Approximation for Minimum Triangulations of Convex 3-Polytopes

Francis Y. L. Chin *

Department of Computer Science and Information Systems,

The University of Hong Kong, Hong Kong.

chin@csis.hku.hk

Stanley P. Y. Fung

Department of Computer Science and Information Systems,

The University of Hong Kong, Hong Kong.

pyfung@csis.hku.hk

Cao-An Wang †

Department of Computer Science,

Memorial University of Newfoundland, Canada.

wang@cs.mun.ca

Abstract

A minimum triangulation of a convex 3-polytope is a triangulation that contains the minimum number of tetrahedra over all its possi-

*This work is supported by RGC grant HKU 7019/00E.

†This work is supported by NSERC grant OPG0041629.

ble triangulations. Since finding minimum triangulations of convex 3-polytopes was recently shown to be NP-hard, it becomes significant to find algorithms that give good approximation. In this paper, we give a new triangulation algorithm with an improved approximation ratio $2 - \Omega(\frac{1}{\sqrt{n}})$ where n is the number of vertices of the polytope. We further show that this is best possible for algorithms that only consider the combinatorial structure of the polytopes.

1 Introduction

Triangulation is the subdivision of d-dimensional polytope into simplices, such that these simplices only intersect at common (possibly empty) faces. Triangulation has important applications in computer-aided design, computer graphics, finite element analysis, etc. In this paper we are concerned with triangulations of 3-dimensional convex polytopes without introducing Steiner points, i.e. all vertices of the simplices must be from the original polytope.

Triangulations in three dimensions (also called *tetrahedralization*) is to partition the polytope into a set of tetrahedra. Three-dimensional triangulations have many interesting properties. Convex polytopes can always be triangulated, but non-convex polytopes may not: the Schönhardt polytope [Sch28] is such an example. It is even NP-complete to determine whether a given non-convex polytope can be triangulated (without using Steiner points) [RS92]. Different triangulations of the same convex polytope may have different *size*, i.e. having different numbers of tetrahedra. *Minimum* *triangulations* of a polytope are triangulations that have the minimum size. Note that minimum triangulation may not be unique for a polytope, i.e. there may be several triangulations having the same minimum size, e.g. an octahedron. The complexity of finding a minimum triangulation was a major open problem [BE95], until quite recently it was shown to be NP-hard [BDR00, BDR00a]. This raises the question of finding good approximation algorithms for this problem.

We first state a few terms and definitions. We consider polytopes with vertices in general position, i.e. no four vertices are coplanar. This implies that the polytope is simplicial. A *cap* of a vertex v in a polytope P is the region between P and the convex hull (CH) of $P \setminus \{v\}$. An edge in a triangulation is called an *interior edge* if it does not lie on the surface of the polytope. A *3-cycle* is a closed path of three edges on the surface graph of a polytope such that each side of the 3-cycle contains at least one other vertex. Throughout this paper, let n denote the number of vertices of the polytope P, e_m the number of interior edges in a minimum triangulation of P, and Δ the maximum degree on the surface graph of P. For any polytope, the number of interior edges e in a triangulation is directly related to the number of tetrahedra t by the formula $t = e + n - 3$ [BBDR00]. This implies a lower bound of $n - 3$ tetrahedra for any triangulation.

There are two common ways to triangulate a convex polytope P. One way is to remove the cap of a vertex v (which can easily be triangulated) to get a smaller polytope $CH(P \setminus \{v\})$, and then to iterate this process for $CH(P \setminus \{v\})$. This is known as 'peeling' [EPW90]. Peeling a vertex v of degree

$deg(v)$ gives $deg(v) - 2$ tetrahedra. Using the 'greedy peeling' heuristic, it can be shown to yield an approximation ratio of $3 - \Omega(\frac{1}{n})$. 'Pulling' [Lee97] (also called 'starring' [Bern93]) is another well-known approach. Pulling picks a highest-degree vertex of the polytope and uses it to form tetrahedra with all facets not containing v, producing $2n - 4 - \Delta \leq 2n - 7$ tetrahedra, which, considering the $n - 3$ lower bound, gives an approximation ratio of $2 - \Omega(\frac{1}{n})$.

The $n - 3$ lower bound is tight, as achieved by 'stacked' polytopes [RS85, BDR00a]. These polytopes contain a lot of 3-cycles. This suggests partitioning a polytope along 3-cycles (if any) may yield a smaller triangulation. However, it is shown in [STT88] that there exist polytopes requiring as many as $2n - 10$ tetrahedra for any triangulation. For $n > 12$, $\Delta \geq 6$ and the pulling heuristic gives no more than $2n - 10$ tetrahedra. Thus the pulling heuristic is worst-case optimal, in terms of the absolute number of tetrahedra produced. In terms of the approximation ratio, however, we can use the idea of 3-cycles to give a better heuristic with ratio $2 - \Omega(\frac{1}{\sqrt{n}})$.

