsection 2., is used to obtain this gradient in an efficient manner. The NSO constraint is implemented by a trust region approach which restricts the search space at each step of the SQP algorithm to a class of node movements that conform with the NSO constraint. Again, we refer to [5] for details.

## 5. Example problems

In order to assess the anisotropic refinement approach proposed in this paper a model PDE problem is considered. This PDE is selected because exact analytical solutions are known in certain cases whilst *a priori* information regarding the solution behaviour is available in other

cases. This allows the quality of the algorithm to be contrasted against well established *a priori* results. It is envisaged that the *a posteriori* approach used here will be much more generally applicable than the *a priori* theory however.

Consider the following reaction-diffusion equation:

$$\left.\begin{array}{rrcll} -\Delta u + \frac{1}{\varepsilon^2} u & = & \frac{1}{\varepsilon^2} & \text{in} & \Omega \\ \text{subject to} \qquad u & = & 0 & \text{on} & \Gamma_D \\ \frac{\partial u}{\partial n} & = & g_N & \text{on} & \Gamma_N, \end{array}\right\} \tag{15}$$

where $\varepsilon > 0$, $\Gamma_D \cup \Gamma_N$ is the boundary $\partial\Omega$ of the domain $\Omega$, $|\Gamma_D \cap \Gamma_N| = 0$ and $|\Gamma_D| > 0$. As quantity of interest consider the integral of the normal derivative of $u$ over the Dirichlet boundary,

$$J(u) := \int_{\Gamma_D} \frac{\partial u}{\partial n} \, \mathrm{d}\Gamma. \tag{16}$$

Numerous alternative choices could have been made but this is selected partly for its simplicity and partly for its similarity to the viscous drag terms that are frequently of interest in fluid dynamics problems. For the purposes of this numerical assessment two example cases are selected with different choices of $\Omega$, $\Gamma_D$, $\Gamma_N$ and $g_N$.

**Example 1**

***Domain:*** *Unit square,* $\Omega = (0,1)^2$.

***Boundary conditions:***

$$\begin{array}{rcll} u = & 0 & \forall (x,y) : x = 1 \\ \frac{\partial u}{\partial n} = & 0 & \forall (x,y) : y = 1 \vee y = 0 \\ \frac{\partial u}{\partial n} = & \frac{1}{\varepsilon} e^{-1/\varepsilon} & \forall (x,y) : x = 0 \end{array}$$

This problem has the exact solution

$$u(x,y) = 1 - e^{(x-1)/\varepsilon}$$

which, when $\varepsilon$ is small, involves a steep boundary layer next to the $x = 1$ boundary. Furthermore, using (16), it is easy to show that $J(u) = -1/\varepsilon$ for this example.

**Example 2**

***Domain:*** *Square with a square hole,*

$$\Omega = (-1,1)^2 \setminus (-\frac{1}{5}, \frac{1}{5})^2.$$

***Boundary conditions:***

$$\begin{array}{rcl} u = & 0 & \forall (x,y) \in \left[-\frac{1}{5}, \frac{1}{5}\right]^2 \\ \frac{\partial u}{\partial n} = & 0 & \forall (x,y) : (x = \pm 1) \vee (y = \pm 1) \end{array}$$

This is a problem for which an exact solution is not known, however when $\varepsilon$ is small the solution

is one throughout almost all of the domain, except near to the inner boundary (the hole at $[-\frac{1}{5}, \frac{1}{5}]^2$) where it changes rapidly to zero.

## 6. Adjoint equations for derivatives of the error estimate $\eta^2$

This section provides an outline of how the discrete adjoint technique may be applied to evaluate the derivative of $\eta^2$ with respect to the node positions. Here $\eta^2$ takes the role of $\widetilde{I}$ in (1) and it is considered to be a function of $\underline{u}$, the coefficient vector of the primal finite element solution, $\underline{z}$, the coefficient vector of the finite element solution of the dual problem, and the vector of node coordinates $s$ for the underlying FE mesh. The quantities $\underline{u}$ and $\underline{z}$ are themselves dependent upon $s$. To complete the notation of Section 2., therefore let

$$u := \left[ \underline{u}^T, \quad \underline{z}^T \right]^T \tag{17}$$

$$\text{and} \qquad R(u,s) := \begin{bmatrix} K(s) & \\ & K_{\text{dual}}(s) \end{bmatrix} u - \begin{bmatrix} \underline{b}(s) \\ \underline{b}_{\text{dual}}(s) \end{bmatrix}, \tag{18}$$

where $K(s)$ and $\underline{b}(s)$ are the stiffness matrix and right-hand side from the FE discretisation of the primal problem and $K_{\text{dual}}(s)$ and $\underline{b}_{\text{dual}}(s)$ are those of the dual problem utilised in the error estimate. The adjoint equation (6) therefore becomes

$$\begin{bmatrix} K^T & \\ & K_{\text{dual}}^T \end{bmatrix} \Psi = \frac{\partial \eta^2}{\partial u} \tag{19}$$

and the total derivatives $\mathrm{D}\eta^2/\mathrm{D}s$ can be evaluated according to (7) as

$$\frac{\mathrm{D}\eta^2}{\mathrm{D}s} = \frac{\partial \eta^2}{\partial s} - \Psi^T \begin{bmatrix} \frac{\partial (K\underline{u} - \underline{b})}{\partial s} \\ \frac{\partial (K_{\text{dual}}\underline{z} - \underline{b}_{\text{dual}})}{\partial s} \end{bmatrix}. \tag{20}$$

