LA-UR- 09-03453

Approved for public release;
distribution is unlimited.

Title: CONFORMAL REFINEMENT OF UNSTRUCTURED QUADRILATERAL MESHES

Author(s): RAO GARMIELLA

Intended for: PROCEEDINGS:
18TH INTERNATIONAL MESHING ROUNDTABLE

Los Alamos
NATIONAL LABORATORY
EST. 1943

Conformal Refinement of Unstructured Quadrilateral Meshes

Rao Garimella[1]

[1] MS B284, Los Alamos National Laboratory, P.O. Box 1663, Los Alamos, NM 87544

[2] rao@lanl.gov

Summary. We present a multilevel adaptive refinement technique for unstructured quadrilateral meshes in which the mesh is kept conformal at all times. This means that the refined mesh, like the original, is formed of only quadrilateral elements that intersect strictly along edges or at vertices, i.e., vertices of one quadrilateral element do not lie in an edge of another quadrilateral. Elements are refined using templates based on 1:3 refinement of edges. We demonstrate that by careful design of the refinement and coarsening strategy, we can maintain high quality elements in the refined mesh. We demonstrate the method on a number of examples with dynamically changing refinement regions.

1 Introduction

Adaptive mesh refinement is a well-known and widely employed technique for accurately capturing special features of the solution in steady and unsteady simulations. In such simulations, adaptive refinement enables the capturing of complex solution features by focusing refinement in critical areas without having to refine the mesh everywhere. Adaptive mesh refinement is now standard practice in simplicial meshes (triangular and tetrahedral) in a wide variety of applications. The unique topological properties of simplices allow the refinement in such meshes to be confined to fairly local regions while maintaining a high element quality [9] and keeping the mesh conforming. Conformity of the mesh implies that the intersection of a pair of elements, if not null, is strictly a lower dimensional mesh entity such as a face, an edge or a vertex. Non-conformity of mesh is commonly interpreted to mean that a lower order boundary entity (e.g. a vertex) of one element lies on a higher order boundary entity (e.g. an edge) of another element.

For quadrilateral meshes, the most common approach to adaptation is to refine elements in a non-conformal way. This allows the refinement to remain local but introduces non-conformal nodes which lie on the edges of neighboring elements. However, mesh non-conformity necessitates augmentation of the PDE solution algorithm to deal with the special nodes. Non-conformity is typically dealt with by constraining the solution at the non-conformal nodes to be dependent on the solution at the nodes of the edge it lies on using constraint equations [13] or Lagrange multipliers [3] or by the use of mortar elements to link the non-matching elements [1].

In this research we describe a technique to refine an unstructured quadrilateral mesh such that the result is also a heirarchically refined, conforming mesh of only quadrilaterals with high quality albeit a little worse than the parent mesh quality.

2 Previous Work

There has been considerable research on conformal triangular refinement for adaptive simulations since termination of refinement for simplices is very easy (see, for example, [10]). However, for quadrilateral meshes most researchers choose to use non-conformal quadtree type refinement with specialized code to handle non-conformal nodes (see, for example, [2]). There have been only a few articles describing conformal quadrilateral mesh refinement and coarsening, and even fewer that deal with the issue in a dynamic setting, i.e., conformally refining and coarsening a quadrilateral mesh that has been previously refined.

One of the best known papers on the issue of conformal quadrilateral refinement is by Schneiders [12]. In the paper, Schneiders discusses 2-refinement (bisection of edges) and 3-refinement (trisection of edges). He chooses the trisection of edges because it simplifies the algorithm. The refinement information is propagated from elements to nodes and refinement templates are defined based on the number of marked nodes (See Figure 2). The refinement templates are chosen such that the scheme is stable, i.e., the quality of elements does not deteriorate with increasing refinement levels. However, even though in this research, uniformly refined quadrilaterals have trisected edges and are split into 9 child quadrilaterals, templates used in adjacent elements to terminate the refinement have bisected edges as seen in the figure. In general, Schneiders scheme is expected to create more elements for the purpose of termination than the scheme presented here. Still, it is a valid scheme for conformal quadrilateral refinement and has been used by other researchers such as Zhang and Bajaj [16]. Schneiders has extended the work to hexahedral refinement as well but correctly points out that certain refinement patterns for the faces of hexahedra may not admit a valid decomposition of the parent hexahedron. Ito et.al. have also used Schneiders' approach for octree based hexahedral refinement templates [7]

Tchon et.al. have proposed a quadrilateral refinement strategy in which they find layers of elements, shrink the layers of elements and reconnect the shrunk layer with the surrounding mesh [15]. Clearly this strategy assumes certain structure to the mesh and specific refinement patterns while ignoring the issues of multiple levels of refinement, mesh quality and dynamic adaptation. Hence, the approach is of limited utility.

