European Congress on Computational Methods in Applied Sciences and Engineering
ECCOMAS Computational Fluid Dynamics Conference 2001
Swansea, Wales, UK, 4-7 September 2001

MESH QUALITY AND ANISOTROPIC ADAPTIVITY FOR FINITE ELEMENT SOLUTIONS OF 3-D CONVECTION-DOMINATED PROBLEMS

Mark Walkley, Peter K. Jimack, and Martin Berzins

Computational PDEs Unit, School of Computing, University of Leeds, LS2 9JT, UK,
e-mail: martin@comp.leeds.ac.uk, web page: http://www.comp.leeds.ac.uk/cpde/

Key words: Convection-Dominated Problems, Stable Finite Element Methods, Unstructured Tetrahedral Meshes, Mesh Quality, Anisotropic Mesh Refinement.

Abstract. *In this work we investigate the use of anisotropic mesh refinement algorithms for the adaptive finite element solution of three-dimensional convection-dominated flow problems. The first task is to consider the question of how to quantify whether or not a given mesh is appropriate for representing a particular solution, which may be highly anisotropic. Secondly, we consider how such measures of quality can be used to anisotropically adapt a mesh to produce one that is more appropriate: thus increasing the accuracy of a series of computed solutions at a faster rate than through the use of more conventional techniques such as local h-refinement alone. Throughout the paper we restrict our attention to meshes of tetrahedra and use piecewise linear SUPG finite elements.*

Early work on mesh quality considered grids purely in terms of the geometry of their elements (e.g. [2]) however such an approach is too restrictive for the class of problem being considered here. We allow meshes to breach the typical angle conditions that are appropriate for isotropic solutions and consider alternative measures of mesh quality which are solution dependent. Such measures either explicitly quantify how well the mesh and solution are aligned (e.g. using the Hessian of the computed solution, [11], or by defining a matching function, [9]) or else this is implicit, as in the case of interpolation error estimates (e.g. [3]) or conventional a posteriori *error estimates (e.g. [17]) for example.*

The approach adopted here is to make use of a mesh alignment strategy based upon the convection direction (as in [7]) plus the simple interpolation error estimates of [3] in order to generate anisotropic meshes through a combination of h-refinement and node movement. This is then contrasted with techniques reported elsewhere (e.g. [11] or [17]).

1 INTRODUCTION

It is well-known that convection-dominated flow problems frequently yield solutions which contain regions of relatively fine-scale activity such as sharp layers or shocks. A feature of the solution in such regions is that it is generally highly anisotropic in that the fine-scale variations only occur in some directions and not others. For example, in the region of a planar shock components of the solution change very rapidly in the direction perpendicular to the shock but may be very smooth in the directions parallel to the shock. It is not surprising therefore that conventional isotropic mesh refinement algorithms, such as standard h-refinement (e.g. [15]), tend to result in an unnecessary level of resolution parallel to this type of flow feature in order to deliver the required resolution across the feature. This in turn leads to a computational grid that, in some qualitative sense, is not optimal and hence the efficiency of the entire solution process is likely to be adversely affected.

Unfortunately the problems associated with assessing quantitatively how well a given mesh is suited to the solution of a particular problem is far from straightforward. As outlined in [5], early work on mesh quality was based upon purely geometric considerations such as the minimal angle condition for triangles of [19], or the maximum angle condition of [2] (later extended to tetrahedra in [8]). However, as we have seen, strong local variations in solution component values make it difficult to assess the quality of a mesh without somehow incorporating solution behaviour.

One way of incorporating solution information into the assessment of mesh quality is to make use of *a posteriori* error estimates such as described in [6, 16] for example. Such estimates allow a the approximate error to be computed as a function and from this it may be feasible to extract directional information about the current mesh (as explored in [1] for example). This is not as trouble-free as it may at first seem however since the theoretical error analysis upon which these, and similar, estimates are based typically assume a bound on the aspect ratio of the elements in the mesh. This clearly cannot be relied upon in the cases where we are seeking to produce grids which accurately and efficiently represent highly directional solutions. Recently therefore, Kunert [9] has considered some of the issues associate with *a posteriori* error estimation on anisotropic triangular or tetrahedral grids. For certain problems he is able to prove error bounds which are independent of the element aspect ratios through the use of a *matching function* which describes how well an anisotropic finite element mesh is aligned with the true solution. When this alignment is good the matching function is close to unity and the error estimates are reliable but when the alignment is poor the matching function is large and the error estimates are poor. It may well be possible to approximate this matching function using a computed solution in order to estimate mesh quality but, as yet, no published evidence of the reliability of such a strategy is known to the authors.