Despite the small improvement, we show that no other approximation algorithm can give a better ratio, under the restriction that only the combinatorial structure of the polytope is considered. By *combinatorial structure* we mean the inclusion relation between the faces, as represented by the *face lattice* [Zie95]. Two polytopes are *combinatorially equivalent* (i.e. have the same combinatorial structure) if there is a bijection between their faces that preserves the inclusion relation. It is known that two 3-polytopes are combinatorially equivalent if and only if their surface graphs are isomorphic. Two polytopes that have the same combinatorial structure may have different coordinatizations, i.e. having different vertex coordinates. In their paper [BBDR00], Below et. al. showed that the size of minimum triangulation of a polytope is not an invariant of its face lattice, i.e. polytopes with the same face lattice may have different minimum size triangulations. They achieved this by considering something called 'vertex-edge chain structure', and showed that it has the property that the size of triangulation of a polytope with that structure changes greatly depending on the presence of a single interior edge. We extend their result to show that this difference can indeed be very large, and use this to prove a lower bound of $2 - O(\frac{1}{\sqrt{n}})$ on the approximability of this problem. This shows that our algorithm is best possible in this sense.

The paper is organized as follows. In Section 2 we first prove some extended properties about the vertex-edge chain structures. Then in Section 3 we modify the pulling heuristic to obtain a better bound of $2 - \Omega(\frac{1}{\sqrt{n}})$ on the approximation ratio. Finally in Section 4 we show that any algorithm that only considers the combinatorial structure of the polytope cannot give an approximation ratio better than $2 - O(\frac{1}{\sqrt{n}})$. We conclude with some open problems.

2 Vertex-Edge Chain Structure

The so-called '*vertex-edge chain structure*' (VECS), that appeared in [BBDR00], will be used extensively in this paper. It consists of $2m+2$ faces (a, q_i, q_{i+1}) and (b, q_i, q_{i+1}) for $i = 0, ..., m$, with the additional restriction that the line

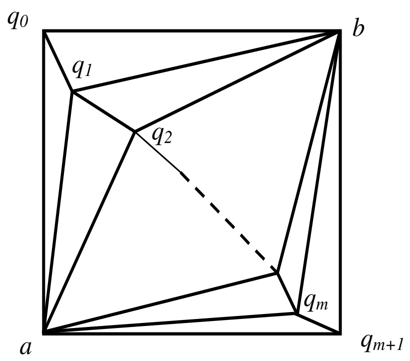


Figure 1: A VECS of size m.

segment q_0q_{m+1} goes through the interior of the polytope $CH(a, b, q_0,\ q_1,$ ..., $q_{m+1})$; we call this polytope a *wedge* (Fig. 1). All the $2m + 2$ faces form a convex surface. We say this VECS has *size* m, and call edge ab the *main diagonal*. A tetrahedron in a triangulation is said to be *incident* to a VECS if at least three out of its four vertices lie on the VECS.

A VECS of size m can be triangulated using $m + 1$ incident tetrahedra abq_iq_{i+1} $(0 \leq i \leq m)$ and one interior edge ab. But if the main diagonal is not allowed to appear in the triangulation, the situation is different: in [BBDR00] it was proved that if ab is not in the triangulation, then at least m other interior edges must be present, and this implies a large number of incident tetrahedra. We now extend this and state some slightly stronger results concerning VECS.

Lemma 1 *(i) In a polytope containing a VECS of size m as a substructure, if the main diagonal is not present in a triangulation, at least* $2m$ *tetrahedra must be incident to the VECS.*

(ii) If a polytope contains k mutually disjoint VECSs, each of size m, as substructures, and all without the main diagonal in a triangulation, then in this triangulation there exist at least $2mk$ tetrahedra incident to these VECSs.

Proof. (i) There are $2m+2$ faces in the VECS. If each face is associated with a separate tetrahedron, then there exist at least $2m$ tetrahedra. Faces aq_iq_{i+1} and bq_iq_{i+1} cannot be in the same tetrahedron because this will induce edge ab. If $aq_{i-1}q_i$ and aq_iq_{i+1} are in the same tetrahedron, (similarly for $bq_{i-1}q_i$ and bq_iq_{i+1}), $aq_{i-1}q_{i+1}$ will be a new face incident to this VECS, which determines another incident tetrahedron. If this new face is in the same tetrahedron as another face, this will in turn determine another incident tetrahedron. This process can continue until the two faces aq_0q_{m+1} and bq_0q_{m+1} appear. A simple induction argument on this idea can show that at least $2m$ tetrahedra are incident to this VECS.
(ii) A tetrahedron incident to one of the VECSs has at least three vertices in that VECS, leaving at most one vertex, thus this tetrahedron cannot be incident to other VECSs. It follows from (i) that the k VECSs must have $2mk$ distinct incident tetrahedra. □

Two VECSs $(a, b, q_0, ..., q_{m+1})$ and $(a', b', q'_0, ..., q'_{m'+1})$ are said to be *interlocked* if ab intersects faces $a'b'q'_0$ and $a'b'q'_{m'+1}$, and $a'b'$ intersects faces abq_0 and abq_{m+1}. More generally, $k > 2$ VECSs are said to be *pairwise-interlocked* (or simply 'interlocked') if any two of them are interlocked.