Several researchers have proposed a quadrilateral refinement strategy where the end result is a mixture of quadrilaterals and triangles, for example [4]. Similarly, others have proposed hexahedral refinement strategies which result in a combination of hexahedra and prisms. However, this conflicts with our stated goal of achieving a conforming all-quadrilateral or all-hexahedral mesh.

Benzley et.al. have proposed quadrilateral mesh coarsening strategies that are quite general and do have an advantage over nested refinement strategies in that they can coarsen beyond the original resolution of the mesh [14].

The research that is closest to our research is the work by Sandhu et.al. [11] although our work was developed without knowledge of this earlier research[3]. In this work, Sandhu et.al. use node marking and trisection of edges to define templates for refining elements and terminating the refinement. They define one less than the number of templates used in this work. Similar to our work, they also recommend undoing non-uniform refinement of quadrilaterals before further refinement to maintain quality. However, all their examples show only static refinement and not adaptation to dynamically changing solution features.

In this research we describe a dynamic mesh adaptation strategy for quadrilateral meshes that results in a conformal all-quadrilateral mesh with nested refinement. Moreover, while not proved, we beleieve that the resulting mesh quality is bounded by the quality of the parent mesh regardless of the number of levels of refinement at each time step or the number of time steps in the mesh. The adapted mesh is suitable for use in a wide range simulations without any special procedures since it is composed of only conformal quadrilaterals. Finally, the nested refinement allows for easy remapping of cell based quantities from one time step to another.

3 Description of Mesh Refinement/Coarsening Algorithm

3.1 Overview

Our adaptive mesh modification algorithm starts with tagging elements that must be refined because they do not adequately represent some geometric feature or because the solution error in these elements is deemed to be too high. These elements and their edges are tagged for refinement (or coarsening), if necessary, to multiple levels below (or above) their current level of refinement[4]. When an edge is adjacent to two elements with different refinement levels, it is refined to the higher of the two levels. Once the appropriate elements have been tagged by the application, the mesh is coarsened wherever the application requests the elements to be larger than they currently are. After coarsening, the mesh is refined wherever the application requests elements to be smaller than they currently are. During both coarsening and refinement, the refinement levels of elements are adjusted so that they are consistent with their siblings (children of their parents) and such that the target refinement levels of two adjacent elements do not differ by more than one. The one-level difference rule ensures that the number of templates required to make the mesh conforming is limited to a manageable number and that the mesh is smoothly graded.

[3]This work has not been published in any journal that frequently publishes meshing related work and therefore, has hitherto gone unnoticed.

[4]Regardless of whether an element is being coarsened or refinement, we will always refer to its target level in the heirarchy of meshes below the coarsest mesh as its target level of refinement.

3.2 Subdivision Templates

When some elements in the mesh get uniformly refined, adjacent quadrilaterals that share an edge with the uniformly refined elements have one or more edges that are refined. To make the mesh strictly conforming, these adjacent elements must also be subdivided into quadrilaterals such that no new edges are refined. To facilitate conformal subdivision of elements that are not uniformly refined, edges are trisected instead of being bisected as in triangular meshes. The reason for choosing trisection over bisection is that if an odd number of edges of a quadrilateral were bisected, the resulting polygon would have an odd number of edges and could not be subdivided into quadrilaterals. The templates used for subdividing quadrilaterals with different edges refined are shown below in Figure 1. Some of these templates have been described in previous works [11] and some are new.

