POST REFINEMENT ELEMENT SHAPE IMPROVEMENT FOR QUADRILATERAL MESHES

Matthew L. Staten, Scott A. Canann
Ansys, Inc.
275 Technology Drive
Canonsburg, PA 15317
matt.staten@ansys.com, scott.canann@ansys.com

Abstract

Schneiders and Debye (1995) present two algorithms for quadrilateral mesh refinement. These algorithms refine quadrilateral meshes while maintaining mesh conformity. The first algorithm maintains conformity by introducing triangles. The second algorithm maintains conformity without triangles, but requires a larger degree of refinement. Both algorithms introduce nodes with non-optimal valences. Non-optimal valences create acute and obtuse angles, decreasing element quality.

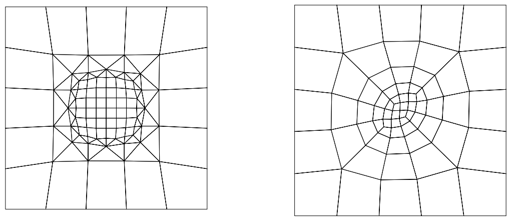
This paper presents techniques for improving the quality of quadrilateral meshes after Schneiders' refinement. Improvement techniques use topology and node valence optimization rather than shape metrics; hence, improvement is computationally inexpensive. Meshes refined and subsequently topologically improved contain no triangles, even though triangles are initially introduced by Schneiders' refinement. Triangle elimination is especially important for linear elements since linear triangles perform poorly. In addition, node valences are optimized, improving element quality.

Introduction

Mesh refinement is the process of taking an existing finite element mesh and changing the size, shape, and/or order of the elements in the mesh in order to increase the accuracy of the finite element solution. Much research has been done on mesh refinement. Rivara (1996) presents a method of refining triangular meshes. Since any arbitrary domain can be completely spanned with triangles, it is trivial to state that Rivara's (1996) refinement results in meshes composed of entirely triangles. The same is not true, however, for quadrilateral meshes. Although refinement of an all quad mesh can result in another all quad mesh, precautions must be taken to ensure this is the case. In addition to ensuring that only quadrilaterals remain in the post refinement mesh, the quality of the quadrilaterals must also be considered.

Schneiders and Debye (1995) present several algorithms for two-dimensional quadrilateral refinement. These algorithms, however, maintain quads at the cost of introducing poor quality quads. On the other hand, one of the algorithms presented by Schneiders and Debye maintains high element quality by introducing triangles.

This paper introduces an extension to Schneiders' quadrilateral refinement in order to eliminate the introduction of triangles and poor quality quadrilaterals (see Figure 1a & b). Element quality is improved by optimization of local topology. Local topology quality can be measured by node valence. Node valence is defined as the number of edges attached to a node. Local topology is improved by eliminating high valence or low valence nodes. The optimal valence of a node in a quadrilateral mesh is four. Schneiders' refinement leaves nodes with valences as high as eight and as low as three. It is impossible for all nodes in a mesh to have a valence of four and still provide transitioning between large and small elements. However, valences can be optimized ensuring that no nodes remain with a valence higher than five or lower than three and that the number of such nodes is minimized. The valence optimization presented is based on the cleanup operations presented by Zhu et al. (1991), Blacker and Stephenson (1990), Canann et. al. (1994), and Canann et al. (1996). Several extensions to those ideas are also presented, including new clean up patterns, experience with clean up patterns, diagonal swapping constraints, and cases where Laplacian smoothing poses problems after cleanup.



(a) Schneiders' 2-Refinement (2-iter.)

(b) 2-Refinement after improvement and smoothing

Figure 1 Schneiders' 2-Refinement before and after cleanup

Previous Research

Schneiders and Debye (1995) present two algorithms for quadrilateral mesh refinement. The first is called 2-refinement because each edge in the refinement region is split into two smaller edges. The second is called 3-refinement because each edge in the refinement region is split into three smaller edges. In both algorithms, a series of nodes are marked where refinement is to be performed. Each quad which is adjacent to any node that was marked is replaced with a transition element template. The only difference between the implementation of 2-refinement and 3-refinement is which templates are used.

Figure 1a illustrates 2-refinement. Numerous triangles have been introduces in the transition region between the refinement region and non-refinement regions. These triangles are undesirable, especially in linear analyses (Zienkiewicz, 1977) and should be eliminated if possible.

