

# A Multilevel Approach for Obtaining Locally Optimal Finite Element Meshes

Peter K. Jimack and Rashid Mahmood  
Computational PDE Unit  
School of Computer Studies  
University of Leeds  
Leeds LS2 9JT, UK

## Abstract

In this paper we consider the adaptive finite element solution of a general class of variational problems using a combination of node insertion, node movement and edge swapping. The adaptive strategy that is proposed is based upon the construction of a hierarchy of locally optimal meshes starting with a coarse grid for which the location and connectivity of the nodes is optimized. This grid is then locally refined and the new mesh is optimized in the same manner. Results presented indicate that this approach is able to produce better meshes than those possible by more conventional adaptive strategies and in a relatively efficient manner.

## 1 Introduction

Automatic mesh generation is an important computational tool for the finite element analysis of a wide variety of engineering problems ranging from structural analysis through to computational fluid dynamics for example. For many of these problems the use of *unstructured* grids offers many advantages over structured grids, such as permitting the straightforward triangulation of geometrically complex domains or allowing the mesh density to be adapted according to the behaviour of the solution. In this paper we are concerned mainly with the latter of these properties of unstructured grids: the natural manner in which numerous forms of mesh adaptivity are permitted.

Broadly speaking mesh adaptivity algorithms may be categorised as belonging to one of two classes. The first of these, generally referred to as *h*-refinement, involves adding vertices and elements to the mesh in some manner. This may be through local refinement (e.g. [10]) or through more global remeshing (e.g. [14]) but has the general aim of increasing the number of vertices and elements in those regions of the domain where some measure of the error is unacceptably high. The second class of approach, often referred to as *r*-refinement, adapts the mesh in such a way that the number of vertices and elements remains essentially unchanged. This is typically achieved through the use of node movement (e.g. [7, 8]), where the mesh is continuously deformed so as to increase the density of vertices in those regions of the domain with the highest errors, or through the use of edge swapping (e.g. [11]), where the number and position of the vertices is held fixed but the way in which they are connected together is allowed to change. (There is also a third class of adaptive algorithm, known as *p*-

refinement, which involves increasing the degree of the finite element approximation space on a fixed mesh, however we do not consider this approach here. See, for example, [1] or [15] for further details.)

In this paper we present a new hybrid algorithm that combines the local insertion and movement of vertices with the local swapping of edges in order to attempt to obtain optimal finite element meshes for a general class of problem. These are variational problems which may be posed in the following form (or similar, according to the precise nature of the boundary conditions):

$$\min_{u: \Omega \subset \mathcal{R}^m \rightarrow \mathcal{R}^n} \int_{\Omega} F(\underline{x}, u, \nabla u) d\underline{x} \quad (1)$$

for some energy density function  $F : \mathcal{R}^m \times \mathcal{R}^n \times \mathcal{R}^{m \times n} \rightarrow \mathcal{R}$ . Physically this variational form may be used to model problems in linear and nonlinear elasticity, heat and electrical conduction, motion by mean curvature and many more (see, for example, [2],[7],[8]). Throughout this paper we restrict our attention to the two-dimensional case where  $m = 2$ .

For variational problems of the form (1), the fact that the exact solution minimises the energy functional provides a natural optimality criterion for the design of computational grids using *r*-refinement. Indeed, the idea of locally minimising the energy with respect to the location of the vertices of a mesh of fixed topology has been considered by a number of authors (e.g. [5],[13]), as has the approach of locally minimising the energy with respect to the connectivity of a mesh with fixed vertices (e.g. [11]). Accordingly, the specific algorithms that we use for node movement are generalizations of those used in [6] and [13], and the edge swapping is based upon [11]. Full details of these algorithms and how they are combined with local *h*-refinement are presented in the following section.

In section 3 it is then demonstrated that combining the above *r*-refinement and *h*-refinement approaches in an appropriate manner allows locally optimal grids to be obtained which are better (in terms of energy minimisation) than using either strategy alone. The approach taken is to start with a very coarse mesh which is optimised using *r*-refinement. This is then refined locally to create a new mesh with a greater number of elements and vertices which can itself be optimised. By repeating this process a number of times a hierarchy of locally optimal meshes is obtained. Since the initial mesh at each level of the hierarchy is produced by local refinement of an optimal mesh at the previous level it follows that this typically provides a reliable starting point when optimising the new mesh. The results presented demonstrate that the proposed hybrid algorithm

is able to provide a mesh which allows the solution of (1) to be approximated to an arbitrary error tolerance using substantially fewer vertices and elements than through  $h$ -refinement alone. Furthermore, it is also demonstrated that, for a fixed size of mesh, this multilevel approach invariably finds a better locally optimal solution than is obtained by applying  $r$ -refinement directly to a regular starting mesh of the same fixed size.

The paper concludes with a discussion of the strengths and weaknesses of the the proposed hybrid algorithm for obtaining locally optimal meshes and a short discussion of necessary further work.

