Linear Complexity Hexahedral Mesh Generation

David Eppstein*

Department of Information and Computer Science
University of California, Irvine, CA 92717
http://www.ics.uci.edu/∼eppstein/

Tech. Report 95-51

December 15, 1995

Abstract

We show that any simply connected (but not necessarily convex) polyhedron with an even number of quadrilateral sides can be partitioned into $O(n)$ topological cubes, meeting face to face. The result generalizes to non-simply-connected polyhedra satisfying an additional bipartiteness condition. The same techniques can also be used to reduce the geometric version of the hexahedral mesh generation problem to a finite case analysis amenable to machine solution.

*Work supported in part by NSF grant CCR-9258355 and by matching funds from Xerox Corp.

1 Introduction

There has recently been a great deal of theoretical work on unstructured mesh generation for finite element methods, largely concentrating on triangulations and higher dimensional simplicial complexes [2]. However in the numerical community, where these meshes have been actually used, meshes of quadrilaterals or hexahedra (cuboids) are often preferred due to their numerical properties. For this reason many mesh generation researchers are working on systems for construction of hexahedral meshes. (At the 4th Annual Meshing Roundtable, Sandia, 1995, 13 of the 28 titles on the agenda related to hexahedral meshing.) There has also been some theoretical work on such meshes [7, 8, 9] but much more remains to be done. In particular it remains open whether one can determine in polynomial time whether a polyhedron admits a hexahedral mesh in which all cells are convex. To solve such a problem, one must typically add *Steiner points* interior to the polyhedron, but it is less acceptable to subdivide the polyhedron's boundary, since that would prevent using the meshing algorithm on domains with internal boundaries between different subdomains.

For the planar case, the corresponding problem is easy: a polygon can be subdivided into convex quadrilaterals, meeting face to face, without extra subdivision points on the boundary, if and only if the polygon has an even number of sides. It may difficult to find the smallest number of quadrilaterals needed for this task (or equivalently to optimize the number of Steiner points) but one can efficiently find a set of $O(n)$ Steiner points that suffice for this problem [9].

Thurston [10] and Mitchell [8] recently independently showed a similar characterization for the existence of hexahedral meshes, with some caveats. First, the polyhedron to be meshed has to be simply connected (although the method generalizes to certain polyhedra with holes). And second, the mesh is *topological*: the elements have curved boundaries and are not necessarily convex. However they must still be combinatially equivalent to cubes, and must still meet face to face. Thurston and Mitchell both showed that any simply connected polyhedron has a topological hexahedral mesh, without further boundary subdivision, if and only if there are an even number of faces all of which are quadrilaterals. (Indeed, even parity of the number of faces is a necessary condition for the existance of cubical meshes in any dimension, regardless of the connectivity of the input, since each individual cube has evenly many faces which either contribute to the boundary or are paired up in the interior.)

The method of both Thurston and Mitchell is to treat a hexahedral mesh

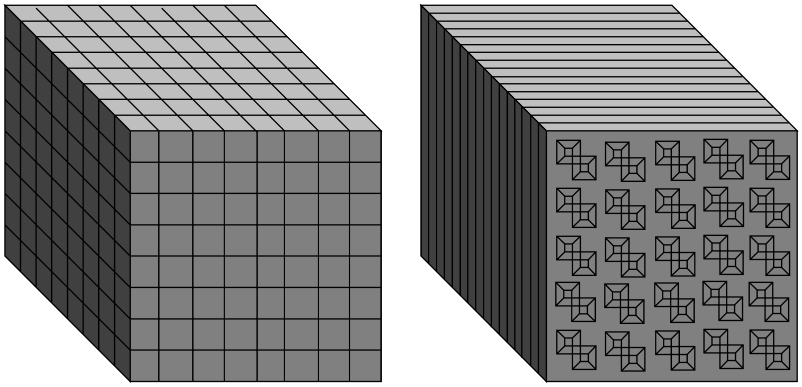

Figure 1. Bad examples for Thurston and Mitchell: (a) $\Omega(n^{3/2})$; (b) $\Omega(n^2)$.

as being the dual to an arrangement of surfaces [7], and a quadrilateral mesh such as the one on the boundary of the polyhedron as being the dual to an arrangement of curves. The problem then becomes one of extending the given surface curve arrangement to an interior surface arrangement, and then fixing up the arrangement locally to satisfy the requirement that cells meet face-to-face. Curves with an even number of self-intersections are extended to surfaces independently of each other, and curves with an odd number of self-intersections are extended to surfaces in pairs.