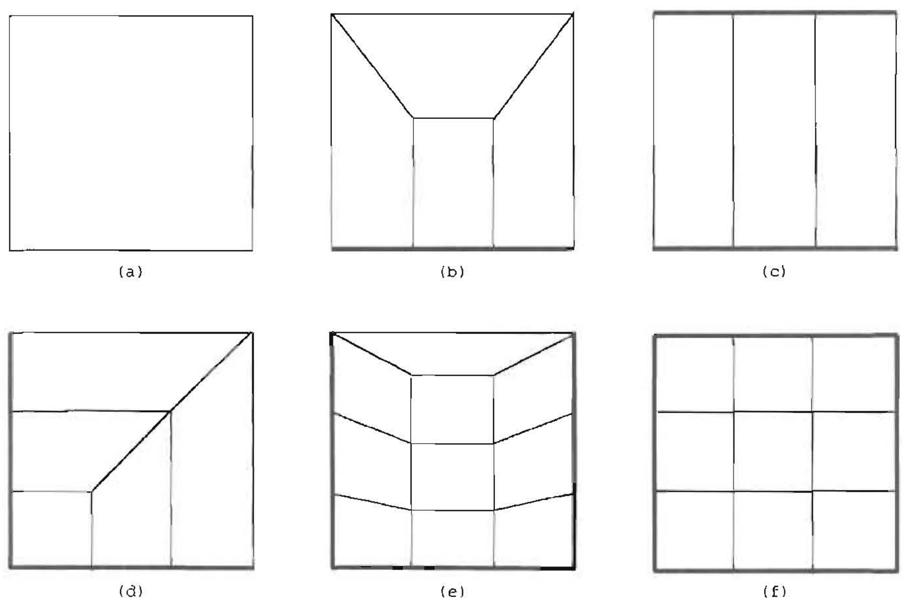


Fig. 1. Subdivision templates for quadrilateral refinement (thick edges are refined edges) (a) unrefined quadrilateral (b) one edge refined (c) two opposite edges refined (d) two adjacent edges refined (e) three edges refined (f) all edges refined (uniform refinement of quadrilateral).

The quadrilaterals that result from uniform refinement of a parent quadrilateral are called *regular* elements as they are geometrically similar to their parents. Quadrilaterals resulting from refinement of one, two or three edges of the parent quadrilateral are called *irregular* quadrilaterals as they are not necessarily similar to their parents.

It must be pointed out that the templates described above are different from the templates in Schneiders' work. In that work, refinement tags are transmitted to vertices of elements and templates derived from the combinations of vertices tagged for refinement. Those templates are shown in Figure 2. As can be seen from the picture, the only template the two approaches have in common is the uniform refinement template. In the remaining cases edges of elements that adjacent to uniformly refined elements are refined using an irregular 1:2 pattern. Also, even if only one edge of an element adjacent to a uniformly refined element is refined, the template proposed by Schneiders refines two other edges of the element. This in turn forces refinement of other elements. This leads us to conclude that the algorithm proposed by Schneiders is more complex to implement and results in greater numbers of elements. Figure 3 shows a simple example of this over-refinement as a consequence ofuniform refinement of the central element in a 3x3 mesh of quadrilaterals. As can be seen in the picture, Schneiders's scheme modifies every element in the 3x3 mesh while the proposed scheme affects only the edge connected neighbors.

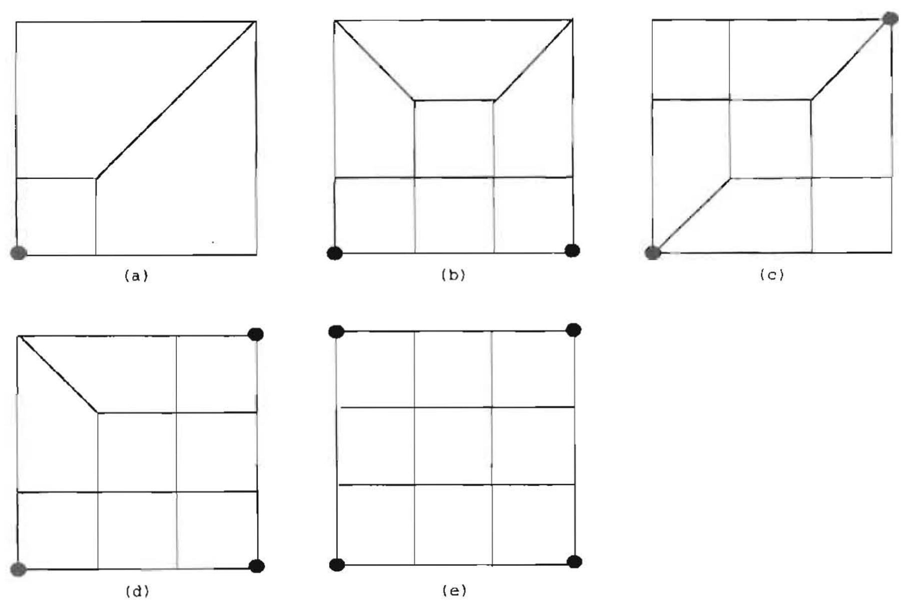


Fig. 2. Schneiders' subdivision templates for quadrilateral refinement (refinement vertices are marked with circles) (a) unrefined quadrilateral (b) one vertex marked (c) two adjacent vertices marked (d) two diagonally opposite vertices marked (e) three vertices marked (f) all vertices marked (uniform refinement of quadrilateral).