Schneiders and Debye (1995) present a variation of 2-refinement which does not introduce triangles. This method works well when refining regular gridded quadrilateral meshes. However, this alternate method is not general enough to be applied to irregular quadrilateral meshes. In addition, this method requires direction information to be stored about the elements in the transition region. The need for this additional information, as well as its inability to be generalized for non-structured meshes, suggests that this is not the desired solution.

Figure 2 illustrates 3-refinement. No triangles have been introduced in the transition region; however, the elements in the transition region are of marginal quality. Node valence can be used to measure element quality. In an optimal quadrilateral mesh, the majority of nodes have a valence of four. With four edges, each of the quadrilaterals adjacent to the node has, on average, an angle of 90º at the node. If a node has a valence greater than four, the average element angle at this node decreases. Likewise, as the valence decreases, the average element angle increases. Node A in Figure 2 has a valence of seven. Element 1 in Figure 2 has an angle of 90º at node A. However, the other six quadrilaterals using node A have an angle of 45º. This is far from optimal. Similarly, node B in Figure 2 has a valence of six. The quadrilaterals using node B have, on average, a face angle of 60º at node B. While this is better than at node A, it is still far from optimal.

Much research with topological cleanup has been done. All cleanup operations can be described as combinations of five basic operations: element opens (Canann et. al., 1994), element closes (Zhu et al., 1991, Blacker and Stephenson 1990, and Canann et. al., 1994), diagonal swaps (Zhu et al., 1991, and Canann et. al., 1994), 2-edge node elimination (Zhu et al., 1991, and Canann et. al., 1994), and 2-edge node insertion (Canann et. al., 1994). Figure 14 illustrates each of these operations.

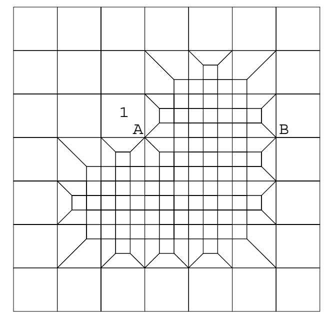


Figure 2 Example of Poor Quads Formed by 3-Refinement

Triangle Elimination

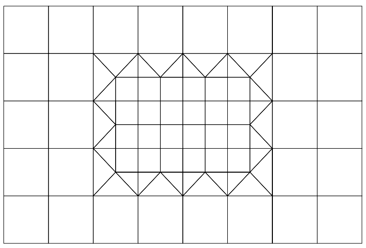
As stated in the previous section, triangles are introduced in Schneiders' 2-Refinement transition templates. Some analyses require all-quad meshes. However, even if the solver allows triangles, triangles typically don't perform as well as quadrilaterals. This is especially true for linear analyses (Zienkiewicz, 1977).

Triangles can be eliminated during 2-refinement by following the steps outlines in Figure 3. Step 1 proceeds, as suggested by Schneiders and Debye, introducing triangles to ensure a conforming mesh. In Step 2, if there are an odd number of triangles in the transition region, a boundary node is inserted to force an even number of triangles. Step 3 combines all triangles into quads without considering the quality of the resulting quads. Step 4 involves performing topological cleanup operations in an attempt to improve node valence. Cleanup is terminated when either no more cleanup operations can be found or a specified maximum number of iterations have been performed. After topological cleanup, smoothing is performed in step 5.

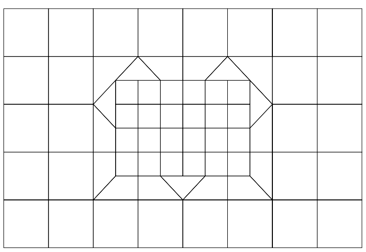
All Quad Refinement
1. Perform Schneiders' 2-refinement.
2. If number of tris is odd, insert a new node on boundary.
3. Combine triangles to make quads, ignoring the quality of the resulting quads.
4. Perform topological cleanup to optimize the valence at nodes.
5. Smooth to improve element shapes.

Figure 3 Steps for All Quad 2-Refinement

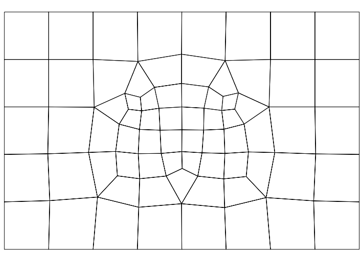
Figure 4 illustrates the refinement of a simple mesh using the algorithm specified in Figure 3. In Figure 4a, the mesh has been refined using 2-refinement. Triangles were introduced in the transition region. Since an even number of triangles were introduced, Step 2 is ignored. Figure 4b shows the same mesh after Step 3 of Figure 3 has been performed. Note that the initial quality of the quads created by combining triangles is poor. Figure 4c shows the same mesh after smoothing. Next, a series of topological cleanup operations are performed according to step 4, followed by smoothing to get the final mesh seen in Figure 4d. The poorly shaped quads have been minimized and no triangles exist in the transition region.