## 2 Multilevel adaptivity

In this section we consider the adaptive finite element solution of problems of the form (1), first using  $r$ -refinement and then adding  $h$ -refinement to obtain a sequence of optimal meshes. The  $r$ -refinement approach is described first and consists of a combination of node movement and edge swapping in order to minimize the energy functional for a given size of mesh. In this work the mesh may be any triangulation of the domain  $\Omega$ , which is assumed to be a subset of  $\mathcal{R}^2$  (i.e.  $m = 2$  in equation (1)), and we only consider piecewise linear finite element trial functions.

### 2.1 Locally optimal meshes

We define a locally optimal mesh for the finite element solution of (1) to be a mesh with the following properties.

1. There exists some number  $\varepsilon > 0$  such that if any node is displaced by any distance  $\delta \leq \varepsilon$  in any direction (subject to the constraint that a boundary node remains on the boundary and the domain is not altered) the finite element solution on the new mesh has an energy which is no less than the energy of the finite element solution on the locally optimal mesh.
2. By noting that each internal edge of a triangulation is shared by exactly two triangles then, if the union of these two triangles is a convex quadrilateral, we may obtain a modified triangulation by swapping the diagonal of this quadrilateral, as shown in Figure 1. The finite element solution on any such modified triangulation has an energy which is no less than the energy of the finite element solution on the locally optimal mesh.

In order to obtain such a mesh from a given starting mesh we use an approach which is based heavily on that of [13]. This approach combines node movement and edge swapping in a manner which only requires the solution of local problems in order to converge to a global solution of the full problem, (1), on a locally optimal grid. For clarity we describe the node movement and the edge swapping algorithms separately and then discuss how they may be combined.

A necessary condition for the position of each node of the triangulation to be optimal is that the derivative of the energy functional with respect to each nodal position is zero. Like the approach of [13] our algorithm seeks to reduce the energy functional monotonically by moving each node in turn until the derivative with respect to the position of each node is zero.

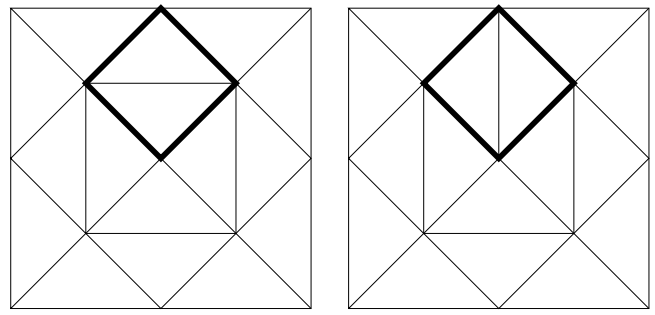


Figure 1: An illustration of the modification of a mesh by the swapping of a single edge.

Whilst this does not guarantee with absolute certainty that we reach a local minimum (as opposed to a saddle point or even a local maximum), the presence of rounding errors combined with the downhill nature of the technique ensures that in practice any other outcome is almost impossible.

The algorithm proposed consists of a number of sweeps through each of the nodes in turn until convergence is achieved. At the beginning of each sweep the gradient, with respect to the position of each node, of the energy functional

$$E = \int_{\Omega} F(\underline{x}, u^h, \nabla u^h) d\underline{x} \quad (2)$$

is found (where  $u^h$  is the latest piecewise linear finite element solution). This is done using a slightly different approach to that described in [13], based upon [8] instead. In [8] it is proved that if  $\underline{s}_i$  is the position vector of node  $i$  then

$$\frac{\partial E}{\partial s_{id}} = \int_{\Omega} \left\{ \left[ F \delta_{dj} - \frac{\partial u_k^h}{\partial x_d} F_{,3kj} \right] \frac{\partial \alpha_i}{\partial x_j} + F_{,1d} \alpha_i \right\} dx, \quad (3)$$

where  $\alpha_i$  is the usual local piecewise linear basis function at node  $i$ ,  $s_{id}$  is the  $d^{\text{th}}$  component of  $\underline{s}_i$  ( $i = 1, 2$ ),  $F_{,p}$  represents the derivative of  $F$  with respect to its  $p^{\text{th}}$  argument, other suffices represent components of tensors,  $\delta_{dj}$  is the Kronecker delta and repeated suffices imply summation ( $j = 1$  to  $2$  and  $k = 1$  to  $n$ ). Note that using (3) the gradients with respect to all of the vertices in the mesh may be assembled in a single pass of the elements. These gradients are then sorted by magnitude and the nodes visited one at a time, starting with the largest gradient.

When each node is visited the direction of steepest decent,

$$\underline{s} = -\frac{\partial E}{\partial \underline{s}_i}, \quad (4)$$

is used in order to determine along which line the node should be moved. The distance that the node is moved along this line is computed using a one-dimensional minimization of the energy subject to the constraint that the node should not move more than a proportion  $w$  ( $0 < w < 1$ ) of the distance from its initial position to its opposite edge (see Figure 2). Once a new position for the node has been found the value of the solution,  $u_i$  say, at that node must be updated by solving the local problem

$$\min_{u_i \in \mathcal{R}^n} \int_{\Omega_i} F(\underline{x}, u^h, \nabla u^h) d\underline{x}. \quad (5)$$

Here  $\Omega_i$  is the union of all elements which have node  $i$  as a vertex and Dirichlet conditions are imposed on  $\partial\Omega_i$  using the latest

values for  $u^h$ . Once this update is complete the same process is undertaken for the next node in the sorted list and when each node has been visited the sweep is complete. Provided convergence has not been achieved the next sweep may then begin.

