THE COMPLEXITY OF MINIMUM CONVEX NESTED POLYHEDRA [1]

GAUTAM DAS – University of Wisconsin

DEBORAH JOSEPH – University of Wisconsin

1. INTRODUCTION.

In this paper we consider the following three dimensional problem:

> Given two concentric convex polyhedra, find a convex polyhedron that nests between them, which has the minimum number of faces.

We show that the problem is NP-hard.

This problem was posed by Victor Klee for arbitrary dimensional polytopes [O], and a polynomial time algorithm has been found for the two dimensional problem [AB]. Since the description of a polyhedron involves vertices, edges and faces, it is natural to look for a polyhedron that minimizes any of these quantities. Using a proof similar to the one that we give for the problem of minimizing faces one can show that the problems of minimizing vertices, and minimizing edges, are also intractable. ¿From a theoretical standpoint, this is one of the few intractable results in three dimensional computational geometry that involves the *simplest* (and *fewest*) objects, namely three convex polyhedra. This problem has obvious practical applications in the areas of *object approximations*, *stock cutting*, and *robotics*. In addition, it arises in the problem of minimizing stochastic automata, [K].

In two dimensions, the problem of minimizing vertices has polynomial time solutions, even for the case of nonconvex polygons [AB]. In three dimensions, we recently showed that minimizing vertices is NP-hard for nonconvex polyhedra [DJ]. However, the reduction generated extremely nonconvex polyhedra. Also, a critical assumption, that surfaces of the polyhedra are of *genus* 0 was used. In this paper we make no such assumptions and our polyhedra are convex.

Before proving our result, we introduce the following preliminaries.

2. PRELIMINARIES.

Let P name a polyhedron. The surface of P is a collection of *faces*, denoted $Faces(P)$. Faces are adjacent at *edges*, and edges are adjacent at *vertices*. The decision version of the *Minimum Nested Convex Polyhedra* problem (abbreviated *MCNP*) is:

> Given a pair of concentric convex polyhedra Q contained in P, (with all vertices having integer co-ordinates), and an integer K, is it possible to construct a convex polyhedron R, with at most K faces, that encloses Q and is contained in P?

For proving NP-hardness, the reduction will be from *Planar-3SAT* [L], which is defined as follows. A *variable-clause* graph of a *3SAT* instance (with n variables and m clauses) is a bipartite graph where vertices are variables and clauses, and edges are between variable vertices and clause vertices, with the rule that if a clause C contains a literal of a variable V, then $[C, V]$ is an edge. Also, an edge is marked $+$ or $-$, depending on whether the variable occurs positively or negatively in the clause. *Planar-3SAT* may be formally stated as: Given a *3SAT* formula expressed by a planar variable-clause graph, is there a satisfying truth assignment for the formula?

We now describe two operations that can be used to alter the surface of a polyhedron. The first operation is known as *face removal*. Consider a convex polyhedron P whose faces are triangles. Let $f = ABC$ be a face, and let the three faces adjacent to it be ABD, ACE, and BCF, as in Figure 1. Let the three adjacent faces, when extended, meet at a point O above ABC. The tetrahedron $ABCO$ is denoted as *Pyramid(f)*. We will assume that our polyhedron is shaped such that, for each face, the polyhedron lies to one side of the plane defined by the face, and the pyramid lies on the other

[1] This work was supported in part by the National Science Foundation under NSF PYI grant DCR-8402375.
Authors' address: Computer Sciences Department, University of Wisconsin, 1210 West Dayton St., Madison, WI 53706, USA.

side. The removal process creates a new polyhedron $P \cup ABCO$, by removing the face ABC and extending the adjacent faces. (Note that the extended faces, $ADBO$, $AECO$, and $BFCO$ are no longer triangles). Clearly, the new polyhedron is convex and has one face less. In general, if we remove k mutually nonadjacent faces, the resultant polyhedron is still convex, and has k faces less.