However this method does not provide much of a guarantee on the *complexity* of the resulting mesh, that is, of the number of hexahedral cells in it. This complexity is very important, as it directly affects the time spent by any numerical method using the mesh; even small constant factors can be critical. It is not hard to provide examples in which this dual surface method constructs meshes with more than linearly many elements (measured in terms of the complexity of the polyhedron boundary); for instance a cube in which each square is subdivided into an $O(\sqrt{n})$ by $O(\sqrt{n})$ grid will end up with a mesh of total complexity $\Omega(n^{3/2})$ (Figure 1(a)). If one incautiously matches odd curves with each other, the complexity can rise even higher, to $\Omega(n^2)$ (Figure 1(b)).

In this paper we discuss an alternate method for hexahedral grid generation, combining refinement of a tetrahedral mesh with some local manipulation near the boundary based on planar graph theory. This technique has three advantages over that of Mitchell and Thurston. First, we prove an $O(n)$ bound on the number of cells needed for a topological hexahedral mesh. Second, because our method avoids duality, it seems easier to extend it to the more practically relevant geometric version of the mesh generation problem. We exhibit a finite collection of polyhedra (formed by subdividing the boundary of a cube) such that if these polyhedra can all be geometrically meshed, any simply connected polyhedron with an even number of quadrilateral sides can also be geometrically meshed, with $O(n^2)$ cells. Third, the method generalizes to a different class of non-simply connected polyhedra than those handled by Mitchell and Thurston's method.

These results are not practical in themselves: the number of elements is too high and we have not satisfactorily completed the solution to the geometric case. Practical hexahedral mesh generation methods are still largely heuristic and will often fail or require the input boundary to be modified. There is a possibility here of a two way interaction between theory and practice: as heuristic mesh generators improve they may soon be good enough to solve the finite number of cases remaining in our geometric mesh generation method, and thereby prove that all even-quadrilateral polyhedra can be meshed. In the other direction, even an impractical proof of the existence of meshes can be helpful, by guaranteeing that an incremental heuristic method such as Mitchell's whisker-weaving idea [7] will not get stuck in a bad configuration.

2 Topological mesh generation

We first describe a method for topological mesh generation in simply connected polyhedra, that is, partitioning the polyhedron into a cell complex in which each cell is combinatorially a cube (but may have curved sides), any two cells intersect in a single face of some smaller dimension (or in the empty set), and each boundary facet of the polyhedron is also a face of some cell. Clearly it is a necessary condition that all polyhedron facets be quadrilaterals. Mitchell [8] and Thurston [10] showed that it is necessary and sufficient that there be an even number of (quadrilateral) facets. We now show that in this problem it is possible to use only a constant factor more cells than the number of facets in the input polyhedron.

Our method has the following main steps:

1. We separate the boundary B of the polyhedron from its interior by a "buffer layer" of cubes. We find a surface S inside the polyhedron, isomorphic to the polyhedron's boundary, and sitting in the same orientation. We then connect corresponding pairs of vertices on the two surfaces with edges, corresponding pairs of edges on the two surfaces with quadrilateral faces spanning pairs of connecting edges, and corresponding pairs of faces on the two surfaces with hexahedra.

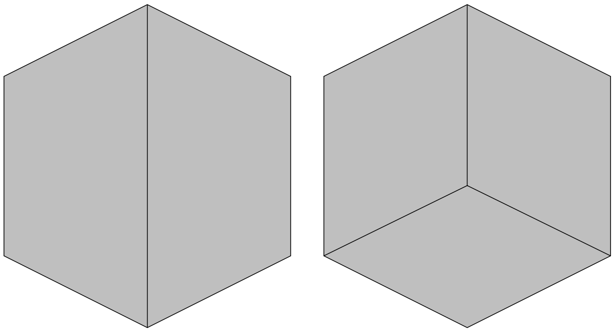

Figure 2. Partition of hexagons into two and three quadrilaterals.