Explicit formulae can be given for the partial derivatives with respect to node positions (the $\partial/\partial s$ terms), see [5]. Any implementation of the expression (20) may have its correctness verified by comparing the values of $\mathrm{D}\eta^2/\mathrm{D}s$ to those computed, at much greater expense, through the use of finite differences.

## 7. Results

Figure 2 illustrates the performance of the proposed refinement algorithm for the two example problems considered. The results are contrasted with the performance of global and local isotropic mesh refinement, and with Shishkin meshes that are based upon *a priori* analysis (e.g. [1]). In this paper only the case $\varepsilon = 10^{-3}$ is considered, and the relative error $|J_{\text{est}}|/|J(u)|$ is plotted against the number of nodes in the primal mesh. In the first example, for which the exact solution is known, $|J(e)|/|J(u)|$ is also plotted.

One of the most significant observations concerning these tests is that the estimated error and the actual error follow each other closely. When $\varepsilon$ is large (e.g. $\varepsilon = 10^{-1}$), then all of the methods perform in a similar, optimal, manner (the solution being essentially isotropic, results have been omitted to keep the presentation short). However, in the anisotropic cases shown in Figure 2 the proposed method provides a significant advantage over the adaptive isotropic refinement approaches (and this advantage becomes more pronounced as $\varepsilon$ gets smaller). Indeed, the desired goal of reaching the asymptotic regime more quickly (with fewer degrees of freedom) and with a better constant is achieved. Tests with even smaller $\varepsilon$ have shown that the proposed *a posteriori* approach may not always perform as well as the use of Shishkin meshes. However, this is unsurprising since the Shishkin approach requires *a priori* knowledge, which will not be available for more general classes of problems. The results of the computational experiments

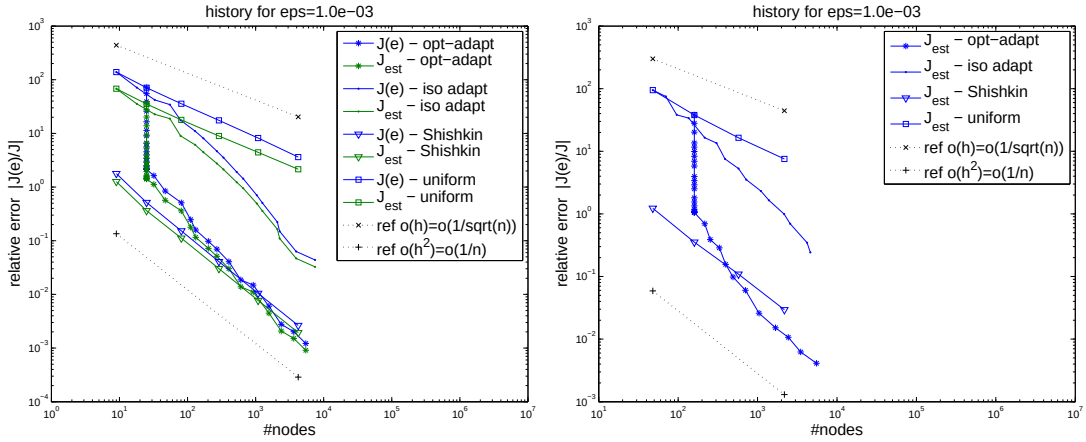


Figure 2: Convergence histories ($\varepsilon = 10^{-3}$) for Example 1 (left) and Example 2 (right)

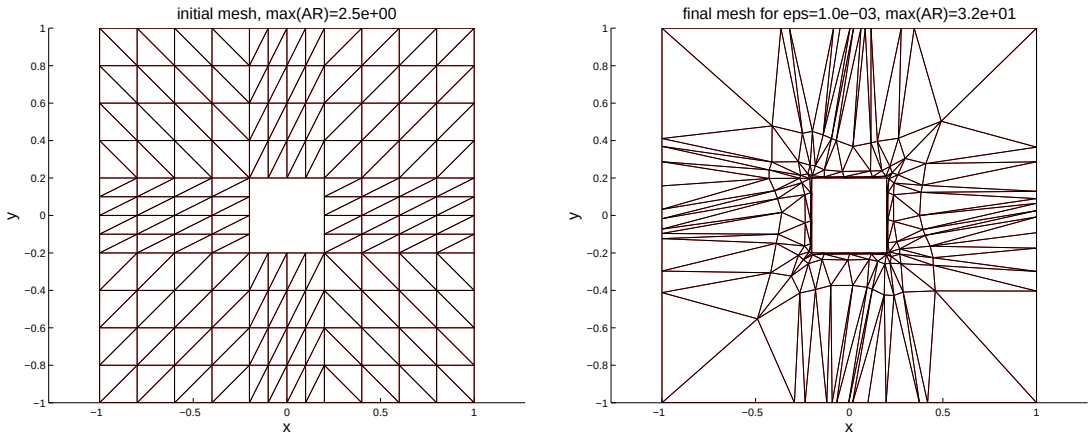


Figure 3: Initial and optimised coarse mesh ($\varepsilon = 10^{-3}$) for Example 2