The second operation is known as a ***plane cut.*** Let P be a convex polyhedron, and p an infinite two dimensional plane that cuts through P, dividing it into two smaller polyhedra, P_1 and P_2. Depending upon the context, we will be interested in only one of these smaller polyhedra. Some faces of P may be split by the plane, with one piece belonging to P_1, and the other piece belonging to P_2. In general, $|Faces(P_1)| \leq |Faces(P)| + 1$, and a similar result holds for P_2. However, if P_2 has k unsplit faces that originally belonged to P, then $|Faces(P_1)| = |Faces(P)| + 1 - k$. A similar result holds for P_2.

Additional definitions will be introduced later. The following section describes the proof of our result.

3. NP-COMPLETENESS RESULTS.

In this section we prove the following theorem.

Theorem 1. *Minimum Convex Nested Polyhedra* problem is NP-hard.

To prove NP-hardness, we will use a reduction from *Planar-3SAT*. We will first consider a variation of the *MCNP* problem where the inner polyhedron Q is convex, but the outer polyhedron P may be nonconvex. Once we prove that this problem is NP-hard, we will show how to modify the outer polygon to generate a convex polyhedron.

Intuitively, the reduction proceeds as follows. We construct the inner polyhedron Q by starting with a sphere and gradually converting it into a polyhedron with triangular faces. Given a planar variable-clause graph, imagine it laid out in a small area on top of a large sphere, as Figure 2 shows. We design components for variables, clauses, and edges of the variable-clause graph. Each component will in turn be a collection of triangular faces of Q. All components (that is, all triangular faces) will be "superimposed" upon the planar drawing of the graph, such that each face is cut out at an appropriate place on the sphere. Once this has been done, the remaining portions of the sphere may be cut out into triangular faces in an arbitrary manner that maintains convexity. The outer (nonconvex) polyhedron P will be designed to act as an "obstacle" near various portions of Q. To make this clear, the construction will be such that the final set $Faces(Q)$ can be partitioned into two sets, ***Fixed(Q)*** and ***Removable(Q)***. A face belonging to the former cannot be removed, because a nearby face of P will be coplanar. A face belonging to the latter can be removed, because three faces of P will be respectively coplanar to the top three faces of its pyramid. In other words, the region occupied by P will be $Q \cup_{f \in Removable(Q)} Pyramid(f)$. If two adjacent faces of Q are ***removable*** faces, it is easy to see that P will be nonconvex.

If the only polyhedra which nest between P and Q are those that can be constructed by the removal of a set of mutually nonadjacent faces of Q, the problem would be discrete, and proving NP-hardness would be reasonably simple. But the actual R's can have faces that have arbitrary orientations, and need not be either coplanar with faces of Q, nor with any upper faces of the pyramids. However, it turns out that there exists an R with ***minimum faces*** that is of the former "discrete" type.

To show that such polyhedra exist we introduce the concept of ***tight*** polyhedra. Given the above construction of P and Q, a concentric polyhedron R is ***tight*** if it can be produced by the removal of a set of mutually nonadjacent faces of Q. We claim that there exists an R with minimum faces that is ***tight.*** To see this, consider any minimum faced R that is not ***tight.*** Then there exists a ***removable*** face f of Q such that the portion of R covering f is

(a) not completely coplanar to f, and
(b) not the upper surface of ***Pyramid(f)***.

In Figure 3, $f = ABC$, the three adjacent faces are ABD, ACE and BCF, and the apex of ***Pyramid(f)*** is O. We locally modify R to form a ***tight*** polyhedron R' with an equivalent number of faces. We first cut R with the plane defined by $ADBO$, and retain the polyhedron that contains Q. We further cut this polyhedron with the planes defined by $AECO$ and $BFCO$, and finally by ABC. This results in a polyhedron that has ABC as a face, and the faces adjacent to its sides are respectively coplanar to the original faces of Q that were adjacent to f. Finally, we remove ABC from this polyhedron by extending adjacent faces, which results in the polyhedron R'. Using some geometry, as well as properties of the cut operation, it can be shown that $|Faces(R)| = |Faces(R')|$.

This process can be repeated for all faces of Q until we obtain a *tight* R. Clearly, the tightening process takes polynomial time. Thus, given such a pair of P and Q, and a minimum faced R, we can compute a *tight* R with the same number of faces in polynomial time.

We now only have to prove the NP-hardness of computing a *tight* minimum faced R. In the following paragraphs the components of the reduction are described in more detail.