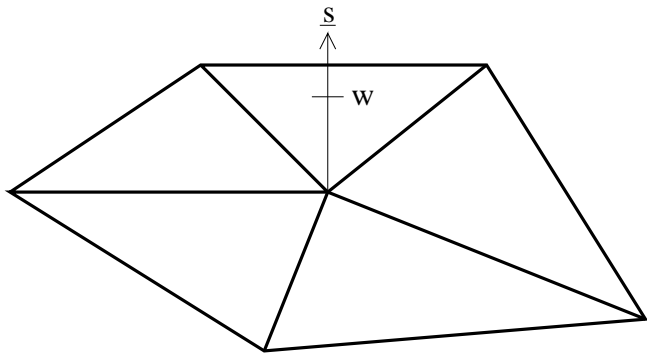


Figure 2: An illustration of local node movement:  $\underline{s}$  is the direction of steepest descent for the node motion and  $w$  represents the maximum distance that the node may move in this direction.

Using the above approach the interior nodes could move in any direction however a slight modification is required for nodes on the boundary of  $\Omega$ . These nodes may only be moved tangentially along the boundary and even then this is subject to the constraint that the domain remains unaltered. Where this constraint is not violated the downhill direction of motion along the boundary is easily computed by projecting  $\underline{s}$  from (4) onto the local tangent of the boundary. The one-dimensional minimization in this direction is then completed as for any other node. On Dirichlet boundaries the updated value of  $\underline{u}_i$  is of course prescribed however on any other type of boundary it must be computed by solving a local problem of the same form as (5).

Once convergence with respect to the position of each node has been achieved a further reduction in the energy of the solution is sought by the use of edge swapping. Following [11] a loop through each of the internal edges of the mesh is completed and, for each edge, the local energy on the two triangles on either side is computed. The edge is then swapped in the manner illustrated in Figure 1 and the new local energy over the two triangles on either side is computed. If this energy is less than the original local energy on the quadrilateral then the edge swap is kept; otherwise it is rejected. Once the loop through each of the edges has been completed it is repeated until there is an entire pass for which no edges are swapped.

Of course the grid is unlikely to be locally optimal at this point since the edge swapping will generally cause the node locations to become sub-optimal. Hence it is necessary to alternate between the node movement and the edge swapping algorithms until the whole process has converged. The downhill nature of each step in the process guarantees that this will eventually occur. Despite this guarantee however, for pragmatic reasons it is useful to be able to impose a small number of additional constraints on the allowable meshes. For example, in our implementation of the edge swapping algorithm parameters are provided for the maximum number of edges that may be connected to a single node and for the smallest angle allowed in any triangle. Similarly, for the node movement algorithm a param-

eter is provided to specify the smallest area allowed for any element (and any triangle which shrinks to a size below this threshold is removed from the mesh by a simple element/node deletion algorithm). Numerous other such parameters could also be included.

## 2.2 Local mesh refinement

The main difficulty with the  $r$ -refinement strategy introduced in the previous subsection is that it is impossible to know *a priori* how many nodes or elements will be required in order to get a sufficiently accurate finite element solution to any given variational problem. Even an optimal mesh with a given number of nodes may not be adequate for obtaining a solution of a desired accuracy. For this reason some form of mesh refinement is essential.

In [13] global mesh refinement is combined with  $r$ -refinement and it is demonstrated that this provides better solutions than the use of global  $h$ -refinement alone. In this work we extend these results in a number of ways. Firstly, by generalizing to systems of equations (i.e.  $n > 1$  in (1)) and secondly, by using local (rather than global)  $h$ -refinement in conjunction with  $r$ -refinement. This, we demonstrate, leads to further efficiency gains above and beyond those observed in [13]. In addition, we also demonstrate that the hierarchical approach of starting with a coarse grid and then optimizing, refining, optimizing, refining, etc., provides a far more robust adaptive algorithm than simply refining first and then optimizing the node positions and the mesh topology at the end.

For the purposes of this work two different local refinement algorithms have been considered. The first of these divides all triangles which are to be refined into four children (as used in [10] for example and illustrated in the top half of Figure 3) whilst the other divides all triangles which are to be refined into just two children (as used in [4] for example and shown in the bottom half of Figure 3). In each case any ‘‘hanging nodes’’ left at the end (again see Figure 3) are removed by bisecting the neighbouring elements and then performing local edge swapping.

There are many possible ways in which the  $h$ -refinement might be combined with the  $r$ -refinement to produce a hybrid algorithm. Our experience suggests that a robust approach is to always refine an optimized mesh and then to interpolate the coarser solution onto the refined mesh as a starting point for the next level of optimization. It also appears to be advantageous to approximately optimize the nodal solution values first before attempting to optimize the positions. The local refinement itself attempts to subdivide all elements for which the local energy is greater than  $X\%$  of the largest energy value calculated on any single element. Typically  $X$  is chosen to be somewhere between 40 and 80.