Lemma 2 *In any triangulation of a polytope containing k pairwise-interlocked*

VECSs, each of size m, there exists at most one VECS having fewer than $2m$ incident tetrahedra.

Proof. If a VECS has fewer than $2m$ incident tetrahedra, its main diagonal must be used in the triangulation: two triangular faces of that VECS have to form a tetrahedra with the main diagonal (see proof of Lemma 1). Thus the main diagonals of any other interlocked VECSs, if used, will penetrate this tetrahedron, thus cannot be used for triangulation. Therefore all other VECSs cannot have fewer than $2m$ incident tetrahedra. $\square$

3 The Triangulation Algorithm

3.1 A restricted case

We first consider the case where the given polytope has no 3-cycles. This restriction will be removed in the next subsection. We analyse the approximation ratio of the pulling heuristic in this case.

Lemma 3 *Let P be a convex polytope that has no 3-cycles, $n > 4$. Then*
(i) At least one interior edge must be incident to the cap of any vertex;
(ii) $e_m + \Delta \geq \sqrt{2n} - 1$.

Proof. (i) Consider a particular vertex v_0 and its neighbors v_1, v_2, ..., v_k where $4 \leq k \leq \Delta$ (note that if $k = 3$, P contains a 3-cycle). The triangle $v_0v_1v_2$ must be associated with a tetrahedron. If the fourth vertex v of this tetrahedron is one from v_3, v_4, ..., v_k, then either v_1v or v_2v (or both) is an interior edge (otherwise v_0v_1v or v_0v_2v would be a 3-cycle). If v is not one from $v_3, v_4, ..., v_k$, then v is not a neighbor of v_0 and hence v_0v is an interior edge. In either case, at least one interior edge is incident to the cap of v_0.

(ii) Suppose we count the number of interior edges by counting the number of interior edges incident to all the n possible caps. By (i) above, this number is at least n. However, each interior edge is counted more than once; it is counted by the caps of their two endpoints as well as the endpoints' neighbors. The total number of these caps does not exceed $2(\Delta + 1)$. Therefore each edge is counted at most $2(\Delta+1)$ times, giving $e_m \times 2(\Delta+1) \geq n$. Applying the Arithmetic-Geometric-Mean inequality, $\frac{e_m+(\Delta+1)}{2} \geq \sqrt{e_m(\Delta+1)}$, thus $e_m + \Delta \geq \sqrt{2n} - 1$. □

The approximation ratio of the pulling heuristic under the no-3-cycle restriction is shown in the lemma below.

Lemma 4 *Using the pulling heuristic, the approximation ratio r is bounded above by $\frac{2n-\sqrt{2n}-2}{n-2}$.*

Proof. The pulling heuristic produces $2n - 4 - \Delta$ tetrahedra while the size of minimum triangulation is $e_m + n - 3$. Thus the approximation ratio is bounded by $\frac{2n-4-\Delta}{n-3+e_m}$. Hence by Lemma 3(ii)

$$r \leq \frac{2n - 4 - (\sqrt{2n} - 1 - e_m)}{n - 3 + e_m} = \frac{2n - \sqrt{2n} - 2 + (e_m - 1)}{n - 2 + (e_m - 1)} \leq \frac{2n - \sqrt{2n} - 2}{n - 2},$$

since $e_m \geq 1$ by Lemma 3(i). □

3.2 The general case

We now deal with the general case for polytopes having 3-cycles. Our algorithm is very simple:

- **Step 1.** Find all the 3-cycles, using algorithms such as those in [PY81] and [CN85].
- **Step 2.** Partition the polytope along all these 3-cycles to produce a set of subpolytopes, each is free of 3-cycles.
- **Step 3.** Apply the pulling heuristic to each resulting subpolytopes.

We first consider the time complexity of this algorithm. Finding all 3-cycles takes linear time [PY81, CN85]. Step 2 can be done as follows: Each 3-cycle divides the surface graph into an interior part and an exterior part. Starting from any 3-cycle, we visit all vertices inside this 3-cycle; whenever we encounter edges on other 3-cycles nested in the current 3-cycle, we process this nested 3-cycle first. In this depth-first search manner, we shall first produce the deepest nested subpolytope. We remove this subpolytope and continue to visit the vertices of the current subpolytope, until we are finished with this 3-cycle. We process similarly for the exterior part until all 3-cycles are processed. All the search and removal can be done in time proportional to the size of the components, thus they take $O(n)$ time in total. Finally, picking vertices of maximum degree and pulling for all subpolytopes also take $O(n)$ time in total. Therefore we have the following.