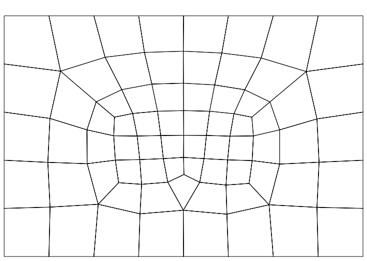
(a)



(b)



(c)



(d)

Figure 4 Improved 2-Refinement

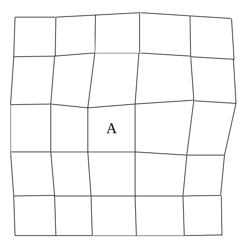
It is not necessary to perform smoothing after triangles are combined before topological cleanup. Smoothing was performed in the example in Figure 4c only to show the quality of elements without topological cleanup. Smoothing is only required after topological cleanup is performed as specified in step 5 of Figure 3.

Triangles are combined in step 3 of Figure 3 using the logic in Figure 5. In Figure 6a, element A is selected for refinement and refined using 2-refinement as shown in Figure 6b. To combine the triangles, triangles that can only be combined with one other triangle are identified as specified in Step 1 of Figure 5. In this example, there are no triangles that fit this criteria, so we continue to Step 2. While processing Step 2, element C in Figure 6b is identified. Element C can be combined with two other triangles (element B or element D). Element B is arbitrarily chosen and combined with element C as shown in Figure 6c. Now, Step 1 is processed again. Step 1 criteria is now satisfied with element D since it can only be combined with element E. These two elements are combined. After this combination, Element F satisfies Step 1 criteria since it can only be combined with element G. These two elements are likewise combined. Step 1 criteria is used to combine the remainder of the elements in a subsequent manner as shown in Figure 6d.

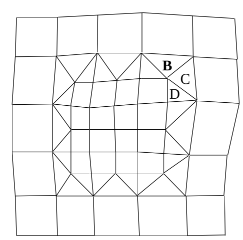
Triangle Combining

1. Find and combine all triangles that can only be combined with one other triangle.
2. If no triangles are identified in Step 1, find a triangle that can be combined with one of two other triangles. Arbitrarily choose one of the possible combinations and combine. Then go back and try Step 1 again.

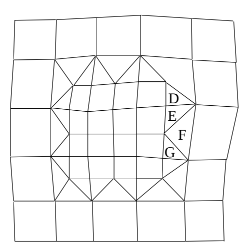
Figure 5 Steps for triangle combining



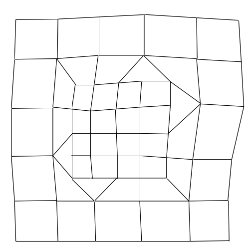
(a)



(b)



(c)



(d)

Figure 6 Triangle combining example

If the transition region is adjacent to a boundary, an odd number of triangles may be introduced by Schneiders' templates. In order to maintain quads in this case, an addition boundary node must be inserted into the mesh as specified in Step 2 of Figure 3. This is illustrated in Figure 7a, where elements A and B have been selected for refinement. Figure 7b shows the mesh before triangle elimination. Note that an odd number of triangles have been introduced. To maintain quads, an additional boundary node is inserted on triangle C making a poorly shaped quad rather than a triangle as shown in Figure 7c. The remainder of the triangles are combined into quads as shown in Figure 7d. At this point, there are no triangles, however, there is one extremely poor boundary quad where the addition node was inserted. To improve the quality of this quad, an element open is performed at node D in Figure 7d to get the mesh in Figure 7e. Figure 7f shows the final mesh after smoothing.