2. We triangulate the inner surface of the buffer layer, and tetrahedralize the region inside this triangulated surface. A tetrahedralization with $O(n)$ complexity can be found by connecting each triangle on S to a common interior vertex.

3. We split each interior tetrahedron into four hexahedra (Figure 3(b)). This subdivision should be done in such a way that any two tetrahedra that meet in a facet or edge are subdivided consistently with each other. As a result, each edge in S becomes subdivided, and each quadrilateral connecting B to S becomes combinatorially a pentagon.

4. Because B is by assumption a planar graph with all faces even, it is bipartite. Let U and V be the two vertex sets of a bipartition of B (without loss of generality, $|U| < |V|$). Each vertex of B corresponds to an edge connecting B to S. We subdivide the subset of those edges corresponding to vertices in U. As a result, each original quadrilateral connecting B to S becomes combinatorially a hexagon.

5. Each of the cells in the buffer layer is now combinatorially a polyhedron with seven quadrilateral facets and four hexagon facets. We subdivide the hexagons into either two or three quadrilaterals each, as shown in Figure 2. We explain below how to do this in such a way that each cell of the buffer layer has an odd number of hexagons subdivided into each type; either one hexagon is subdivided into two quadrilaterals and three hexagons are subdivided into three quadrilaterals each, or three hexagons are subdivided into two quadrilaterals and one hexagon is subdivided into three quadrilaterals.

6. At this point, all the buffer cells are combinatorially polyhedra with

either 16 or 18 boundary facets. If the triangulation of S is chosen carefully (using the same bipartition used above) there will only be two combinatorial types of cell. We partition each cell into a mesh of $O(1)$ hexahedra. (The existence of such a mesh is guaranteed by Mitchell and Thurston's results; alternately it is an amusing exercise to fill out these cases by hand.)

The remaining step that has not been described is how we choose whether to subdivide each hexagon connecting B to S into two or three quadrilaterals, so that each buffer cell has an odd number of subdivided faces of each type. In fact we can do this in such a way as to minimize the total number of diagonals, using a technique familiar from the solution to the Chinese postman problem.

Recall that B is a planar graph, and construct the dual graph B'. Construct a metric on the vertices of B' with distances equal to the lengths of shortest paths in B'. By assumption B has an even number of faces, so there are perfect matchings in this metric; take the minimum weight perfect matching. This corresponds to a collection of paths in B'; any two paths must be edge-disjoint since otherwise one could perform a swap and find a shorter matching. The union of these paths is a subgraph G of B' (actually a forest) in which every vertex has odd degree. Each face connecting two buffer cells corresponds to an edge in B'; subdivide that face in three if it corresponds to an edge in G, and subdivide it in two otherwise.

We summarize the results of this section, without proof.

Theorem 1 *Given any simply connected polyhedron P with an even number n of faces, all quadrilaterals, it is possible to partition P into $O(n)$ topological cubes meeting face-to-face, such that each face of P is a face of some cube.*

3 Geometric mesh generation

We would like to extend the topological mesh generation method described above to the more practically relevant problem of geometric mesh generation (partition into convex polyhedra combinatorially equivalent to cubes). Although our extension seems unlikely to be practical itself, because of its high complexity and the poor shape of the hexahedra it produces, it would be of great interest to complete a proof that all polyhedra (with evenly many quadrilateral faces) can be meshed. Also, it might make sense to include a powerful but impractical theoretical method as part of a more heuristic mesher, to deal with the difficult cases that might sometimes arise.