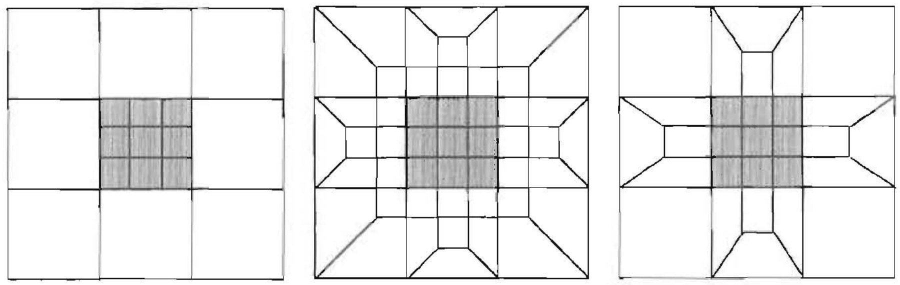


Fig. 3. Comparison of Schneiders' scheme and the proposed scheme of refinement on a 3x3 grid of quadrilaterals (a) Central element refined uniformly (b) Surrounding mesh made conforming by Schneiders' scheme (c) Surrounding mesh made conforming by proposed scheme

3.3 Coarsening

In the mesh adaptation method presented here, coarsening of elements is done first before refinement. In this approach, coarsening is performed strictly using the knowledge of the hierarchical structure of the adapted mesh, i.e., if an element is to be deleted then its siblings are also deleted simultaneously and the parent element is restored. For this reason, the coarsening strategy of this paper cannot coarsen beyond the original mesh. Coarsening is performed on elements whose current refinement level is higher than the target refinement level. Before actual deletion of elements, however, the target levels of elements are adjusted to ensure that there is not more than one level of difference between two adjacent elements and that the target levels of siblings are consistent.

Consider an element whose current refinement level is L_c and target refinement level is L_t. Assume the maximum refinement level of all of its edge connected neighbors (and therefore, all of its edges) is L_a. Then, if the target refinement level of this elememnt is less than one level lower than the maximum target of its edges, then set the target level to be exactly one less than the maximum target level of its edges. Algorithmically, this can be expressed more succintly as: if $L_t < L_a - 1$, then $L_t = L_a - 1$. For example, if for a particular element $L_c = 5$, $L_t = 1$ and the neighbors have targets of 1, 3, 1, 2. Then $L_a = 3$ and we set $L_t = 2$.

For making the refinement levels consistent between siblings, we take a conservative approach and mark an element and its siblings for coarsening only up to the maximum level (smallest size), L_s, allowed by the element and all its siblings. So, if $L_t <= L_s < L_c$ then $L_t = L_c$. For example, if $L_c = 5$, $L_t = 1$ for an element, but $L_s = 3$, i.e., one of the siblings of the element has a target level of 3. Then we cannot coarsen the current element to a level lower than $L_t = 3$. On the other hand if $L_t < L_c < L_s$, then $L_t = L_c$. For example, $L_c = 5$, $L_t = 1$ as before, but $L_s = 7$, i.e., a sibling wants to be refined from the current level while the element wants to be coarsened. Then the element cannot be coarsened above the current level, $L_t = L_c$.

Next the elements are coarsened one level at a time starting from the highest level. Everytime an element and its siblings are deleted we transmit the target refinement level to its parent. After coarsening the mesh at a particular level, we redo the level adjustment before coarsening at the next lower level.

3.4 Refinement

The most important rule imposed during refinement is that *irregular elements are never refined* as their repeated subdivision can lead to unbounded deterioration of quality. Instead, whenever an irregular element is tagged for refinement, the element and its siblings are deleted and its parent element is tagged for uniform refinement upto the maximum level requested by the element and its siblings. This rule ensures that the quality of the refined mesh is always bounded by the quality of the parent mesh. Schneiders defines refinement schemes with this property as being stable [12].

Refinement of the mesh and adjustment of levels before refinement is bit more complex than coarsening because regular and irregular elements have to be dealt with separately. On the other hand, level adjustment has to be done only once before multilevel refinement as opposed to the doing it at each level for coarsening.

To do level adjustment for refinement, we look at each element whose target refinement level L_t is higher than its current refinement level, L_c. Then we get the maximum refinement level, L_a of all its edge connected neighbors. As before, if its target refinement level is one less than maximum target level of the neighboring elements, i.e. $L_t < L_a - 1$, then adjust the target level of this element as $L_t = L_a - 1$. Also, if the element is irregular and one of its edges is to be refined, then mark the element and its siblings for deletion and mark its parent to be refined to L_t. Finally, if two adjacent elements are to be subdivided irregularly, ensure that the common edge of the two is also to be subdivided. This ensures better element quality as shown in Figure 4.

Then we delete the irregular faces and subdivide the remaining faces according to the templates based on the number of edges that are refined. Everytime we refine an element, we mark all its children with the target refinement level. We continue to iterate over the mesh elements until all elements have reached their target level of refinement.