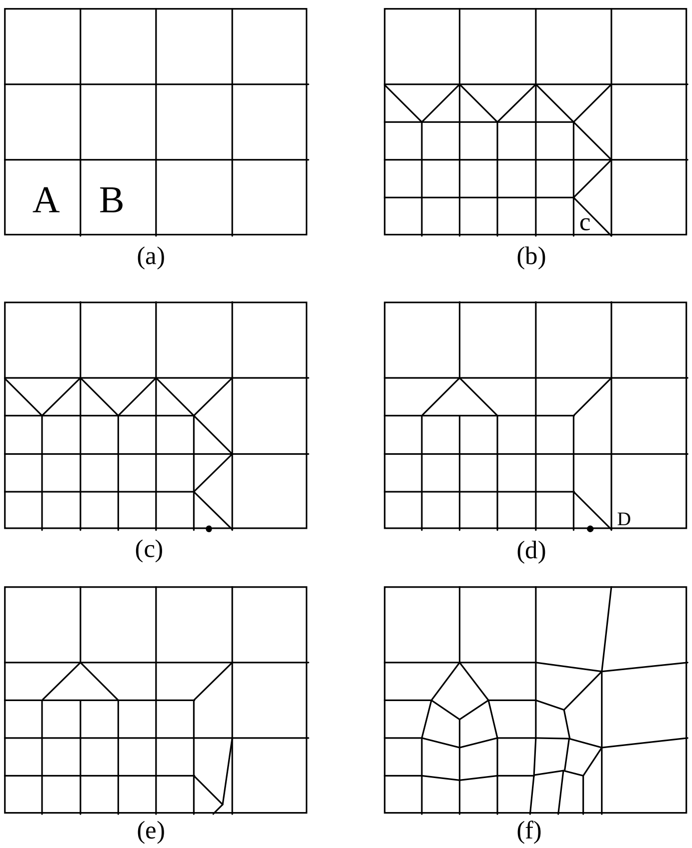


Figure 7 Boundary triangle elimination

The preceding examples of triangle elimination have involved only a single level of refinement. The steps in Figure 3 can be used to clean up refinement regions resulting from multiple levels of refinement. In such cases, there is more than one band of triangles surrounding the refinement region. As a result, there are many more options in how the triangles can be combined into quadrilaterals. Thus, it is generally much more difficult to maintain mesh symmetry in the transition region since each triangle can be combined with numerous other triangles. To maintain symmetry, it is suggested that one level of refinement be performed followed by the combining of triangles. More levels of refinement can then be performed, each followed by an iteration of triangle combining as specified in Figure 3.

In addition, the preceding examples illustrate refinement on a regular gridded mesh. The refinement algorithm presented in Figure 3 and Figure 5 can be used with irregular and unstructured quadrilateral meshes also without any increase in complexity or special cases. Simple gridded meshes were used in this example to simplify for illustration purposes only.

Topological cleanup of 3-Refinement

It has been shown that 2-refinement can be followed by a post processing step to eliminate all triangles from the transition region of all quad meshes. 3-refinement also guarantees that no triangles will be left in the transition region because no triangles are ever introduced. 3-refinement, however, requires a much more dramatic transition, resulting in numerous nodes with far from optimal valences.

In an all quad mesh, the optimal valence at all internal nodes is four. With a valence of four, each quad connected to the node has, on average, an angle of 90º. It is impossible to transition from a small element size to a larger element size while maintaining a valence of four at every node. As a result, some non-optimal valence nodes must be introduced. However, 3-refinement routinely introduces six valent nodes and numerous three valent nodes. Even higher valences can result if the refinement region contains non-optimal valence nodes before refinement is performed. The goal of this section is to decrease the number of nodes with non-optimal valences while maintaining the refinement transition. In addition, the six valent nodes will be decreased to five valent nodes. By decreasing the valence of the nodes from six to five, the average angle in quads connected to the node increases from 60º to 72º, which in turn increases element quality.

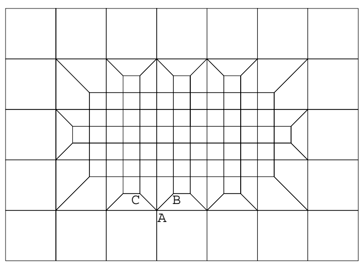
Figure 8 lists the refinement steps for increasing element quality with 3-refinement. Step 1 uses Schneiders' 3-refinement templates to ensure a conforming mesh. In Step 2, the mesh is processed looking for topological cleanup operations that can improve the local topology of the mesh. Many of these cleanup operations are explained by Zhu et al. (1991), Blacker and Stephenson (1990), and Canann et al. (1994). Other new cleanup operations have also been implemented and are illustrated in the appendix. The goal of these topological cleanups is to increase the number of nodes that have a valence of four. These topological cleanup operations are performed regardless of the quality of the resulting elements. Finally, in Step 3, the mesh is smoothed to improve element quality.