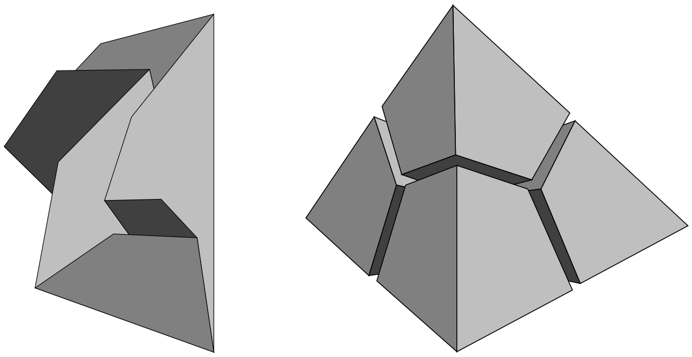

Figure 3. Geometric hexahedralization: (a) a vertex neighborhood which can not be extended by a single layer of cubes; (b) the partition of tetrahedra into hexahedra on planes through edges and opposite midpoints.

In any case, we have made some progress towards a geometrical mesh generation algorithm, but have not solved the entire problem. We have been able to solve the seemingly harder unbounded parts of the problem, leaving only a bounded amount of case analysis to be done. It seems likely that heuristic mesh generation methods may soon be capable of performing this case analysis and finishing the proof.

We go through the steps of our topological mesh generation algorithm, and describe for each step what changes need to be made to perform the analogous step in a geometric setting.

1. Our topological method separates the boundary B of the polyhedron from its interior by a single buffer layer of cubes connecting B to an isomorphic surface S inside the polyhedron. Unfortunately there exist polyhedra for which no isomorphic interior surface can be connected to the boundary; Figure 3(a) shows an example of a vertex surrounded by six quadrilaterals in such a way that, no matter where the corresponding interior vertex is placed, some faces are invisible to it and hence can not be connected by geometric hexahedra. This example is easily completed to a polyhedron with the same property. Instead we form a more complicated buffer layer in the following way. We first cover each face f of B by a cube, with the opposite cube face very close to f and somewhat smaller than f, so that the other four sides of f are "beveled" to be nearly parallel to f. For any two faces f and f' sharing an edge of B, we add two more cubes, both also sharing the same edge, connecting the two cubes attached to f and

f'. The faces of these cubes attached to edges can be classified into three types: two are adjacent to other such cubes or to the cubes on f and f'. Two more are incident to the endpoints of the shared edge and are again beveled to be nearly parallel to that edge. The final two point towards the interior of the polygon. These two faces are very close to parallel to each other, so that the two faces incident to the endpoints of the shared edge have a “kite”-like shape resembling a slightly dented triangle. Finally, we must cover the region near each vertex of B. As seen from the vertex, the faces of the cubes we have already added form a vertex figure that can be represented as a even polygon on the surface of a sphere. We triangulate this polygon and add a small cube corresponding to each triangle, with the three faces incident to the vertex at B corresponding to the edges of the triangle. This determines seven of the eight vertices of each cube; the eighth is then fixed geometrically by the positions of the other seven. Since the three faces incident to the vertex of B are all kite-shaped, the three opposite faces are close to parallel to each other. By making all these cubes attached to B small enough, and by making their faces close enough to parallel, this can all be done in such a way that no two cubes interfere with each other.

2. The second step of our topological method was to triangulate the inner surface of the buffer layer, and tetrahedralize the region inside this triangulated surface. A tetrahedralization with $O(n^2)$ complexity can be found using a method of Bern [1]. (The bound claimed in that paper is $O(n + r^2)$ where r is the number of reflex edges, however our first step creates $\Omega(n)$ reflex edges. Perhaps it is possible to use the information that many of these edges are very close to flat, to reduce the complexity to depend only on the reflex edges of B.)

3. We third step of our topological method was to split each interior tetrahedron into four hexahedra. In order to do this geometrically in a way consistent across adjacent pairs of tetrahedra, we subdivide each tetrahedron using planes through each edge and opposite midpoint (Figure 3(b)). It is not hard to show that these four planes meet in a common point (e.g. by affine transformation from the regular tetrahedron). The subdivision on each tetrahedron face is therefore along lines through each vertex and opposite midpoint.

4. The next step of our topological method was to find a bipartition of B, and subdivide the interior edges incident to one of the two vertex

classes of the bipartition. This step remains unchanged except that each vertex in the given class may be incident to many interior edges; all are subdivided.