Fact 1 *The algorithm takes* $\Theta(n)$ *time.*

In the following, we prove that the for polytopes with 3-cycles the algorithm achieves the same approximation ratio as the special case in the previous subsection.

Lemma 5 *Let P be a convex polytope with maximum degree Δ and having a 3-cycle. Suppose we cut through the 3-cycle to produce two polytopes P_1 and P_2, having maximum degrees Δ_1 and Δ_2 respectively. Then $\Delta \leq \Delta_1+\Delta_2-2$.*

Proof. Clearly, $3 \leq \Delta_1 \leq \Delta, 3 \leq \Delta_2 \leq \Delta$. We apply a case-by-case analysis.
Case 1. Vertice(s) with degree Δ are on one side (or both) of P_1 and P_2, but not on the 3-cycle, say $\Delta_1 = \Delta$. Then $\Delta_1 + \Delta_2 - 2 \geq \Delta + 3 - 2 > \Delta$.
Case 2. Vertice(s) with degree Δ are on the 3-cycle being cut. Let the degrees of the two new vertices cut from the vertex with degree Δ be d_1 and d_2, respectively. Clearly $\Delta = d_1 + d_2 - 2$. Since $\Delta_1 \geq d_1, \Delta_2 \geq d_2$, we have $\Delta \leq \Delta_1 + \Delta_2 - 2$.
Thus in all cases the inequality holds. □

Lemma 6 *Let P, P_1 and P_2 be the same polytopes as defined in Lemma 5, with n, n_1 and n_2 vertices respectively. Let t, t_1 and t_2 be the number of tetrahedra produced by the pulling heuristic applied to P, P_1 and P_2 respectively. Then $t \geq t_1 + t_2$.*

Proof. Note that $t = 2n - 4 - \Delta, t_1 = 2n_1 - 4 - \Delta_1$, and $t_2 = 2n_2 - 4 - \Delta_2$. Thus

$$t_1 + t_2 = 2(n_1 + n_2) - 8 - (\Delta_1 + \Delta_2) = 2(n + 3) - 8 - (\Delta_1 + \Delta_2)$$

$$= 2n - 4 - (\Delta_1 + \Delta_2 - 2) \leq 2n - 4 - \Delta = t$$

The last inequality follows from Lemma 5. $\square$

The above lemma shows that cutting along a 3-cycle and separately pulling on each piece will not produce more tetrahedra than that of pulling on the original polytope. Note also that any cut will not create new 3-cycles since no new surface edge is created, and the cuts are respect to 3-cycles, i.e. two 3-cycles will never 'cross' each other. Therefore the lemma is also true for multiple cuts.

Lemma 7 *For a polytope with k 3-cycles, the algorithm produces a triangulation with at most $2n - 7 - k$ tetrahedra.*

Proof. We prove this by induction on k. When $k = 0$, the algorithm reduces to a single pulling, which gives $2n - 4 - \Delta \leq 2n - 7$ tetrahedra, so the claim holds. For $k > 0$, assume the statement is true for all $k' < k$. For a polytope with k 3-cycles, the algorithm picks a 3-cycle and partitions the polytope into two subpolytopes. Suppose they have n_1 and n_2 vertices, and k_1 and k_2 3-cycles, respectively. We have $n = n_1 + n_2 - 3, 0 \leq k_1 < k, 0 \leq k_2 < k, k = k_1 + k_2 + 1$. Since each subpolytopes have less than k 3-cycles, by induction assumption their triangulations have size at most $2n_1 - 7 - k_1$ and $2n_2 - 7 - k_2$ respectively. So the size of triangulation of the polytope is at most $(2n_1 - 7 - k_1) + (2n_2 - 7 - k_2) = 2(n + 3) - 14 - (k_1 + k_2) = 2n - 7 - (k_1 + k_2 + 1) = 2n - 7 - k$. Thus the claim holds. $\square$

Theorem 1 *The algorithm gives a bound of $2 - \Omega(\frac{1}{\sqrt{n}})$ on the approximation ratio.*

Proof. Suppose there are k 3-cycles in P, and we apply k cuts to partition P into $k+1$ parts, each is free of 3-cycles. Note that $k = O(n)$. Consider the following two cases:

(i) $k = o(n)$. Since each 3-cycle contains exactly three vertices, there are at most $3k$ vertices lying on 3-cycles. There are at most k degree-3 vertices, since each of them must determine a 3-cycle. Thus there are at least $n-4k = \Theta(n)$ vertices that are neither lying on 3-cycles nor of degree-3. By the same argument as in Lemma 3(i), each of these $\Theta(n)$ vertices has an interior edge incident to their corresponding caps in any triangulation. Like Lemma 3(ii), we have $2e_m(\Delta+1) \geq \Theta(n)$, giving $e_m+\Delta \geq \Theta(\sqrt{n})$. Using a proof similar to that of Lemma 4, we have $r \leq \frac{2n-4-\Delta}{e_m+n-3} \leq 2 - \Omega(\frac{1}{\sqrt{n}})$. This means even if we apply the pulling heuristic directly to the original polytope (without cutting 3-cycles), we still have a bound of $2 - \Omega(\frac{1}{\sqrt{n}})$. By Lemma 6, partitioning the polytope along all 3-cycles before pulling will not increase the number of tetrahedra. Thus our algorithm gives a bound of $2 - \Omega(\frac{1}{\sqrt{n}})$ in this case.

(ii) $k = \Theta(n)$. By Lemma 7, the approximation ratio is thus

$$r \leq \frac{2n-7-cn}{e_m+n-3} \leq 2 - c < 2 - \Omega(\frac{1}{\sqrt{n}})$$

for some constant $c > 0$. □

Although cutting 3-cycles is an important step in our algorithm, it does not always produce minimum triangulations. A counterexample can be constructed using a 'cupola' [BDR00a], i.e. three VECSs of size m attached to the three non-convex sides of a Schönhardt polytope. We add a vertex joining to the triangular bottom face of the cupola, making sure that it is inside the 'visibility cone' of the top face, and the whole polytope remains convex

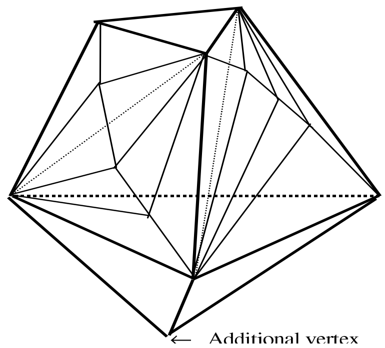


Figure 2: A cupola with a 3-cycle.

(Fig. 2). Now the original bottom face becomes a 3-cycle. This polytope has $3m+7$ vertices. If we cut through the 3-cycle, the three main diagonals cannot be used together, because then the Schönhardt polytope cannot be triangulated. Hence at least one main diagonal cannot be used, and therefore the cupola needs at least $4m$ tetrahedra to triangulate. On the other hand, since the new vertex can see all non-convex facets of the Schönhardt polytope from inside, we can triangulate the Schönhardt polytope by pulling from the added vertex, while the three VECSs are triangulated using their own main diagonals. This gives only $3m+10$ tetrahedra. This shows that cutting 3-cycles does not always produce minimum triangulations.

Note that finding 3-cycles, cutting them, and pulling the resulting subpolytopes, all depends only on the combinatorial structure of the polytope. In other words, they only require the surface graph of the polytope to be given. We show in the next section that this restriction implies a lower bound of approximability.

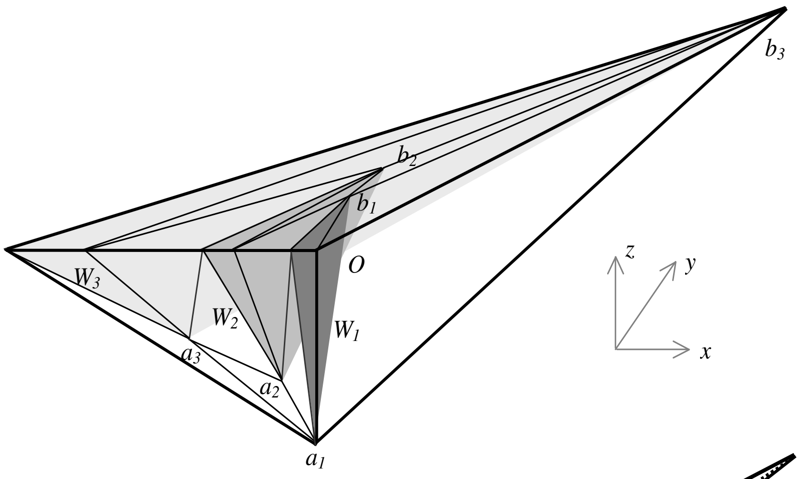
4 Lower Bound of Approximation

In this section, we shall extend the idea in [BBDR00] to show a large difference in the size of minimum triangulations for two different convex polytopes with the same face lattice. As a major result we show that, given only the face lattice, no approximation algorithms for finding minimum triangulations of convex polytopes can have an approximation ratio better than $2 - O(\frac{1}{\sqrt{n}})$.