These difficulties with rigorous *a posteriori* approaches have led many authors to consider less rigorous measures of mesh quality based more directly upon features of the solution. A popular quantity to estimate is the Hessian of one or more components of the solution, u say. In three dimensions this is given by

$$\mathcal{H} = \begin{pmatrix} \frac{\partial^2 u}{\partial x_1^2} & \frac{\partial^2 u}{\partial x_1 \partial x_2} & \frac{\partial^2 u}{\partial x_1 \partial x_3} \\ \frac{\partial^2 u}{\partial x_2 \partial x_1} & \frac{\partial^2 u}{\partial x_2^2} & \frac{\partial^2 u}{\partial x_2 \partial x_3} \\ \frac{\partial^2 u}{\partial x_3 \partial x_1} & \frac{\partial^2 u}{\partial x_3 \partial x_2} & \frac{\partial^2 u}{\partial x_3^2} \end{pmatrix} . \tag{1}$$

When u is approximated by a piecewise linear finite element function, u^h say, $\mathcal{H}$ must also be approximated. In [11], for example, this is achieved by first obtaining a Galerkin approximation to $\frac{\partial u}{\partial x_k}$ at each node, i say, given by

$$\frac{\partial u}{\partial x_k}|_i \approx L_i^{-1} \int N_i \frac{\partial u^h}{\partial x_k} d\underline{x} \ , \tag{2}$$

where N_i is the usual linear basis function at node i and L_i is the i^{th} diagonal of the row summed lumped mass matrix. From this it is straightforward to calculate a piecewise constant estimate of (1). The Hessian may then either be used as the basis for a new metric (with isotropic meshes as measured in this metric corresponding to anisotropic meshes in the Euclidean metric when the solution is highly directional), as in [11], or its eigen-decomposition may be used, as in [13, 12] for example. In these latter papers the Hessian is used to drive a remeshing algorithm based upon the assumption that a mesh with size inversely proportional to the eigenvalue magnitudes in the directions of the corresponding eigenvectors is desirable.

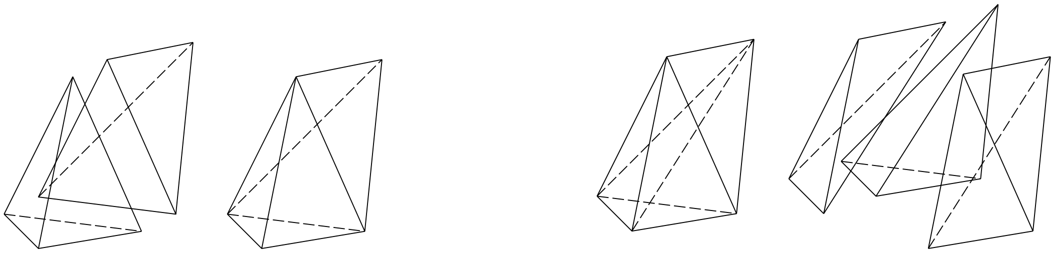
Another empirical technique that may be used to assess mesh quality is to construct an indicator based upon interpolation error estimates. This is the approach followed in [3, 4] for example, which build upon earlier work in [10], where an indicator is defined to take into account both geometry and the solution behaviour. The approach used is to assume that the exact solution can be approximated in a locally quadratic form on each tetrahedral element and then to consider the difference between this quadratic and a linear finite element interpolant. After some manipulation this yields an expression of the form

$$\int_K e^2 d\underline{x} = \frac{6}{4} V^K \frac{2}{7!} \left(\left(\sum_{s=1}^{6} d_s \right)^2 - d_1 d_4 - d_2 d_5 - d_3 d_6 + \sum_{s=1}^{6} d_s^2 \right) \tag{3}$$

for the interpolation error on each element K, of volume V^K. Here d_s denotes the directed edge second derivative for an edge $s = s(i,j)$, that connects nodes i and j, and it may be defined in a similar manner to that used to recover the Hessian of a piecewise linear approximation described above. In the next section we describe a scheme which uses this edge information to drive an adaptive strategy which aims to improve mesh quality for convection-dominated problems.