In the following section the performance of this hybrid algorithm is assessed using a number of test problems. In each case comparisons are made with the approach of [13], in which  $r$ -refinement is combined with global  $h$ -refinement, and with the more conventional approach of using local  $h$ -refinement on its own.

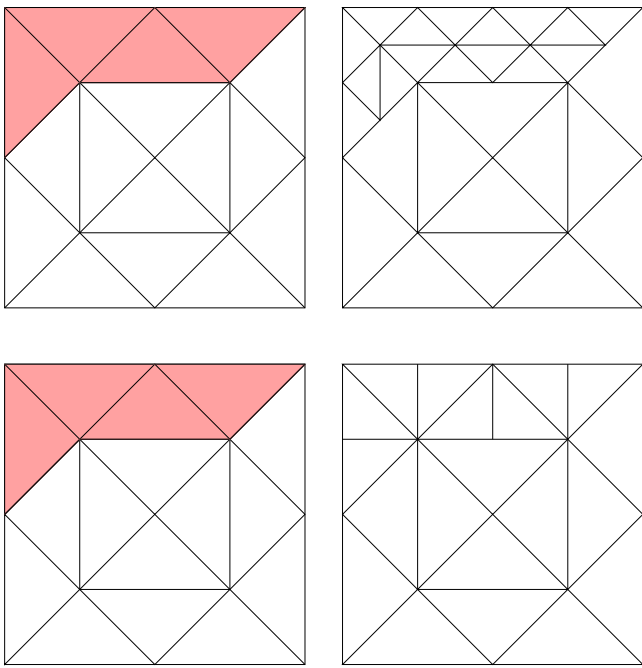


Figure 3: An illustration of the refinement of certain (shaded) elements of a mesh using one-to-four subdivision (top) and one-to-two subdivision (bottom).

### 3 Numerical results

In this section we study two representative test problems in order to assess the quality of the adaptive technique that has been described. The first of these problems is taken from [8] and [12] whilst the second appears in [3], [9] and [13].

#### 3.1 Problem one

We begin by considering the displacement field for a two-dimensional linear elastic model of an overhanging cantilever beam supporting a vertical point load at the end of the cantilever, as illustrated in Figure 4. For this problem  $m = n = 2$  and the energy functional is given by

$$E = \frac{1}{2} \int_{\Omega} \frac{\partial u_i}{\partial x_j} C_{ijkl} \frac{\partial u_k}{\partial x_l} d\mathbf{x} - \int_{\Omega} \rho b_i u_i d\mathbf{x} - \int_{\partial_{\theta}} \theta_i u_i ds. \quad (6)$$

Here, all repeated suffices are summed from 1 to 2,  $\mathbf{C}$  is the usual fourth order elasticity tensor (in this case corresponding to a Young's modulus  $E = 100$  and a Poisson ratio  $\nu = 0.001$ ),  $\rho \underline{b}$  provides the external body forces due to gravity and  $\underline{\theta}$  represents the traction boundary condition (in this case a point load as illustrated in Figure 4). The left half of the lower boundary is fixed whilst the rest of the boundary,  $\partial_{\theta}$  say, is free.

Initially the problem is solved on a uniform coarse grid containing just 64 elements. This grid is then optimized using the  $r$ -refinement approach of Subsection 2.1 and the total energy reduces from  $-0.201352$  to  $-0.253210$ . Following [13] this optimal grid may now be uniformly refined to produce 256 elements which may themselves be optimized. This leads to a solution with a total stored energy of  $-0.302353$ . A further global refinement and optimization then leads to a solution with a total stored energy of  $-0.338964$  on a mesh of 1024 elements: this sequence of locally optimal meshes is shown in Figure 5.

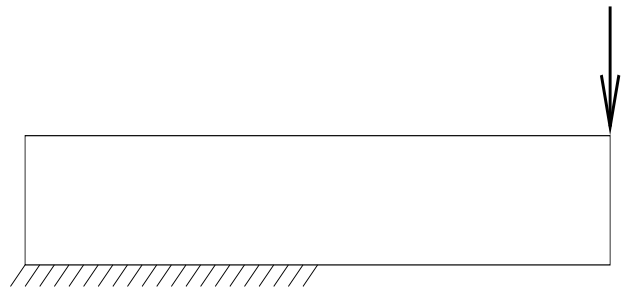


Figure 4: An illustration of the overhanging cantilever beam with a vertical point load at the end of the cantilever.