5. At this point, the cells of the buffer layer fall into several classes. The cells coming from faces of B are like those of our topological construction, polyhedra with seven quadrilateral facets and four hexagon facets. The cells coming from edges of B have four quadrilaterals, three hexagons, and an octagon. The cells coming from vertices of B on one side of the bipartition have 18 quadrilaterals and three hexagons. The cells coming from the other side of the bipartition have 18 quadrilaterals and three octagons. In any case, the hexagon and octagon sides need to be subdivided, in such a way that all cells end up with an even number of sides. We can use the same idea of matching here; in fact the cells at each vertex can be matched independently, leaving one larger matching connecting the cells on faces and edges.

6. Finally, each buffer cell needs to be meshed. This can be done independently for each cell, but it would require a case analysis (which we have not done) to show that each possible type cell can be meshed.

Thus of the steps in our topological mesh generation procedure, it is only the final finite case analysis which we have been unable to extend to the geometric problem.

4 Generalizations

The only important property we used of simply connected polyhedra (with quadrilateral faces) is that their boundaries form bipartite graphs; but the same extends to simply connected domains with cubically meshed surfaces in any dimension, as can easily be seen via homology theory. (Hetyai [5] has an alternate proof of bipartiteness for shellable complexes.) Thus there seems no conceptual obstacle to extending this technique to higher dimensional meshing problems, although it again requires a case analysis or other technique such as that of Thurston and Mitchell to prove that the resulting buffer cells are meshable.

An alternate direction for generalization is to non-simply-connected polyhedra in three dimensions. Mitchell [8] describes a generalization of his method which applies whenever the input polyhedron forms a handlebody that can be cut along evenly-many-sided disks to reduce its complexity.

(Clearly, such a simplification can be used independently of the mesh generation method to be used.) Our method can handle an alternate class of polyhedra, such as knot complements or bodies with disconnected boundaries, for which no simplifying disk cut exists. The only step where we used the connectivity of the input boundary was in the result that a planar graph with even faces is bipartite; instead we can simply require that the input polyhedron be bipartite with evenly many sides. We can topologically mesh any such polyhedron; alternately, if we could solve the same finite set of cases as before we can geometrically mesh any such polyhedron. (The geometric case needs an extension of Bern's surface-preserving tetrahedralization to non-simply-connected polyhedra, due to Chazelle and Shouraboura [3].)

References

[1] M. Bern. Compatible tetrahedralizations. 9th ACM Symp. Comp. Geom. (1993) 281–288.

[2] M. Bern and D. Eppstein. Mesh generation and optimal triangulation. *Computing in Euclidean Geometry*, 2nd ed., World Scientific (1995) 47–123.

[3] B. Chazelle and N. Shouraboura. Bounds on the size of tetrahedralizations. 10th ACM Symp. Comp. Geom. (1994) 231–239.

[4] E. Heighway. A mesh generator for automatically subdividing irregular polygons into quadrilaterals. IEEE Trans. Magnetics 19, 6 (1983) 2535–2538.

[5] G. Hetyai. On the Stanley ring of a cubical complex. *Disc. Comp. Geom.* 14 (1995) 305–330.

[6] E. F. Johnston, J. M. Sullivan, and A. Kwasnik. Automatic conversion of triangular finite meshes to quadrilateral elements. Int. J. Numerical Methods in Engineering 31 (1991) 67–84.

[7] S. Mitchell. Hexahedral mesh generation via the dual. 11th ACM Symp. Comp. Geom. (1995) C4–C5.

[8] S. Mitchell. A characterization of the quadrilateral meshes of a surface which admit a compatible hexahedral mesh of the enclosed volume, 5th MSI Worksh. Computational Geometry, 1995. Available online at
http://ams.sunysb.edu/∼held/usb_comp_geo-95/samitch.ps

[9] S. Ramaswami, P. Ramos, and G. Toussaint. Converting triangulations to quadrangulations. 7th Canad. Conf. Comp. Geom., 1995. Available online at http://www.uqac.uquebec.ca/DIM/prof/jmrobert/Proceedings/godfried.ps.gz

[10] W. Thurston. Hexahedral decomposition of polyhedra. Posting to sci.math, 25 Oct 1993. Available online at http://www.ics.uci.edu/∼eppstein/junkyard/Thurston-hexahedra