2 MESH GENERATION

The adaptive mesh generation algorithm that we outline here is based upon a combination of standard h-refinement, node movement (r-refinement) and a limited amount of *a priori* edge swapping. In [7] it is demonstrated that the solution accuracy for a linear hyperbolic problem can be increased by simple mesh reconnection to better align the mesh with the convection direction. This involves edge swapping in 2-d or a face swapping operation in 3-d that is shown in Figure 1.



(a) Two initial tetrahedra

(b) Three aligned tetrahedra

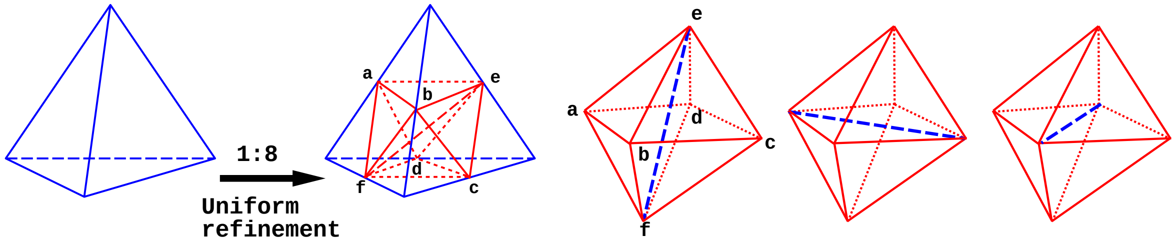
Figure 1: 3-d alignment operation for 2 tetrahedra

Having modified the initial mesh through local face swapping the algorithm proceeds through a combination of local h- and r-refinement. The SUPG piecewise linear finite element solution is first obtained and the resulting edge second derivatives are computed. These derivative values are then used to drive a simple node movement scheme which aims to locally equidistribute them via a relaxation scheme in which the nodal positions, $\underline{x}_i$, are updated by

$$\underline{x}_i^{av} \;=\; \frac{\sum_{s\in\Omega_i} |d_s|\; \underline{x}_j}{\sum_{s\in\Omega_i} |d_s|}, \qquad \underline{x}_i \;\rightarrow\; (1-\gamma_i)\underline{x}_i + \gamma_i \underline{x}_i^{av}, \tag{4}$$

where Ω_i is the set of all edges, $s(i,j)$, connected to node i. Here γ_i is a safety factor at each node i that prevents the mesh from becoming tangled by restricting the node movement to be within one half of the perpendicular distance to the closest opposing tetrahedral face. In practice this procedure is implemented in a Jacobi fashion by first assembling all of the node position increments and then updating the entire mesh, and several sweeps are performed at each r-refinement stage. Note the similarity of equidistribution of the edge second derivatives with the Hessian approach, i.e. a tetrahedron with $|d_s|$ equidistributed over the edges should appear isotropic within the Hessian metric space.

Following r-refinement, conventional h-refinement is undertaken. The interpolation error estimate (3) is used to mark for refinement the 20% (say) of tetrahedra whose estimated error is the largest. When these elements are refined each edge is bisected and 8 children are produced as described in [15] for example (see Figure 2(a)). Note that this refinement procedure requires a choice to be made between three possible diagonal edges for the central octahedral volume, as illustrated in Figure 2(b). Typically this is selected based upon some geometric criteria however in this work we generalize the ideas of [7] and choose the diagonal that is best aligned with the convection direction. Some *hanging* nodes are left on edges of unrefined cells that have neighbours which have been refined and so, to ensure that the new mesh is conforming, temporary transitional refinements are introduced on such (initially) unrefined elements. Again see [15] for details. The key feature of these transitional refinements is that they are always removed before further regular refinement is permitted in these regions.