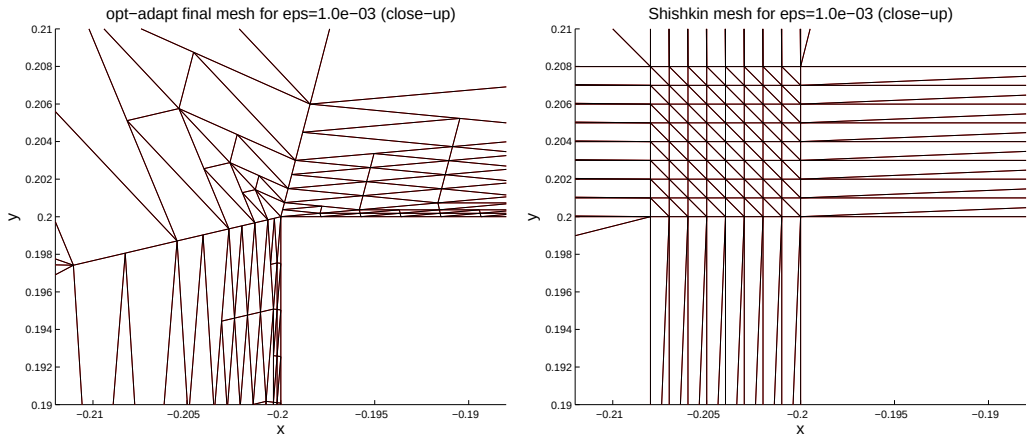


Figure 4: Closeups of final meshes for Example 2, $\varepsilon = 10^{-3}$, new algorithm (left) and Shishkin (right)

indicate that the fully automated *a posteriori* approach can deliver anisotropic meshes that are a significant step towards this optimal behaviour. To conclude this section Figures 3 and 4 show initial and final meshes obtained for Example 2. Note that the final meshes and solutions do not appear to be particularly sensitive to the precise choice of parameters such as the maximum interior angle allowed.

## 8. Conclusions and Further Work

The main contribution of this work is to apply the discrete adjoint technique to calculate the sensitivity of an *a posteriori* error estimate to the positions of the node points in the underlying finite element mesh. This allows the *a posteriori* error estimate to be used not only for isotropic local refinement but also as a means of relocating the nodes in the mesh. An adaptive algorithm has been presented which combines two forms of adaptivity (refinement and movement), making use of nonlinear optimisation techniques including SQP and the trust-region concept. The main conclusion of this work is that it is indeed feasible to both efficiently calculate and make use of the sensitivity of the error estimate within an automatic adaptive algorithm. Furthermore, it has been demonstrated that this approach can deliver optimal convergence properties, reaching the asymptotic regime with significantly fewer unknowns than is possible with standard isotropic local refinement alone.

Having demonstrated the feasibility of this approach here it is now necessary to extend the work to larger classes of PDE and a wider range of *a posteriori* error estimates. It has already been demonstrated in [5] that a key factor is to ensure the reliability of the error estimate so as to ensure that as the mesh becomes more anisotropic this does not deteriorate and lead to optimisation of a meaningless quantity. Other problem classes that may be considered include convection-diffusion problems, whose solutions may involve very steep layers, or the use of more complex geometries or richer finite element spaces. In principle however all of the techniques developed here can be extended to these situations.

## References


[1] T. Apel and G. Lube. Anisotropic mesh refinement for a singularly perturbed reaction diffusion model problem. *Applied Numerical Mathematics*, 26:415–433, 1998.

[2] W. Bangerth and R. Rannacher. *Adaptive Finite Element Methods for Differential Equations*. Birkhäuser Verlag, 2003.

[3] M.B. Giles and N.A. Pierce. An introduction to the adjoint approach to design. *Flow Turbulence and Combustion*, 65(3–4):393–415, 2000.

[4] J. Nocedal and S.J. Wright. *Numerical Optimization*. Springer Series in Operations Research. Springer, 1999.

[5] R. Schneider and P.K. Jimack. Toward anisotropic mesh adaption based upon sensitivity of a posteriori estimates. School of Computing Research Report Series 2005.03, University of Leeds, 2005. Available at `http://www.comp.leeds.ac.uk/research/pubs/reports/2005/2005_03.pdf`.