Variable Components. There is one variable component per variable and the design of each component is dependent upon the total number of clauses, m. In particular, each component consists of $4m$ *removable* triangular faces, arranged in a cycle, as Figure 4 shows. Adjacent faces can be paired such that the base edges of both faces in a pair are either on the outer boundary, or on the inner boundary of the cycle. In each outer pair, the clockwise face is called a *positive* face, while the counterclockwise face is called a *negative* face. Thus, there are m *positive* and m *negative* outer faces. The inner pairs simply exist to connect the component into a cycle. We superimpose each variable component as a "macro vertex" in the variable-clause graph on the sphere, with the faces cut out such that convexity is maintained. Later, other triangular faces will be laid adjacent to each edge along the outer and inner boundary of the cycle.

In order to understand the purpose of the variable components, assume that such additional faces have indeed been laid out, and that they are all *fixed* faces. Consider the portion of any *tight* minimum faced R covering the faces of this component. We can imagine R being created by the process of face removals that is caused by the growth of adjacent faces. Suppose a face has been removed. Clearly, neither of its two adjacent faces in the cycle can be removed. Thus, at most we can remove the positive faces together with some of the connecting faces, or we can remove the negative faces together with some of the connecting faces. Our reduction equates removing the negative faces (and leaving the positive faces) with setting the variable to *true*, and equates removing the positive faces (and leaving the negative faces) with setting the variable to *false*.

This geometric problem has a graph-theoretic analog. If we define the *face-graph* of a component such that *removable* faces correspond to vertices, and edges are between vertices that correspond to adjacent faces, then computing a best R for the component is equivalent to finding a maximum independent set in its face-graph. •

Clause Components. The clause components are shown in Figure 5. Each component has 9 *removable* faces together with a *fixed* central face. Three of the *removable* faces are called *literal* faces. The *literal* faces are connected by pairs of *connecting* faces, see Figure 5. As with the variable components, we superimpose each clause component on the sphere as a "macro vertex" in the variable-clause graph, taking care to maintain convexity while cutting out faces. Later we will lay out other triangular faces adjacent to each edge on the outer boundary of the clause component cycles. To understand the role of the clause components in the reduction assume that such faces have been laid out, and that they are all *fixed* faces. We can see that in the best case, any *tight* minimum faced R can remove a maximum of four faces from the cycle; three connecting faces and one literal face. We will see that this is possible if and on hly if the clause in the 3SAT formula is satisfiable.

Again, the problem of maximally removing faces is equivalent to the graph theoretic problem of finding a maximum independent set in the face-graph of this component. •

Edge Components. Edge components provide a means of linking together the variable and clause components. We will illustrate this with an example. Consider an edge of the variable-clause graph. (Components for "+" and "-" edges will be the same.) The corresponding edge component is realized as shown in Figure 6. It has an even number of faces adjacent in a chain. (The actual number of faces is not particularly important and is determined by the geometry of the planar embedding of the variable-clause graph.) We superimpose the chain on the edge in the variable-clause graph such that one end face is adjacent to a face in the variable component, and the other is adjacent to the appropriate *literal* face in the clause component. If the edge is a "+" edge, then it is attached to a positive face in the variable component, and if it is a "-" edge it is attached to a negative face in the variable component. There are enough faces in each variable component to accommodate edge components originating from all clause components. The planarity of the variable-clause graph ensures that edge components can be laid out without encountering crossover situations. As before, in laying out the edge components convexity must be maintained.

To understand the role of the edge components we once again assume that *fixed* faces are laid out adjacent to each edge on the boundary of the chain. Since an edge component has an even number of faces, the best *tight* R for an edge component has to select exactly one of the end faces for removal. However, the face that is adjacent to it, in either the variable or the clause component, can not be removed. To understand the importance of this with respect to the reduction, consider a variable that occurs positively in a clause and the corresponding variable, edge and clause components constructed in the reduction. A truth assignment that sets the variable to "true" will sastisfy the clause. In our reduction setting the variable to "true" corresponds to removing the negative faces (and leaving the positive faces) in the variable component. If the positive faces remain, then the edge face adjacent to the positive variable face can be removed. But, this will means that the edge face at the clause component end of the edge component can not be removed. However, if the edge face adjacent to the clause component is not removed, then the adjacent literal face can be removed. Thus, if we leave the positive variable faces we can remove a literal face in the corresponding clause component. •

Before we complete the construction note that since the edge components connect all variable and clause components, computing the best R is equivalent to finding a maximum independent set in the global face-graph.