4 Remapping or Solution Transfer

Unlike mesh adaptation for capturing geometry, adaptation to reduce solution error for solving a PDE is tightly coupled with the issue of remapping or transfer of quantities from the base mesh to the adapted mesh. This remapping must be done accurately and in a conservative manner (for example, the densities of the child elements must be assigned such that the total mass of the parent element is conserved). When we coarsen a group of elements, we can just sum up the mass (or energy) of the child elements and assign it to the parent. On the other hand, when we refine an element uniformly, then one can equidistribute the mass over the children (less accurate) or do a linear reconstruction of the density function over the parent element

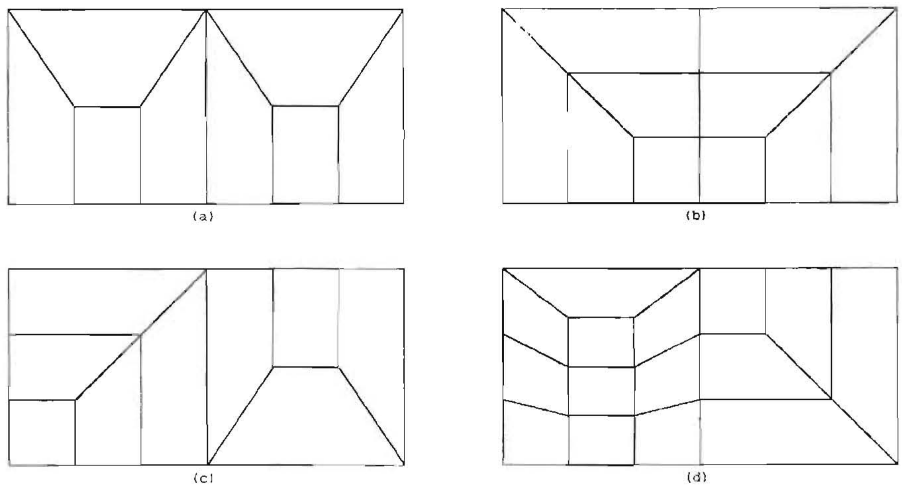


Fig. 4. Refining the common edge between two irregularly refined elements (a) Two adjacent elements with one edge refined (b) same elements with their common edge refined (c) One element with two edges refined next to an element with one element refined (d) same elements with their common edge refined

and integrate over each child to get its mass (more accurate). Field variables (such as velocity) can be obtained by either evaluating an interpolant over the parent at nodes of the child or by solving a local problem over the refined elements using the solution over the base mesh to impose boundary conditions for the local problem. Also, in the proposed algorithm, special care must be taken for remapping when irregular elements are targeted for refinement since the mesh is coarsened back to the parent element and refined down uniformly. Using a summation of masses of the irregular children to get a mass for the parent element and then redistributing it to the regular children can be a poor choice and will lead to lower order accuracy remapping. Rather, it is better to use an intersection based remapping routine locally to get second order accuracy [5].

5 Results

We first present a static example of refinement of a structured mesh mesh adapted to a superimposed line in the mesh. Any element that is intersected by the line is refined up to level 3 (level 0 is the original mesh). The superimposed line goes from $(-0.3082071.106007)$ to $(1.106007 - 0.308207)$. The quality of the mesh before and after the refinement is also compared in terms of the average condition number of the element, $\bar{\kappa}$, defined as the mean of the condition numbers [8, 6] at all corners of the element. One can see from the histogram of the refined mesh that it is not shifted dramatically from the ideal case and that the worst quality element has an average condition number of only 1.69. In fact, in simulations where a line is moved diagonally across the domain and the mesh refined around it, the worst element condition number stays at 1.69. Also, the worst element quality stays at 1.69 regardless of what the refinement level is applied to elements intersected by the line.