All Quad Refinement

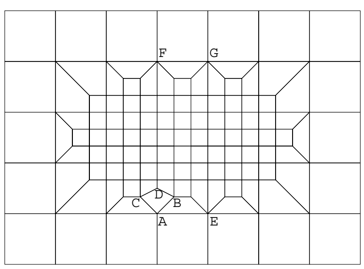
1. Perform Schneiders' 3-refinement.
2. Perform topological cleanup to optimize the valence at nodes.
3. Smooth to improve element shapes.

Figure 8 Steps in 3-Refinement Node Valence Optimization

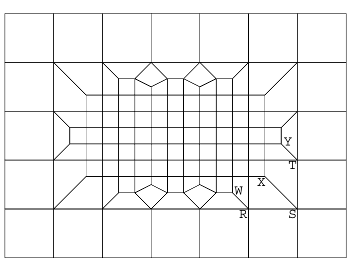
Figure 9 illustrates the refinement of a simple mesh using the steps outlined in Figure 8. Figure 9a shows a simple mesh refined using the Schneiders' 3-refinement templates. Node A has a valence of six and is therefore a target for topological cleanup. In addition, nodes B and C each have a valence of three and are directly adjacent to the six valent node A. The valence of these nodes can be decreased by performing an element open. Figure 9b shows the mesh after an open is performed at node A. Nodes B and C are now optimal each with a valence of 4. Node A has been decreased to a valence of five. A new three valent node was also introduced. This same operation can be performed again on Nodes E, F, and G in Figure 9b. The resulting mesh is shown in Figure 9c



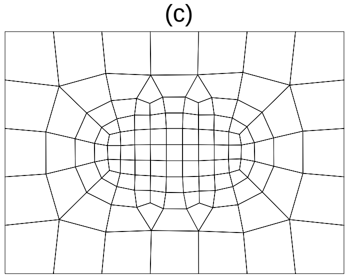
(a)



(b)



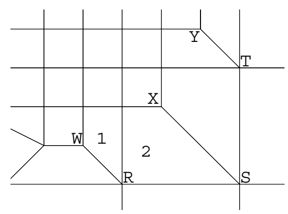
(c)



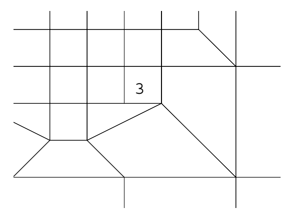
(d)

Figure 9 Topological Cleanup of 3-Refinement

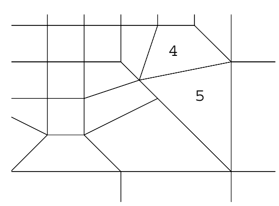
Attention is now turned to the corners of the transition region in Figure 9c. Nodes R, S, and T each have a valence of five. In addition, Nodes W, X, and Y each have a valence of three. Figure 10a shows a close up of these nodes. A multi-step cleanup operation is performed to optimize the valence at these nodes. First the diagonal between elements 1 and 2 is swapped as shown in Figure 10b. Second, an element close is performed on element 3. Finally, the diagonal between elements 4 and 5 is swapped. The result of the multi-step cleanup operation is shown in Figure 10d. This same multi-step cleanup operation can be performed at the other three corners of the transition region in Figure 9c. Figure 9d illustrates the resulting mesh after performing this multi-step operation on each corner followed by smoothing.



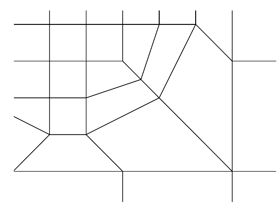
(a)



(b)



(c)



(d)

Figure 10 Topological Corner Cleanup of 3-Refinement

One measure of element quality is a sum of valence variance. The amount that the valence of each node in the mesh varies from four is summed. For example, in Figure 9a, there are 20 three valent nodes, 12 five valent nodes and 4 six valent nodes. The resulting valence variance is 40 (40 = [1*20]+[1*12]+[2*4]). The valence variance of the same mesh after cleanup (Figure 9d) is 16. Since the valence variance is smaller, the average quad angle is closer to 90º, resulting in higher quality elements. Figure 9d allows the same level of refinement while providing a much smoother transition and a much lower valence variance.

Canann's (1994) Order of Irregularity (OI) also decreases after doing topological cleanup. The OI of the mesh in Figure 9a is 48. The OI of the mesh in Figure 9d is 16. Overall mesh OI is better since all 6-edge nodes were eliminated and 5-edge and 3-edge nodes were minimized. Eliminating the 6-edge nodes decreases the OI dramatically since 6-edge nodes are certainly worse than two 5-edge nodes.