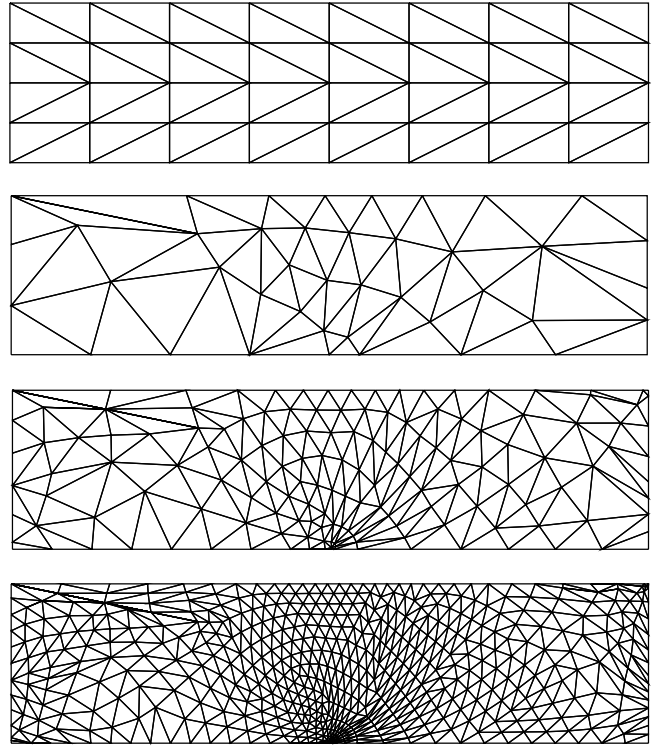


Figure 5: An initial mesh followed by a sequence of meshes obtained by  $r$ -refinement and then combinations of global  $h$ -refinement with  $r$ -refinement.

Figure 6 illustrates two further meshes of 1024 elements: the first obtained by global refinement of the initial uniform mesh and the second by optimizing this grid directly. The energies of the solutions on these meshes are  $-0.306791$  and  $-0.329249$  respectively thus illustrating, for this example at least, the superiority of the hierarchical approach when  $r$ -refinement is combined with global  $h$ -refinement.

The purpose of this paper however is to propose that the hybrid algorithm should combine  $r$ -refinement with local  $h$ -refinement and Figure 7 shows a sequence of meshes obtained in this manner. The first mesh is the same one, containing 64 elements, that appears in Figure 5, whilst the second and third meshes contain 224 and 455 elements respectively and were obtained by refining into 2 children only those elements whose local energy exceeded 60% of the maximum local energy on any element. The total energies of the solutions on these three meshes are  $-0.253210$ ,  $-0.308351$  and  $-0.363313$  respectively: clearly illustrating the superiority of the use of local rather than global  $h$ -refinement within the hybrid algorithm.

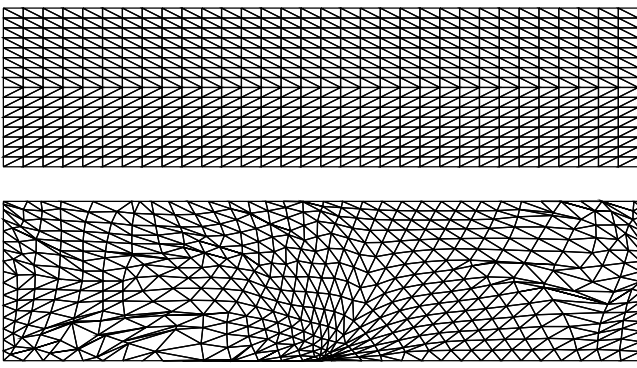


Figure 6: A globally refined mesh of 1024 elements and the corresponding locally optimized mesh.

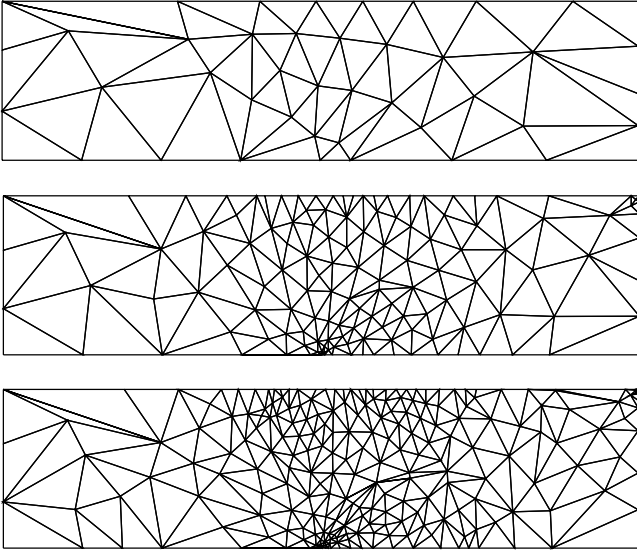


Figure 7: A sequence of meshes obtained by  $r$ -refinement of an initial coarse grid and then combinations of local  $h$ -refinement followed by  $r$ -refinement.

To conclude our discussion of this example we illustrate the advantage of applying the hybrid approach hierarchically by contrasting it with the use of local  $h$ -refinement alone, possibly followed by a single application of  $r$ -refinement. Figure 8 shows two grids of 674 elements and two grids of 462 elements that were obtained in this manner (again using a threshold of  $X = 60$  for the local refinement). The total energies of the solutions on the 674 element meshes (obtained by local one-to-four refinement alone and then a single application of the mesh optimization at the end) are  $-0.325679$  and  $-0.342525$  respectively, whilst the total energies of the solutions on the 462 element meshes (obtained by local one-to-two refinement plus a final optimization) are  $-0.325879$  and  $-0.342355$  respectively. We see that in both cases, despite the fact that the second of each pair of meshes is locally optimal, the quality of these local optima are not as good as that obtained using the hierarchical approach.