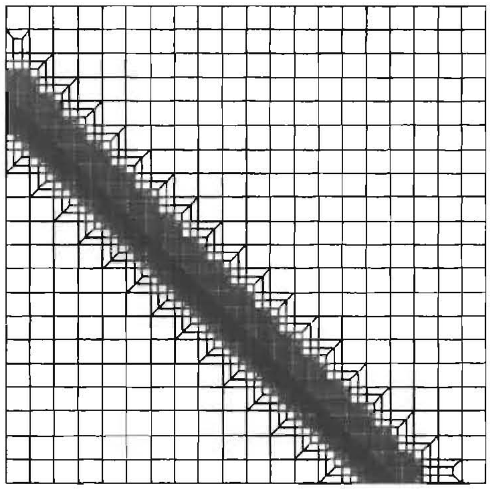


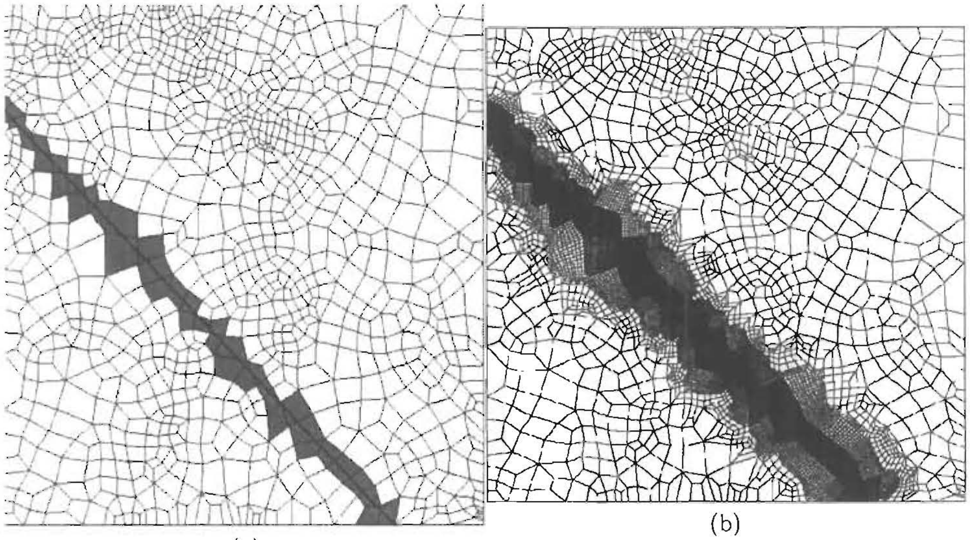
Fig. 5. A 20x20 structured mesh refined using distance from center as the refinement criterion

$\bar{\kappa}$	Original	Refined
1.0 – 1.5	400	12412
1.5 – 2.0	0	1396
2.0 – 3.0	0	0
4.0 – 5.0	0	0
5.0 –	0	0

Next we show refinement induced by the same line in an unstructured quadrilateral mesh. Figure 6a shows the original mesh with the elements marked for refinement to level 3 due to intersection with the line (also shown). Figure 6b shows the refined mesh after the levels have been adjusted to enforce a one-level difference between adjacent elements. Also included is a table showing the distribution of condition numbers before and after refinement. The worst condition number goes from 3.12 to 3.79 after refinement.

In the following example, we show the several snapshots from an dynamic adaptation procedure where a circle of radius 0.1 is moved along a circular path in the domain. The center of the circle traces a circle of radius 0.2 centered at (0.5, 0.5). The starting point of the circle center is (0.7, 0.5). The target size for the elements to be refined is $0.05d$ where d is the distance between the centroid of the element and the center of the circle. As the circle moves, previously refined parts of the mesh are coarsened and new parts are refined with considerable overlap between the coarsened and refined regions. As expected, the worst element quality stays at 1.69 throughout the dynamic adaptation procedure.

Finally, we show several snapshots of a dynamic adaptation procedure in which elements intersecting two expanding circles are refined to level 4. One circle is centered at (0.0, 0.0) and the other circle is centered at (1.0, 0.25). Both circles start with a radius of 0.11 with their radii increasing in increments of 0.05. As the circles grow, they intersect each other and eventually grow out of the domain. Elements



(a) (b)

$\bar{\kappa}$	Original	Refined
1.0 – 1.5	1007	42491
1.5 – 2.0	15	1776
2.0 – 3.0	1	504
3.0 – 4.0	1	15
4.0 – 5.0	0	0
5.0 – 7.5	0	0

(c)

Fig. 6. Refinement of an unstructured mesh along a line (a) Target refinement levels (b) Refined mesh (c) Histograms of condition numbers

that intersect one or the other circle are refined to a level of 3 while elements that intersect both circles are refined to a level of 4. Again the worst quality is stays fixed at 1.69 throughout the adaptation process.

6 Discussion

This paper presented a comprehensive mesh adaptation procedure for quadrilaterals that results in conformal meshes with nested refinement. The refinement is based on templates devised from a consistent 1:3 refinement of element edges. It also presented algorithms for adjustment of refinement levels of elements, both for coarsening and for refinement, such that there is never more than a one level difference between the refinement levels of adjacent elements. The quality of the refined mesh is kept high by never refining irregular elements used to bridge refined and coarse regions of the