Standard element shape metrics can also be used to measure the quality of the elements. For this study, an angle based metric presented by Lee and Lo (1993) was used to measure element quality.

With this metric a perfect quadrilateral with four 90º angles has a metric of 1.0. A quadrilateral on the verge of going concave has a metric of 0.0, and a quadrilateral on the verge of going inverted has a metric of -1.0. The results of this study are summarized in Table 1. Results indicate that the lowest quality element in the refinement without cleanup is considerably worse than the lowest quality element with cleanup. In addition, the number of elements with a metric value less than 0.5 decreases from 27.8 % to 3.5 % when using cleanup operations. Table 1 does indicate that the average metric value decreases slightly. This, however, is insignificant when compared to the dramatic increase in the minimum metric value. Babuska (1976) and Gifford (1979) show that by improving the minimum element quality rather than average element quality, the accuracy of the solution increases.

	Before smoothing and topological cleanup (Figure 10a)	After smoothing without topological cleanup	After smoothing and topological cleanup (Figure 10d).
Ave. Metric	0.828415	0.790809	0.767391
Min. Metric	0.288888	0.383083	0.425812
% < 0.5	27.82	25.21	3.478

Table 1 Results of Element Quality Study

The preceding example used only a single level of refinement. There is no limitation specifying that only a single level of refinement can be performed. Indeed, any number of levels can be performed. After all of the levels have been done, topological cleanup can be performed to improve the quality of all transition elements from each level of refinement.

In addition, the preceding example illustrates refinement on a regular gridded mesh. The refinement algorithm presented in Figure 8 can be used with irregular and unstructured quadrilateral meshes also without any increase in complexity or special cases. Simple gridded meshes were used in this example to simplify for illustration purposes only.

Undoing Refinement

Some care must be taken not to do too much cleanup. Topological cleanup tends to decrease the number of elements. If refinement is performed by marking only one or two nodes followed by topological cleanup, cleanup may decrease the number of elements to what it was before refinement. This is only a problem when a relatively small region is selected for refinement (i.e. a single node or element). In such cases, cleanup operations should be performed carefully and sometimes not at all in order to preserve the refinement.

Speed Considerations

Since the topological cleanup operations are based on topology alone, no costly element quality metrics are computed. Integer comparisons of the valence at adjacent nodes is all that is required to detect areas for cleanup. Boundary nodes require an angle calculation to determine the optimal number of edges; however, this angle can be calculated as a pre-processing step and stored with the boundary node rather than recalculated each time the boundary node is processed. (Canann et al., 1996).

Smoothing

Smoothing is an essential key to successful topological cleanup. The goal of topological cleanup is to optimize nodal valence. Global smoothing is not needed during topological cleanup; however, it is necessary after all cleanup operations have been performed. For example, Figure 4b is the result of some cleanup operations, still, the quality of elements is poor. In addition, element opens and diagonal swaps (Zhu et al., 1991) result in optimal valences, yet, element quality is often poor. Constrained Laplacian smoothing is normally sufficient to improve element quality dramatically.

If cleanup is followed by Laplacian smoothing, care must be taken to ensure that cleanup operations do not invert elements. Laplacian smoothing behaves poorly with inverted elements. Inverted elements can occur when performing most cleanup operations. Figure 11 illustrates creating inverted elements during a swap operation. Diagonal A in Figure 11a is swapped to eliminate two 3-edge nodes and two 5-edge nodes. The resulting elements, however, are inverted as shown in Figure 11b. In cases such as this, a local Laplacian smooth can be performed before the swap is performed (Figure 11c). After local smoothing, the swap does not invert elements.

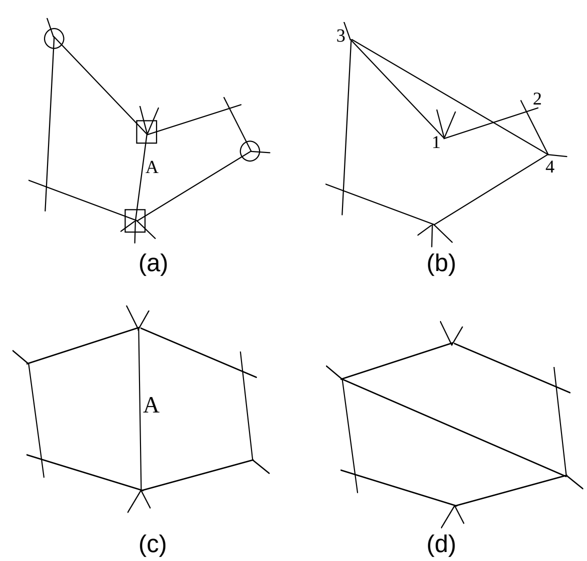


Figure 11 Inverted Elements From Swap

Inversion Constraints

In order to avoid inverted elements as shown in Figure 11, intersection and angle tests can be performed before the swap is performed to ensure that the resulting topology will not be inverted. An intersection test could be performed on the edges connecting nodes 1 and 2 and nodes 3 and 4. This however, would not detect the case illustrated in Figure 12 where diagonal A is swapped for diagonal B. In this case an angle test as shown in Figure 13 can be performed. If either α or β is larger than γ, the element is inverted at node A.