### 3.2 Problem two

The second problem that we consider involves just a single variable (i.e.  $n = 1$  in (1)) but has a solution which is singular at the origin. The energy functional corresponds to the Lapla-

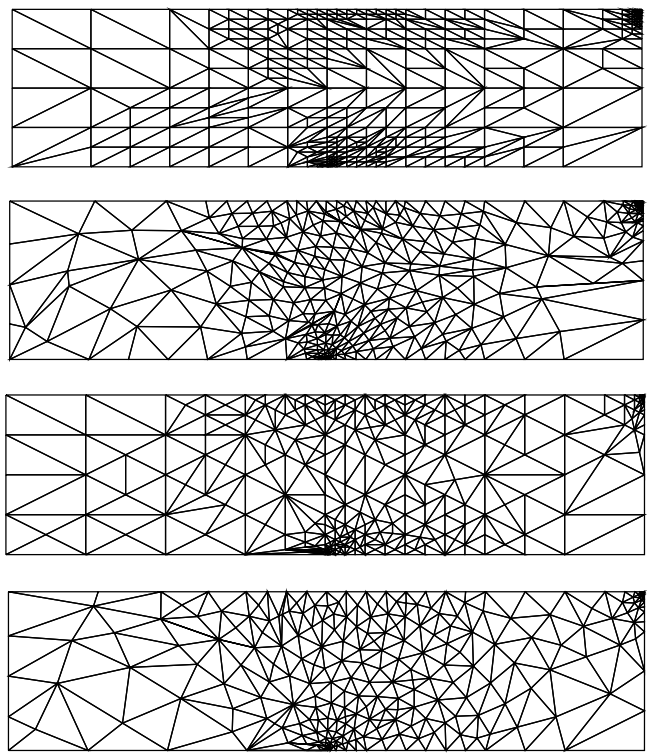


Figure 8: A pair of meshes of 674 elements obtained using local one-to-four  $h$ -refinement (top) followed by optimization (second) and a pair of meshes of 462 elements obtained using local one-to-two  $h$ -refinement (third) followed by optimization (bottom).

cian operator and is given by

$$E = \frac{1}{2} \int_{\Omega} \frac{\partial u}{\partial x_i} \frac{\partial u}{\partial x_i} d\mathbf{x}, \quad (7)$$

where the presence of repeated suffices again implies summation from 1 to 2. The domain,  $\Omega$ , is the unit disc with a  $45^\circ$  sector removed, as illustrated in Figure 9, and Dirichlet boundary conditions consistent with the exact solution  $u = r^{2/7} \sin \frac{2\theta}{7}$  are applied throughout  $\partial\Omega$ . Since the exact solution is known in this case, so is the true value of the global minimum of  $E$  in (7):  $0.392699$ .

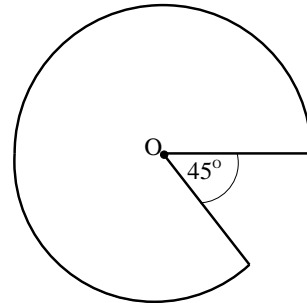


Figure 9: An illustration of the domain for the singular problem.

As with the previous example the problem is first solved on a coarse initial mesh, in this case with just 28 elements, which is then optimized. This locally optimal mesh is then refined globally and optimized to three further levels, giving meshes of 112,

448 and 1792 elements respectively. These meshes are shown in Figure 10 and their corresponding solutions have energies of 0.549242, 0.434828, 0.404352 and 0.396215.

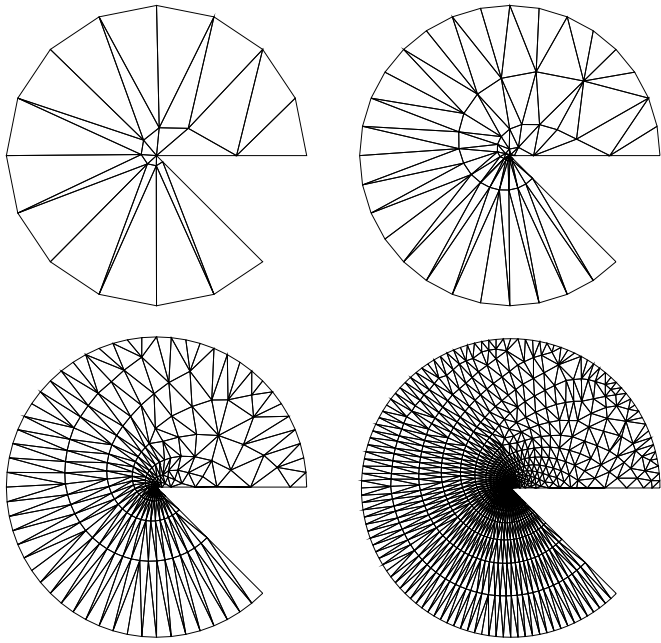


Figure 10: A sequence of meshes obtained by  $r$ -refinement of an initial coarse grid (top left) and then combinations of global  $h$ -refinement followed by  $r$ -refinement.

Once again, it may be observed that the approach of optimizing the mesh at each level after global refinement is superior to applying global  $h$ -refinement alone and then optimizing the resulting mesh. Figure 11 shows two meshes, each containing 1792 elements, that were obtained by this method. The energies of the solutions on these meshes are 0.438164 (uniform  $h$ -refinement only) and 0.405547 (after optimization), which are significantly worse than for the final mesh of Figure 10.