Finally, to complete the reduction, we identify all faces that do not yet have three adjacent faces, and lay out *fixed* faces alongside them. We then triangulate the remaining portion of the sphere, taking care to maintain convexity. These additional faces are *fixed*. Finally, we design P such that

$$P = Q \bigcup_{f \in Removable(Q)} Pyramid(f).$$

It remains to find an integer K such that the original $3SAT$ formula is satisfiable *if and only if* there is a nested polyhedron between Q and P with K, or fewer, vertices. To do so we let k be the number of faces of Q, and let $2r$ be the number of faces in all edge components. Then, $K = k - 2nm - 4m - r$. Notice that the second term corresponds to the reduction in faces if all variable components are optimally covered. Similarly, the third and fourth terms correspond to optimal coverings for the clause and edge components, respectively.

Thus, constructing a minimum faced R nested between a nonconvex P and a convex Q is NP-hard.

It remains to show that the problem is NP-hard even when the exterior polyhedron is convex. Suppose that we replace P by its convex hull P', and we ask the question: Is there an R between P' and Q such that the number of faces of R is at most K? Since there is more space between P' and Q we might be able to find an R between P' and Q that has fewer faces than any R that fits between P and Q. However, this is not so, because as we show in the full version of this paper, any minimum faced R in this more relaxed environment can be converted (in polynomial time) into a *tight* R without increasing the number of faces, and any *tight* R between P' and Q obviously fits between P and Q. This in turn shows that the construction of P' suffices for proving NP-hardness of our original problem.

4. REFERENCES.

[AB] A. Aggarwal, H. Booth J. O'Rourke, S. Suri, C.K. Yap, "Finding Minimal Convex Nested Polygons", Proc. 1^{st} Annual Symp. on Comp. Geometry (1985), pp 296-303.

[CY] J.S. Chang, C.Y. Yap, "A Polynomial Solution for Potato Peeling and Other Polygon Inclusion and Enclosure Problems", IEEE Symp. on Foundations of Computer Sciences, (1984), pp 408-417.

[DJ] G. Das, D. Joseph, "Minimum Vertex Hulls for Polyhedral Domains", Proc. 7^{th} STACS (1990), to appear in LNCS, Springer-Verlag.

[K] V. Klee, Private communication.

[KL] V. Klee, M.C. Laskowski, "Finding the Smallest Triangle Containing a Given Convex Polygon", Journal of Algorithms, (1985).

[L] D. Lichtenstein, "Planar Formulae and their Uses", SIAM Journal Comp., (1982), pp 329-343.

[O] J. O'Rourke, "Computational Geometry Column", SIGACT News, (1988).