The idea for our proof is to construct two combinatorially equivalent polytopes, P1 and P2, such that their minimum triangulations differ by a large number of tetrahedra. Both polytopes consist of m thin wedges, each wedge being a VECS of size m. The polytopes have the same number of vertices n where $n = m^2 + 4m$. In P1 the m VECSs are pairwise-interlocked, thus only one of the m main diagonals can be present in any triangulation, resulting in a greater number of tetrahedra. P2 is constructed similarly but the wedges are not interlocked, and thus all m the main diagonals can be present in a triangulation, resulting in a much smaller number of tetrahedra.

4.1 The detailed construction

Fig.3 shows a rough idea of the construction. We first arrange m wedges in an xyz-coordinate system. Wedge W_k $(1 \leq k \leq m)$ has vertices a_k, b_k, c_k, d_k where $a_k b_k$ is the main diagonal. All faces $a_k c_k d_k$ lie on the vertical plane $y = -1$ and all faces $b_k c_k d_k$ lie on the horizontal plane $z = 1$. W_1 is at $a_1(0, -1, -1), b_1(0, 1, 1), c_1(0, -1, 1), d_1(-0.1, -1, 1)$. Assuming wedges W_1



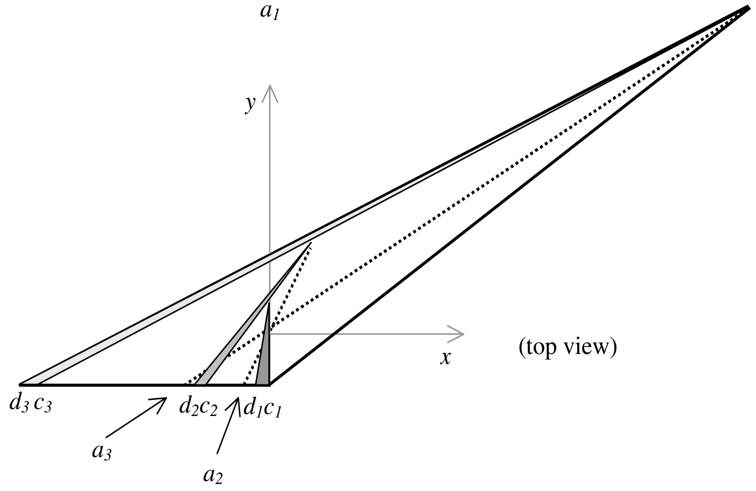


Figure 3: Construction of P1 and P2, showing 3 wedges. q^i's are not shown.

to W_k have been constructed, W_{k+1} is constructed as follows (Fig.3 also shows a top view from the positive z-axis):

Pick a_{k+1} so that its x-coordinate is smaller than d_k and z-coordinate larger than a_k, and such that $a_1, a_2, ..., a_k, a_{k+1}$ form a convex chain w.r.t. the point (0, -1, $-\infty$). b_{k+1} is uniquely determined on the $z = 1$ plane such that $a_{k+1}b_{k+1}$ passes through the origin (0, 0, 0). Find a point c_{k+1} such that $b_{k+1}c_{k+1}$ does not intersect all previous wedge faces on the horizontal plane. d_{k+1} is placed slightly to the 'left' (negative x direction) of c_{k+1}.

The following two facts are not difficult to establish:

1. b_{k+1} has larger y-coordinate than b_k.
2. Vertices $b_1, b_2, ..., b_k, b_{k+1}$ form a convex chain w.r.t. the point $(\infty, 0, 1)$.

Moreover, m points $q_k^1, q_k^2, ..., q_k^m$ are created between each interval (c_k, d_k) and edges $a_k q_k^i, b_k q_k^i$ $(1 \leq i \leq m)$ are added so that they form m VECSs of size m.

This construction puts all the vertices on two planes. This degeneracy can be removed by slightly bending the horizontal and vertical planes so that the polytope becomes strictly convex, while keeping all the main diagonals intersect at the origin.

Now all the wedges intersect at a single point $(0, 0, 0)$. For P1, we apply a final step, moving all the wedges slightly along the positive x-axis: the k-th wedge W_k is moved by a distance of $k\delta$ where $\delta > 0$ is a small constant. This makes any two wedges interlocked. The surface graph of the polytope will not be changed if the perturbations are sufficiently small. P1 is formed by taking the convex hull of all the wedges, having the following edges:

$a_k c_k, a_k d_k, b_k c_k, b_k d_k (1 \leq k \leq m)$;

$d_k a_{k+1}, d_k c_{k+1}, b_k c_{k+1}, a_k a_{k+1}, b_k b_{k+1} (1 \leq k \leq m-1)$;

$a_1 a_k, b_1 b_k (3 \leq k \leq m)$;

$q_k^i q_k^{i+1}, q_k^1 c_k, q_k^m d_k (1 \leq i \leq m-1, 1 \leq k \leq m)$;

$q_k^i a_k, q_k^i b_k (1 \leq i \leq m, 1 \leq k \leq m)$;

$c_1 b_m, a_1 b_m, a_1 d_m$.