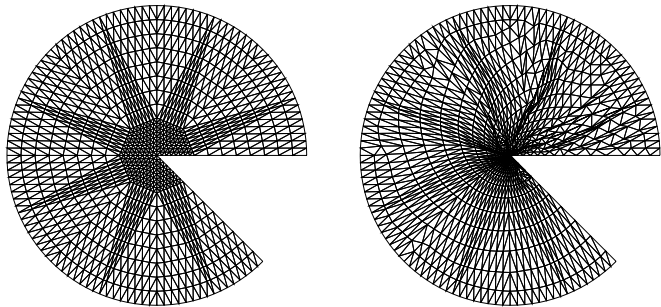


Figure 11: A globally refined mesh of 1792 elements and the corresponding locally optimized mesh.

To conclude this example, we now consider the application of local  $h$ -refinement in our hybrid algorithm. Figure 12 shows a sequence of four meshes of 28, 107, 255 and 1275 elements respectively. In order to contrast the solutions on these meshes with those obtained on the meshes shown in Figure 10 we have forced refinement of each of the edges on the circular boundary so that the domains correspond to the four domains in Figure 10. Further refinement (one element to two children) has then been permitted locally for any elements whose energy is greater than 60% of the maximum energy on any single element. This

local refinement is executed repeatedly on each domain until it is necessary to refine the boundary elements again. The total energies of the solutions on the four meshes shown in Figure 12 are 0.549242 (the same mesh as in Figure 10), 0.431777, 0.402413 and 0.395183 respectively.

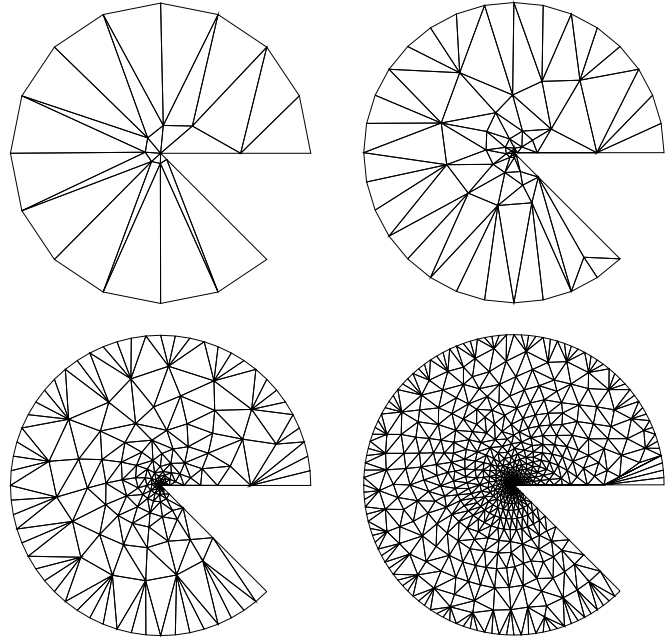


Figure 12: A sequence of meshes obtained by  $r$ -refinement of an initial coarse grid (top left) and then combinations of local  $h$ -refinement followed by  $r$ -refinement.

Again we have seen the advantage of using the hierarchical mesh optimization approach with local, rather than global, refinement. Furthermore, when local  $h$ -refinement is used on its own, even if this is followed by mesh optimization, the resulting grids are not as good. Two pairs of such grids, containing 1437 (one-to-four refinement) and 1413 (one-to-two refinement) elements respectively, are illustrated in Figure 13. For these examples the corresponding finite element solutions have total energies of 0.407613 and 0.398523 (1437 elements before and after optimization) and 0.402199 and 0.398123 (1413 elements before and after optimization) respectively. (For the purposes of comparison, we have artificially refined those edges on the circular boundary so as to ensure that the domains are identical to the final domains in Figures 10 to 12.)

## 4 Discussion

The two examples of the previous section have clearly illustrated that the quality of the final mesh produced when using the proposed hybrid algorithm is better, in the sense that the finite element solution has a lower energy, than that obtained by using either  $h$ -refinement or  $r$ -refinement alone. Furthermore it is demonstrated that combining the mesh optimization with local  $h$ -refinement is superior to the global refinement approach used in [13]. Finally, the advantage of using the hierarchical approach, whereby the intermediate level meshes are optimized, is also apparent: an excellent combination of small mesh sizes and low energies for the corresponding finite element solutions being achieved.