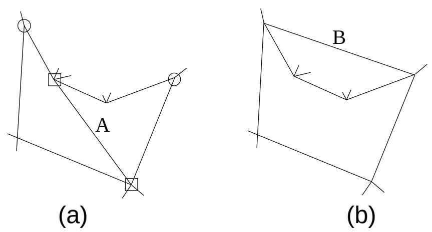


Figure 12 Inverted Elements From Swap

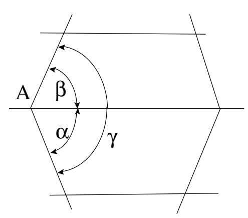


Figure 13 Angle Test for Inversion

Conclusions

High quality all-quad refinement is attainable using Schneider's refinement algorithms. 3-Refinement always yields an all-quad mesh; however, the quality of some elements may be poor. 3-Refinement followed by topological cleanup operations and smoothing results in a high quality all-quad mesh. High and low valence nodes have been minimized and no nodes remain with valences higher than five or lower than three.

2-Refinement results in triangles in the transition region. However, these triangles can be combined into high quality quadrilaterals. It has been shown that all triangles can be eliminated from the transition region. If boundary nodes are involved in the refinement region, an odd number of triangles may be introduced making it necessary to insert an addition boundary node before triangle combining.

The main disadvantage of 2-refinement is the introduction of triangles. However, since these triangles can be easily eliminated, 2-refinement is just as viable as 3-refinement. In addition, 2-refinement provides a much more gradual transition than 3-refinement, indicating that 2-refinement is, in some cases, preferable.

References

Babuska, Aziz, A. K., 1976, "On the Angle Condition in the Finite Element Method," SIAM J. Numer. Anal. Vol. 13, No. 2, pp. 214-227.

Blacker, T. D. and Stephenson, M. B., "Paving: A New Approach to Automated Quadrilateral Mesh Generation," SAND90-0249, Sandia National Laboratories, 1990.

Canann, S. A., Muthukrishnan, S. N., and Phillips, R. K., 1994, "Topological Improvement Procedures for Quadrilateral and Triangular Finite Element Meshes," Presented at the 3rd International Roundtable, Albuquerque, NM.

Canann, S. A., Muthukrishnan, S. N., and Phillips, R. K., 1996, "Topological Refinement Procedures for Triangular Finite Element Meshes," Engineering With Computers, Vol. 12, pp. 243-255.

Gifford, L. N., 1976, "More on Distorted Isoparametric Elements," Int. J. Numer. Methods Eng., Vol. 14, No. 2, pp. 290-291.

Lee, C. K., Lo, S. H., 1993, "A New Scheme for the Generation of a Graded Quadrilateral Mesh," Computers and Structures, Vol. 52, No 5, pp. 847-857.

Rivara, M. C., 1996, "New Mathematical Tools and Techniques for the Refinement and/or Improvement of Unstructured Triangulations," Proceedings, 5th International Meshing Roundtable 96, pp. 77-86.

Schneiders, R., Debye, J., 1995, "Refinement Algorithms for Unstructured Quadrilateral or Brick Element Meshes," Modeling, Mesh Generation and Adaptive Numerical Methods for Partial Differential Equations; Proceedings IMA.

Zhu, J. Z., Zienkeiwicz, O. C, Hinton, E., and Wu, J., 1991, "A New Approach to the Development of Automatic Quadrilateral Mesh Generation," Int. J. Numer. Methods Eng., Vol. 32, pp. 849-866.

Zienkiewicz, O. C., 1977, *The Finite Element Problem*, 3rd Edition. McGraw-Hill, New York.