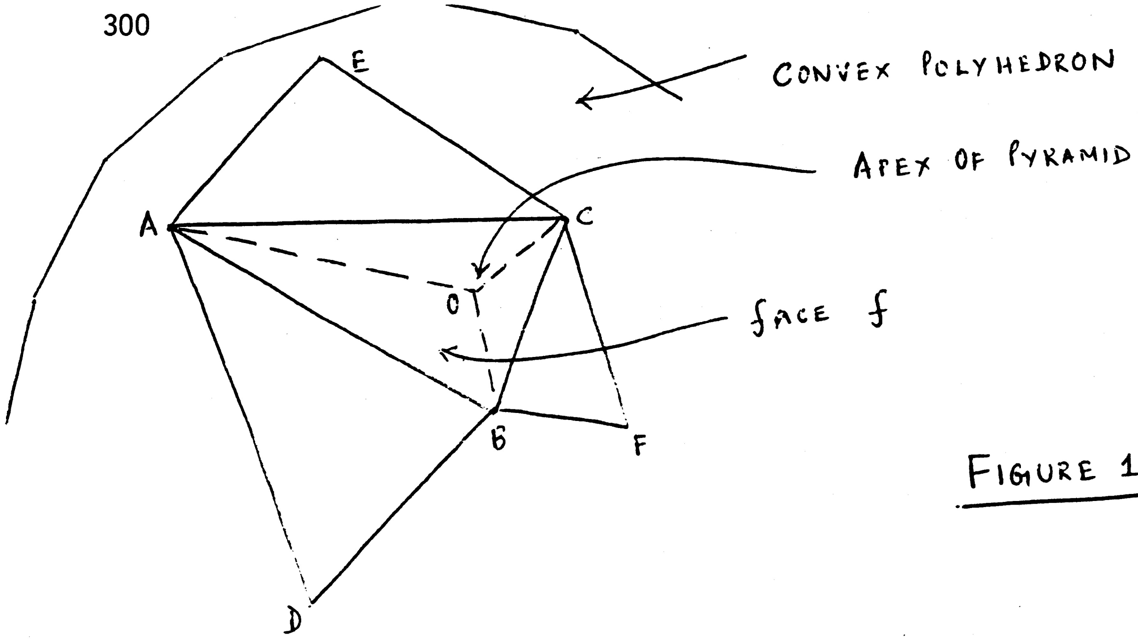


FIGURE 1.

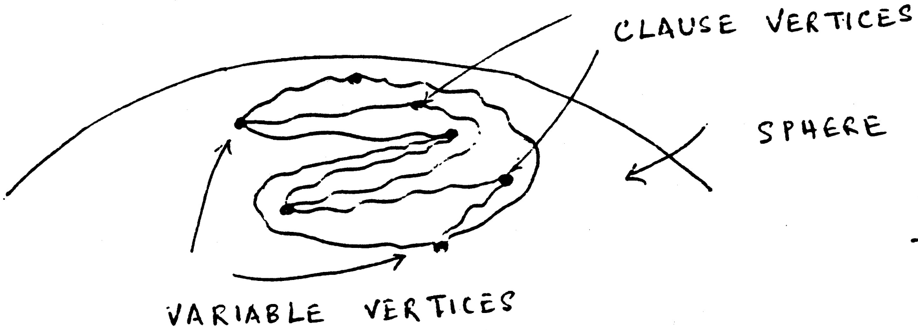


FIGURE 2.