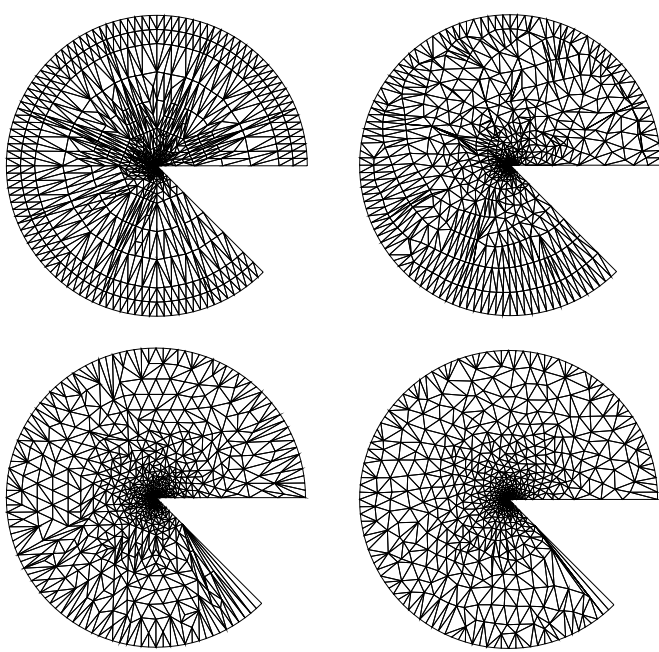


Figure 13: A pair of meshes of 1437 elements obtained using local one-to-four  $h$ -refinement (top left) followed by optimization and a pair of meshes of 1413 elements obtained using local one-to-two  $h$ -refinement (bottom left) followed by optimization.

When discussing the merits of our proposed algorithm it is important to note that there are some problems for which the benefits may not be quite so substantial as those observed in the two examples above. A common feature to both of these examples is the desirability of clustering the majority of the mesh elements in a relatively small subset of the domain. When a problem is such that the optimal mesh is more uniformly distributed across the domain the local refinement algorithm will show little or no advantage over the global approach of [13] since almost all elements of the mesh will need to be refined when moving from one level to the next. This is a phenomenon that we have observed in at least one example that we have considered (the nonlinear problem used as the second test problem in [13]). Nevertheless, even in this case, our variant of the algorithm performed no worse than that used in [13].

Further enhancements that need to be investigated relate mainly to the efficiency of the hybrid process that we propose. It is still an open question as to how accurately the mesh needs to be optimized at each level of the hierarchy before local refinement takes place for example. Also, it is unclear how accurately it is necessary to solve each of the one-dimensional minimization problems that are encountered at each node within each sweep of the node movement algorithm (see Figure 2). Other issues that should be considered further concern the importance of the order in which nodes and edges are visited during local optimization sweeps and the possibility of making more aggressive use of the element/node deletion algorithm that is currently only employed when elements shrink to zero.

## Acknowledgements

RM gratefully acknowledges the Government of Pakistan for financial support in the form of a merit scholarship.

## References

- [1] I. Babuska, B.A. Szabo and I.N. Katz, *The  $p$ -Version of the Finite Element Method*, SIAM Journal on Numerical Analysis, vol.18, pp.515–545, 1981.
- [2] J.M. Ball, P.K. Jimack and T. Qi, *Elastostatics in the presence of a temperature distribution or inhomogeneity*, Zeitschrift Fur Angewandte Mathematic Und Physik, vol.43, pp.943–973, 1992.
- [3] R.E. Bank, *PLTMG Users' Guide 7.0*, SIAM, Philadelphia, 1994.
- [4] E. Bänsch, *An adaptive finite element strategy for the 3-dimensional time-dependent Navier-Stokes equations*, Journal of Computational and Applied Mathematics, vol.36, pp.3–28, 1991.
- [5] M. Delfour, G. Payre and J.-P. Zolésio, *An optimal triangulation for second-order elliptic problems*, Computer Methods in Applied Mechanics and Engineering, vol.50, pp.231–261, 1985.
- [6] L.A. Frietag and C. Ollivier Gooch, *Tetrahedral mesh improvement using swapping and smoothing*, International Journal for Numerical Methods in Engineering, vol.40, pp.3979–4002, 1997.
- [7] P.K. Jimack, *A best approximation property of the moving finite element method*, SIAM Journal on Numerical Analysis, vol.33, pp.2206–2232, 1996.
- [8] P.K. Jimack, *An optimal finite element mesh for elastostatic structural analysis problems*, Computers and Structures, vol.64, pp.197–208, 1997.
- [9] C. Johnson, *Numerical solution of partial differential equations by the finite element method*, Cambridge University Press, Cambridge, 1987.
- [10] R. Lohner *An adaptive finite element scheme for transient problems in CFD*, Computer Methods in Applied Mechanics and Engineering, vol.61, pp.267–281, 1987.
- [11] S. Rippla and B. Schiff, *Minimum energy triangulations for elliptic problems*, Computer Methods in Applied Mechanics and Engineering, vol 84, pp.257–274, 1990.
- [12] B.H.V. Topping and A.I. Khan, *Parallel Finite Element Computations*, Saxe-Coburg Publications, Edinburgh, 1996.
- [13] Y. Tourigny and F. Hulsemann, *A new moving mesh algorithm for the finite element solution of variational problems*, SIAM Journal on Numerical Analysis, vol. 35, pp.1416–1438, 1998.
- [14] N.P. Weatherill, *Grid adaptation using a distribution of sources applied to inviscid compressible flow simulations*, International Journal for Numerical Methods in Fluids, vol 19, pp.739–764, 1994.
- [15] O.C. Zienkiewicz, D.W. Kelly and J.P. Gago, *The Hierarchical Concept in Finite Element Analysis*, International Journal of Computers and Structures, vol.16, pp.53–65, 1982.