The construction of P2 is almost identical to that of P1, except in the final step the wedges do not move but shrink: we draw a vertical line passing through each vertex c_k and intersect edge $d_k a_k$ at a'_k. (By construction, c_k must be on the 'left' of a_k.) We shrink the wedge W_k from $a_k b_k c_k d_k$ to $a'_k b_k c_k d_k$. Then all wedges do not intersect, as can be seen in Fig.3 where a_k coincides with c_k in the top view after shrinking. The chain $a'_1(= a_1), a'_2, ..., a'_m$ is still convex, and the surface graph will not be changed by this transformation. Thus we have the following.

Claim 1 *P1 and P2 have the following properties.*
(i) Both P1 and P2 are strictly convex.
(ii) P1 and P2 have the same surface graph.
(iii) All VECS faces $a_k q_k^i q_k^{i+1}$ *and* $b_k q_k^i q_k^{i+1}$ *are on the surface of P1 or P2.*
(iv) For P1, all wedges are interlocked. For P2, all wedges do not intersect one another.

4.2 Size of the triangulations

We next show that minimum triangulations of P1 contains a much greater number of tetrahedra than that of P2.

Lemma 8 *Any minimum triangulation of P1 contains at least* $2m^2 - m + 1$ *tetrahedra.*

Proof. By Lemmas 1 and 2, among the m main diagonals of the interlocked wedges, only one can be present in any triangulation of P1. Note that all

m VECSs are disjoint in our construction, therefore at least $(m-1)(2m)$ tetrahedra must incident to these $m-1$ VECSs not using their main diagonals. The VECS with the main diagonal creates $m+1$ tetrahedra. Thus, a total of at least $(m-1)(2m)+(m+1) = 2m^2 - m + 1$ tetrahedra must appear in any triangulation. □

Lemma 9 *Any minimum triangulation of P2 contains at most m^2+8m-8 tetrahedra.*

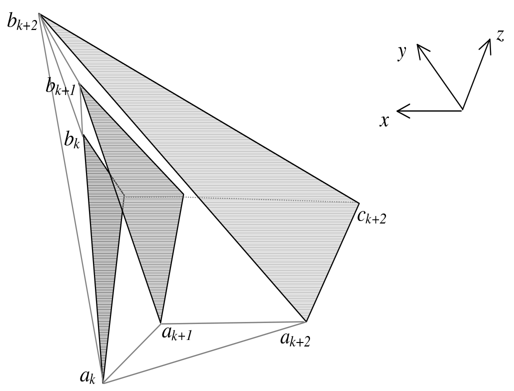


Figure 4: The space between three wedges, viewing from 'the back'. Thickness of each wedge is not shown.

Proof. We triangulate each wedge by using the main diagonal of its corresponding VECS, producing $m+1$ tetrahedra each. Now we have to triangulate the remaining space. Consider any three consecutive wedges W_k, W_{k+1}, W_{k+2} $(1 \leq k \leq m-2)$. We add the interior edges $a_k b_{k+1}$ and $a_{k+1} b_{k+2}$ so that the two regions $a_k b_k d_k a_{k+1} b_{k+1} c_{k+1}$ and

$a_{k+1}b_{k+1}d_{k+1}a_{k+2}b_{k+2}c_{k+2}$ are convex, and they can be triangulated using four tetrahedra each. Finally, the remaining space can be triangulated with three tetrahedra $a_k a_{k+1} a_{k+2} b_{k+2}$, $b_k b_{k+1} b_{k+2} a_k$, and $a_{k+1} b_{k+1} a_k b_{k+2}$, by adding the edge $a_k b_{k+2}$ (Fig.4). Using the above procedure, wedge W_{k+1} is effectively 'shielded' and the triangulation can proceed with the remaining $m-1$ wedges as if there is no W_{k+1}. We use this to triangulate P2 starting from $W_1W_2W_3$, then $W_1W_3W_4$ and so on until only two wedges W_1 and W_m remain. The space beyond W_1 and W_m can be triangulated by adding the tetrahedra $a_1b_1c_1a_m$ and $a_mb_md_ma_1$ for the two outer faces. This completes the triangulation.

There are m VECSs, each giving $m+1$ tetrahedra, each of the $m-1$ convex regions between W_k and W_{k+1} gives three tetrahedra, each of the $m-2$ inductive steps gives three more tetrahedra, and the final step gives two more. Thus there is a triangulation with total number of tetrahedra $m(m+1)+4(m-1)+3(m-2)+2 = m^2+8m-8$. □

Theorem 2 *Any approximation algorithm for finding minimum triangulations of convex polytopes, provided with only the face lattice, cannot have approximation ratio better than* $2 - O(\frac{1}{\sqrt{n}})$.