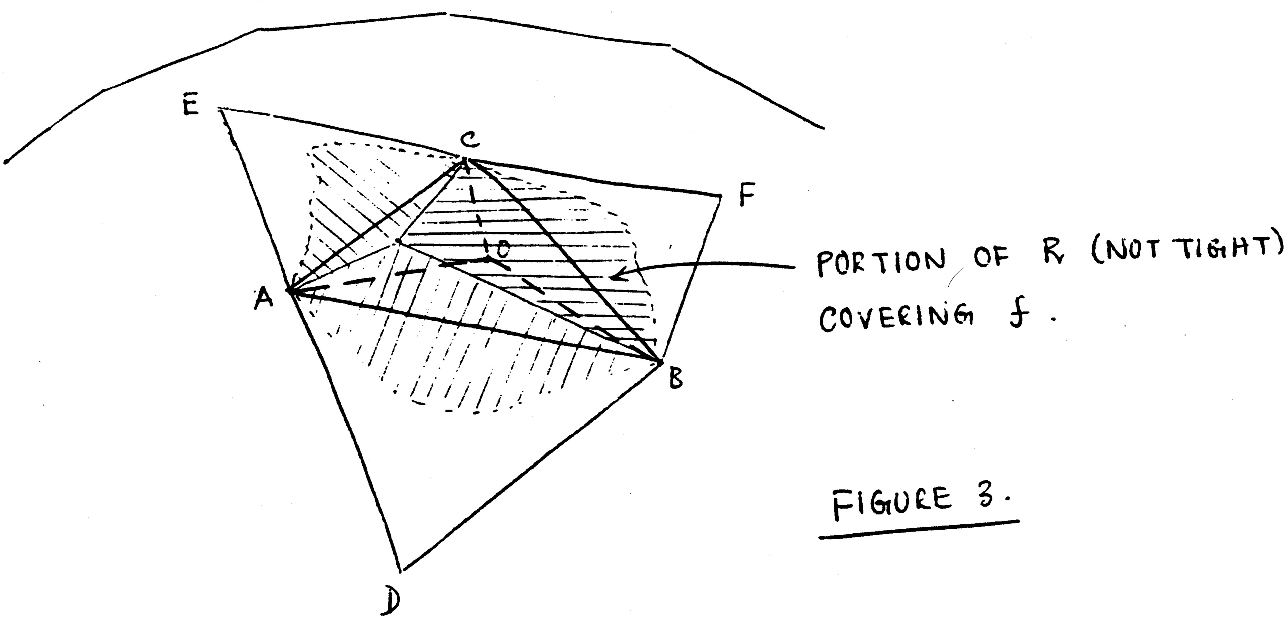
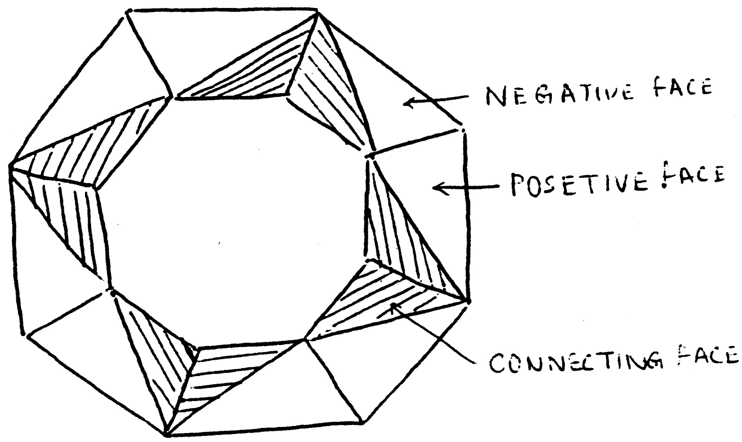
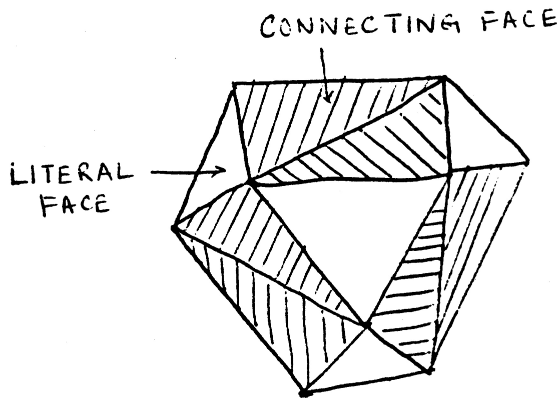


FIGURE 3.



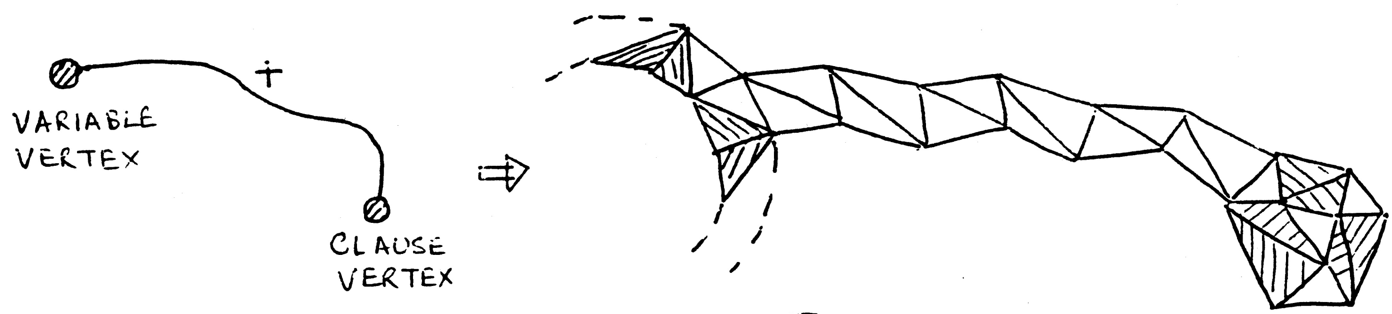
VARIABLE COMPONENT

FIGURE 4.



CLAUSE COMPONENT

FIGURE 5.



EDGE COMPONENT

FIGURE 6.