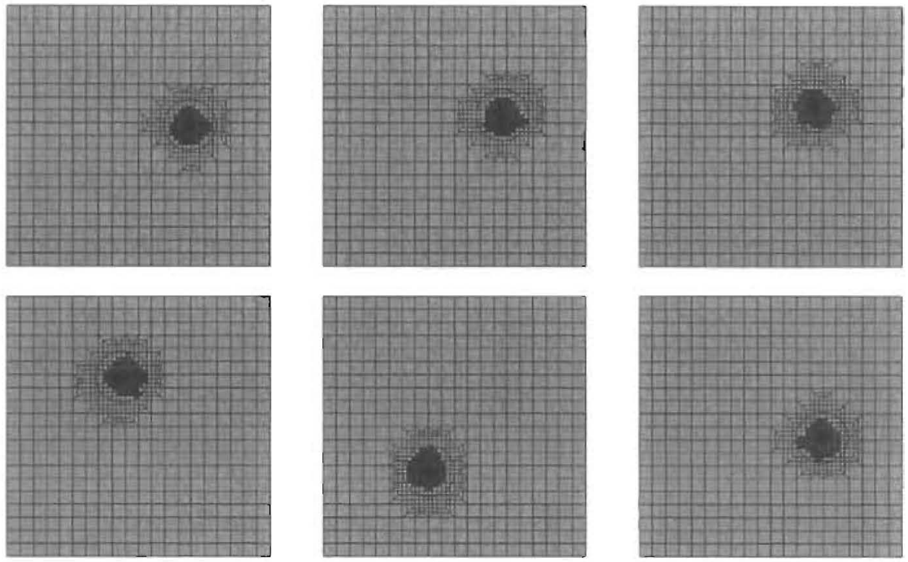


Fig. 7. Snapshots of dynamic mesh adaptation of a 20x20 structured mesh with respect to a circle rotating about the center of the domain

mesh. Instead if irregular elements must be refined, they are deleted and their parent is uniformly refined instead. Using several dynamic mesh adaptation examples, it was shown that the procedure effectively refines the mesh where necessary and coarsens it where it is not.

Although one other paper discussing similar templates and strategy was found after this algorithm was devised, that work is not very well known in meshing circles. Also, that paper does not discuss dynamic mesh adaptation and mesh coarsening explicitly although it also suggests that irregular elements not be refined.

Compared to the algorithm proposed by Schneiders and the templates in his papers, this algorithm produces fewer elements and is simpler due to the consistent use of 1:3 edge refinement. Also, Schneiders does not discuss the issue of mesh quality when forced to refine irregular elements. Finally, the issue of solution transfer or variable refinement is addressed in the current paper which is often ignored in most conformal quadrilateral refinement papers.

In 3D, the combinatorial complexity of the current algorithm could be more complex than that of Schneiders' algorithm. That is because this algorithm tags edges instead of vertices for refinement, thereby resulting in $\sum_{i=0}^{12} {}^{12}C_i = 4096$ possible combinations. Of course, many of these can be eliminated due to symmetry of rotation and inversion. Even so, the number is expected to be higher than in Schneiders' algorithm. Also, it is possible, just like in Schneiders' algorithm, that some subdivisions of the hexahedron faces may not admit a subdivision into hexahedra. In such a case, one can refine additional edges of such hexahedra to be able to mesh