Proof. By Lemmas 8 and 9, P1 has at least $2m^2-m+1 = 2n-\Theta(\sqrt{n})$ tetrahedra while P2 has at most $m^2+8m-8 = n+\Theta(\sqrt{n})$ tetrahedra. Note that P1 and P2 are combinatorially equivalent but have different coordinates. Since minimum triangulations of two combinatorially equivalent polytopes can have either at least $2n-\Theta(\sqrt{n})$ or at most $n+\Theta(\sqrt{n})$ tetrahedra, any approximation algorithm that considers the face lattice only must produce a triangulation with the worst case number, i.e. $2n - \Theta(\sqrt{n})$. Otherwise, this algorithm would produce a triangulation with less than optimal number of tetrahedra in some cases, a contradiction. Since a minimum triangulation may in fact contain at most $n + \Theta(\sqrt{n})$ tetrahedra, the approximation ratio is bounded by $r \geq \frac{2n - \Theta(\sqrt{n})}{n + \Theta(\sqrt{n})} \geq 2 - O(\frac{1}{\sqrt{n}})$. $\square$

5 Conclusion

In this paper, we give an approximation algorithm for finding a minimum triangulation of convex polytopes with approximation ratio $2 - \Omega(\frac{1}{\sqrt{n}})$. We also show a $2 - O(\frac{1}{\sqrt{n}})$ bound on the approximability of this minimum triangulation problem using combinatorial algorithms. Thus there is no better constant-ratio approximation algorithm without considering the actual coordinates of the vertices. A problem that follows naturally is to consider whether algorithms with better approximations ratios can be achieved, or similar non-approximability results exists, if vertex coordinates are taken into account. Another interesting problem is to consider the case when the maximum degree is bounded by a constant. It is not known whether the problem remains to be NP-hard in this case, and it is also unclear whether the bounds we gave can be improved.

References

[BBDR00] Below A., Brehm U., De Loera J., and Richter-Gebert J., Minimal Simplicial Dissections and Triangulations of Convex 3-Polytopes, *Discrete and Computational Geometry* **24** (2000), pp. 35-48.

[BDR00] Below A., De Loera J., and Richter-Gebert J., Finding Minimal Triangulations of Convex 3-Polytopes is NP-Hard, *Proceedings of 11th Annual ACM-SIAM Symposium On Discrete Algorithms* (2000), pp. 65-66.

[BDR00a] Below A., De Loera J., and Richter-Gebert J., The Complexity of Finding Small Triangulations of Convex 3-Polytopes, manuscript, 2000. Available at

http://www.math.ucdavis.edu/~deloera/RECENT_WORK/recent.html

[Bern93] Bern M., Compatible Tetrahedralizations, *Proceedings of 9th ACM Symposium on Computational Geometry* (1993), pp. 281-288.

[BE95] Bern M. and Eppstein D., Mesh Generation and Optimal Triangulation, in *Computing in Euclidean Geometry*, World Scientific, Singapore, 1995.

[CN85] Chiba N. and Nishizeki T., Arboricity and Subgraph Listing Algorithms, *SIAM Journal on Computing* **14** (1985), pp. 210-223.

[EPW90] Edelsbrunner H., Preparata F. P., and West D. B., Tetrahedrizing point sets in three dimensions, *Journal of Symbolic Computation* **10** (1990), pp. 335-347.

[Lee97] Lee C. W., Subdivision and Triangulations of Polytopes, in *Handbook of Discrete and Computational Geometry* (Goodman J. E. and O'Rourke J.

eds.), CRC Press, 1997.

[PY81] Papadimitriou C. H. and Yannakakis M., The Clique Problem for Planar Graphs, *Information Processing Letters* **13** (1981), pp.131-133.

[RS85] Rothschild G. L. and Straus E. G., On Triangulations of the Convex Hull of n Points, *Combinatorica* **5** (1985), pp.167-179.

[RS92] Ruppert J. and Seidel R., On the Difficulty of Tetrahedralizing 3-Dimensional Non-convex Polyhedra, *Discrete and Computational Geometry* **7** (1992), pp. 227-253.

[Sch28] Schönhardt E., Über die Zerlegung von Dreieckspolyedern in Tetraeder, *Math. Annalen* **98** (1928) pp. 309-312.

[STT88] Sleator D., Tarjan R. and Thurston W., Rotation Distance, Triangulations, and Hyperbolic Geometry, *Journal of the American Mathematical Society* **1** (1988), pp. 647-681.

[Zie95] Ziegler G. M., *Lectures on Polytopes*, Graduate Texts in Mathematics 152, Springer-Verlag, New York, 1995.