(a) Isotropic h-refinement (b) Diagonal choices for central volume

Figure 2: Isotropic refinement of a tetrahedron

By looping between h-refinement and a number of sweeps of r-refinement a simple anisotropic adaptive algorithm is created. Table 1 illustrates the performance of this algorithm (with two sweeps of r-refinement per loop) against the standard h-refinement approach for a simple linear hyperbolic model problem. This takes the form

$$(\underline{a}\cdot\underline{\nabla})u = f \quad \text{in } \Omega = (0,1)^3, \tag{5}$$

$$u = \begin{cases} 1 - x/\delta & x < \delta \\ 0 & x \geq \delta \end{cases} \quad \text{on } \Gamma_{in} = \{\underline{x} \in \partial\Omega : \underline{a}\cdot\underline{n}(\underline{x}) < 0\}, \tag{6}$$

where $\underline{n}(\underline{x})$ is the unit outward normal to the boundary $\partial\Omega$. Figure 3 illustrates the nature of the solution of this problem when $f(\underline{x}) = 0$, $\underline{a} = (2,1,1)^T$ and $0 < \delta \ll 1$. The figure shows the isosurface $u(\underline{x}) = 0.5$ and the solution changes rapidly from 0 to 1 as one moves across this isosurface. For the results shown in Table 1 we use $\delta = 0.01$ and the above choices of $f(\underline{x})$ and $\underline{a}$.

3 DISCUSSION

The results in Table 1 show a consistent reduction in error of about 20% over standard h-refinement when hr-refinement is used. Whilst these improvements are typical, there are various parameters in the node movement algorithm that can be tuned, such as the number of sweeps per loop and the number of times the solution is recomputed, so that even better results can nearly always be obtained with some experimentation. In practice the hr-refinement scheme always does better than h-refinement alone although in some

	Np	$\|e\|_1$	Np	$\|e\|_1$	Np	$\|e\|_1$	Np	$\|e\|_1$
hr-ref.	729	0.092 0.088 0.080	2690	0.054 0.051 0.048	11323	0.029 0.028 0.028	48930	0.020 0.018 0.017
h-ref.	729	0.092	2637	0.059	11182	0.035	49225	0.021

Table 1: a comparison of the performance of the anisotropic and the standard adaptive algorithms for the linear test problem

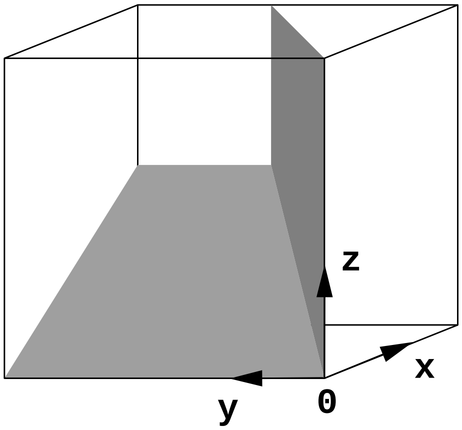


Figure 3: The strongly directional layer that is present in the solution of the model problem

cases the improvement is much greater than in others. It should also be emphasized that for both sets if results presented in Table 1 the initial mesh alignment procedure has been undertaken and, as demonstrated in [7, 17], this has a significant improvement on the quality of the results even with standard h-refinement. The effect of the aligned choice of diagonal when element subdivision takes place, whilst certainly beneficial, is substantially less marked than the effect of this *a priori* alignment.

In [17] and [18] results for the same test problem that is used here are also presented using two different node movement and h-refinement strategies, both of which make explicit use of the equation being solved (unlike the strategy here which depends only upon the solution computed). The first of these makes use of the residual of the equation once an approximate solution has been computed (see also [14] for further motivation) whilst the other makes use of an *a posteriori* error estimate (see also [6, 16], for example). The simple algorithm described here is highly competitive with both of these.

Like this simple algorithm, a more sophisticated adaptive procedure recently proposed in [11] also relies entirely on the computed solution, rather than the underlying equation, in order to drive refinement. A complex combination of edge collapsing, edge splitting, face to edge swapping, edge to face swapping, edge swapping and node movement are used to adapt an initial mesh based upon a local cost function which relies upon a Hessian metric.

It would be interesting to contrast the performance of these two algorithms in order to quantify the benefits of the additional complexity of the adaptivity in [11]. In particular, our empirical evidence suggests that when it is possible to apply the *a priori* alignment approach of [7], the subsequent edge and face swapping steps may be of only marginal benefit for many steady-state problems.

Further work to be undertaken also includes the introduction of a restricted version of the r-refinement algorithm, so that only nodes at a given level of the mesh's h-refinement hierarchy are allowed to move independently, with all nodes at higher levels being dragged with them in a consistent manner. This idea has been proposed in [18] in the context of a residual-based adaptive algorithm and solver and may be applied in this work by modifying (4) accordingly. Once implemented the hr-adaptive solution of time-dependent problems becomes a simple extension since, by only moving nodes at the lowest level of the mesh hierarchy, it is still possible to apply both h-refinement and derefinement (necessary for time-dependent problems, [15]) without disturbing the mesh hierarchy or data structures within the adaptive h-refinement code.