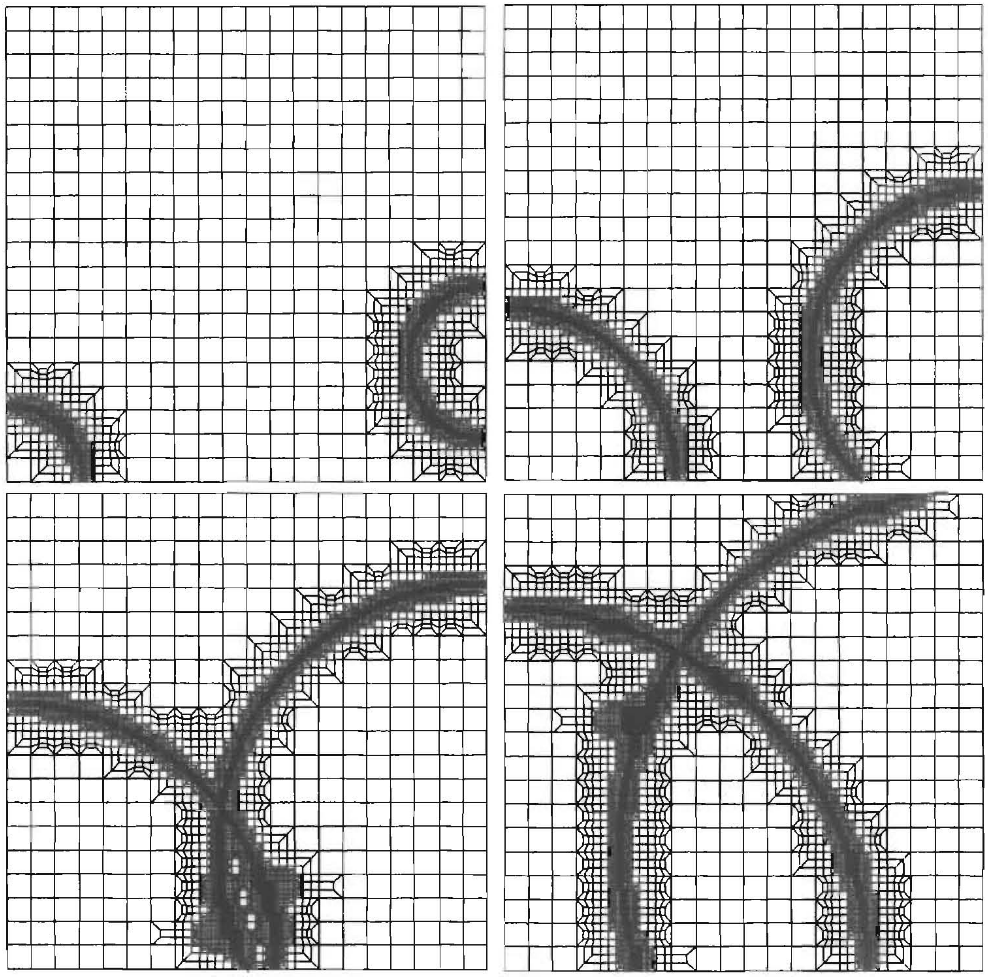


Fig. 8. Snapshots of dynamic mesh adaptation of a 20x20 structured mesh with respect to two expanding circles

them and propagate the refinement further. In such a case, one can only hope that the refinement does not consume the entire mesh.

7 References

References

1. T. Arbogast, L.C. Cowsar, M.F. Wheeler, and I. Yotov. Mixed finite element methods on nonmatching multiblock grids. *SIAM Journal on Numerical Analysis*, 37(4):1295–1315, 2000.
2. J.M. Bass and J.T. Oden. Adaptive finite element methods for a class of evolution problems in viscoplasticity. *International Journal of Engineering Science*, 25(6):623–653, 1987.
3. K.J. Bathe and P.A. Bouzinov. On the constraint function method for contact problems. *Computers and Structures*, 64:1069–1085, 1997.
4. M.E. Botkin, R. Wentorf, B.K. Karamete, and R. Raghupathy. Adaptive refinement of quadrilateral finite element shell meshes.
5. R.V. Garimella, M. Kucharik, and M.J. Shashkov. An efficient linearity and bound preserving conservative interpolation (remapping) on polyhedral meshes. *Computers and Fluids*, 36(2):224–237, 2007.
6. R.V. Garimella, M.J. Shashkov, and P.M. Knupp. Triangular and quadrilateral surface mesh quality optimization using local parametrization. *Computer Methods in Applied Mechanics and Engineering*, 193(9-11):913–928, Mar 2004.
7. Y. Ito, A.M. Shih, and B.K. Soni. Octree-based reasonable quality hexahedral mesh generation using a new set of refinement templates. *International Journal for Numerical Methods in Engineering*, 77(13):1809–1833, 2009.
8. P.M. Knupp. Achieving finite element mesh quality via optimization of the jacobian matrix norm and associated quantitites. Part I - a framework for surface mesh optimization. *International Journal for Numerical Methods in Engineering*, 48(3):401–420, 2000.
9. A. Liu and B. Joe. On the shape of tetrahedra from bisection. *Mathematics of Computation*, 63(207):141–154, July 1994.
10. M.-C. Rivara. Algorithms for refining triangular grids suitable for adaptive and multigrid techniques. *International Journal for Numerical Methods in Engineering*, 20:745–756, 1984.
11. J.S. Sandhu, F.C.M. Menandro, H. Liebowitz, and E.T. Jr Moyer. Hierarchical mesh adaptation of 2d quadrilateral elements. *Engineering Fracture Mechanics*, 50(5/6):727–735, 1995.
12. R Schneiders. Refining quadrilateral and hexahedral element meshes. In *Proceedings of the 5th International Conference on Numerical Grid Generation in Computational Fluid Simulations*, pages 679–688. ississippi State University, 1996.
13. M.S. Shephard. Linear multipoint constraint applied via transformation as part of a direct assembly process. 20:2107–2112, 1984.
14. M. Staten, S. Benzley, and M. Scott. A methodology for quadrilateral finite element mesh coarsening. *Engineering with Computers*, 24(3):241–251, 2008.
15. K.-F. Tchon, J. Dompierre, and R. Camarero. Automated refinement of conformal quadrilateral and hexahedral meshes. *International Journal for Numerical Methods in Engineering*, 59:1539–1562, 2004.
16. Y. Zhang and C. Bajaj. Adaptive and quality quadrilateral/hexahedral meshing from volumetric data. *Computer Methods in Applied Mechanics and Engineering*, 195:942–960, 2006.