Appendix

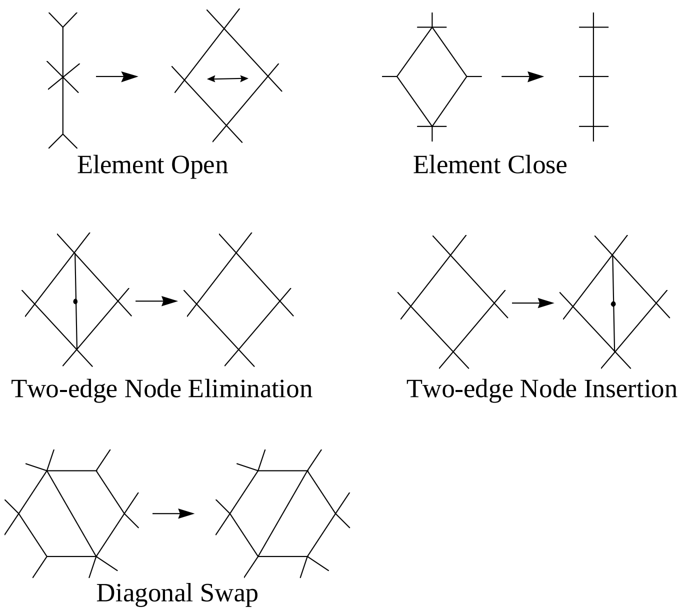


Figure 14 Five basic cleanup operations.

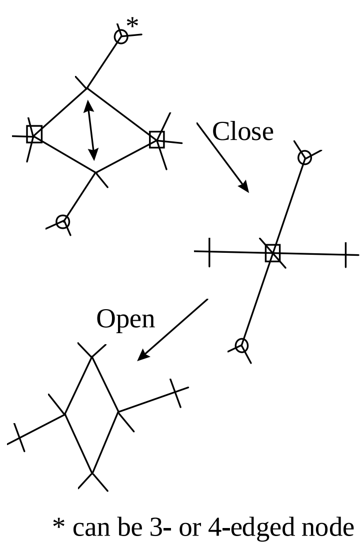


Figure 15 Multi-step cleanup operation

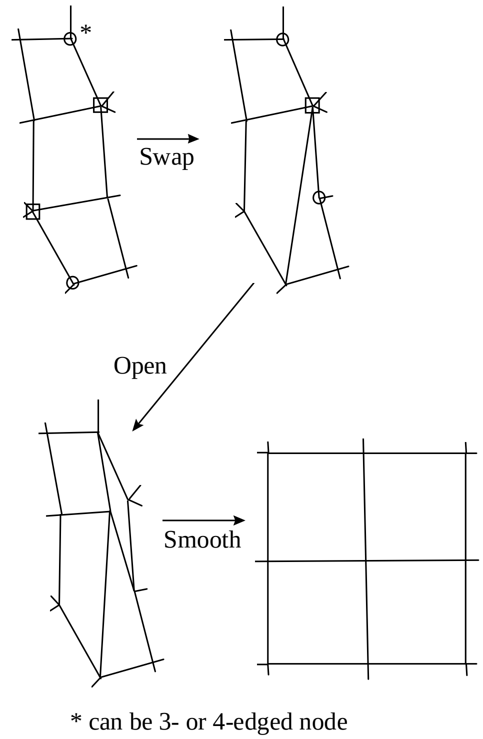


Figure 16 Multi-step cleanup operation

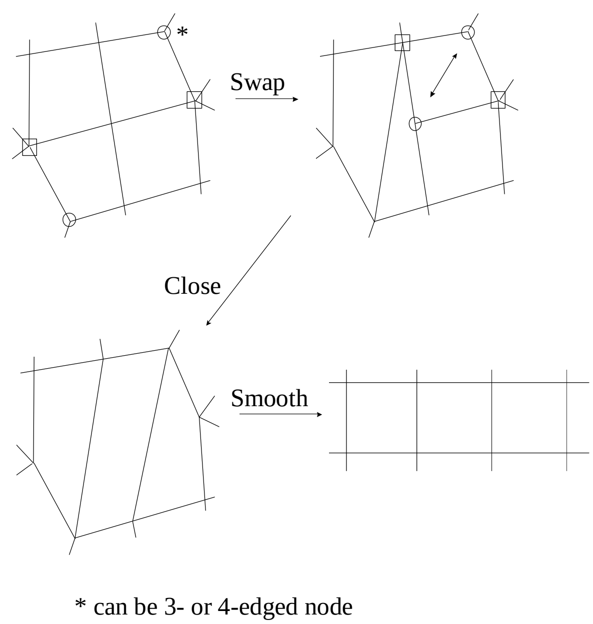


Figure 17 Multi-step cleanup operation