Acknowledgments: This work was funded by EPSRC research grant GR/M00077.

REFERENCES

[1] T. Apel, S. Grosman, P.K. Jimack and A. Meyer, "A new methodology for anisotropic mesh refinement based upon error gradients", *Sonderforschungsbereich 393, Technical Report*, **SFB393/01-11**, Technische Universität Chemnitz, (2001).

[2] I. Babuska and J. Aziz, "On the angle condition in the finite element method", *SIAM J. Num. Anal.*, vol. **12**, 394–409, (1968).

[3] M. Berzins, "A solution-based triangular and tetrahedral mesh quality indicator", *SIAM J. Sci. Comp.*, vol. **19**, 2051–2060, (1998).

[4] M. Berzins, "Solution-based mesh quality indicators for triangular and tetrahedral meshes", *Int. J. Comp. Geom. Appl.*, vol. **10**, 333–346, (2000).

[5] M. Berzins, "An introduction to mesh quality", *31st Computational Fluid Dynamics Lecture Series*, ed. N.P. Weatherill and H. Deconinck, von Karman Institute, (2000).

[6] P. Houston and E. Süli, "A posteriori error indicators for hyperbolic problems", *Oxford University Computing Laboratory Technical Report*, **97/14**, Oxford University, (1997).

[7] T. Iliescu, "A 3D flow-aligning algorithm for convection-diffusion problems", *App. Math. Letters*, vol. **12**, 67–70, (1991).

[8] M. Krizek, "On the maximum angle condition for linear tetrahedral elements", *SIAM J. Numer. Anal.*, vol. **29**, 513–520, (1992).

[9] G. Kunert, *A Posteriori Error Estimation for Anisotropic Tetrahedral and Triangular Finite Element Meshes*, PhD thesis, Math. Dept., Tech. Univ. Chemnitz, 1999.

[10] E.J. Nadler, "Piecewise linear best L_2 approximation on triangles", *Approximation Theory V: Proceedings Fifth International Symposium on Approximation Theory*, ed. L.L. Schumacher and J.D. Ward, Academic Press, New York, 499–502, (1986).

[11] C.C. Pain, A.P. Umpleby, C.R.E. de Oliviera and A.J.H. Goddard, "Tetrahedral mesh optimisation and adaptivity for steady-state and transient finite element calculations", *Comput. Methods Appl. Mech. Engrg.*, vol. **190**, 3771–3796, (2001).

[12] J. Peraire, J. Peiro, L. Formaggia, K. Morgan and O.C. Zienkiewicz, "Finite element Euler computations in three dimensions", Int. J. Num. Meth. Eng., vol. **26**, 2135–2159, (1988).

[13] J. Peraire, M. Vahdati, K. Morgan and O.C. Zienkiewicz, "Adaptive remeshing for compressible flow computations", *J. Comp. Phys.*, **72**, 449–466, (1987).

[14] P. Roe, “Compounded of many simplices. Reflections on the role of model problems in CFD”, *Barriers and Challenges in Computational Fluid Dynamics*, 241–258, Kluwer Academic, (1998).

[15] W. Speares and M. Berzins, “A 3-d unstructured mesh adaptation algorithm for time-dependent shock dominated problems”, *Int. J. Num. Meth. Fluids*, vol. **25**, 81–104, (1997).

[16] T. Strouboulis and J.T. Oden, “A posteriori error estimation of finite element approximations in fluid mechanics” *Comput. Methods Appl. Mech. Engrg.*, vol. **78**, 201–242, (1990).

[17] M. Walkley, P.K. Jimack, and M. Berzins, “Mesh quality for three-dimensional finite element solutions on anisotropic meshes”, *Proc. of FEM3D, Univ. of Jyvaskyla, Finland, 2000*, (to appear).

[18] M. Walkley, P.K. Jimack, and M. Berzins, “Anisotropic adaptivity for finite element solutions of 3-d convection-dominated problems”, *Numerical Methods for Fluid Dynamics 7*, ed. M.J. Baines, ICFD, Oxford (to appear).

[19] M. Zlamal, “On the finite element method”, *Numer. Math.*, **12**, 394–409, (1968).