Abstract of thesis entitled
'Approximation for Minimum Triangulations
of Convex Polyhedra'

submitted by
Fung Ping Yuen

for the degree of Master of Philosophy
at the University of Hong Kong in July 2001

Triangulation is a fundamental problem in Computational Geometry. In three dimensions, it means partitioning a polyhedron into a set of tetrahedra, such that they only overlap in common (possibly empty) faces. Triangulation has important applications in finite element analysis, computer graphics, mesh generation, computer-aided design, etc., and is also of fundamental theoretical significance.

A minimum triangulation of a polyhedron is a triangulation that contains the minimum number of tetrahedra among all the possible triangulations. It was proved to be NP-hard to find a minimum triangulation of convex polyhedra. Therefore finding efficient approximation algorithms to this problem becomes significant. This thesis aims to address this problem.

In this thesis, a new triangulation algorithm for convex polyhedra is given with approximation ratio $2 - \Omega(\frac{1}{\sqrt{n}})$, where n is the number of vertices of the polyhedron. This improves upon the approximation ratios of previous algorithms, and is the first algorithm specifically targeted at minimizing the approximation ratio instead of the actual size of the triangulation. This improvement is obtained by discovering new relationships between the size of minimum triangulations and the combinatorial properties of convex polyhedra.

A lower bound $2 - O(\frac{1}{\sqrt{n}})$ on the approximation ratio is also shown for algorithms that only consider the combinatorial structure of the polyhedra. This is achieved by showing that polyhedra with the same combinatorial structure can have vast difference in their sizes of minimum triangulations.

Finally, better approximation ratios are achieved using the new algorithm for certain classes of polyhedra, namely, polyhedra with constant degrees, and polyhedra without degree-4 vertices and 3-cycles.

Approximation for Minimum Triangulations of Convex Polyhedra

by

Fung Ping Yuen

B.Eng. (Computer Engineering), the University of Hong Kong, 1999

A thesis submitted in partial fulfilment of
the requirements for the degree of
Master of Philosophy

in

Department of Computer Science and Information Systems
at
the University of Hong Kong

July 2001

Declaration

I declare that this thesis, submitted in partial fulfilment of the requirements for the degree of Master of Philosophy, represents my own work (except where due acknowledgement is made), and that it has not been previously included in a thesis, dissertation or report submitted to this University or to any other institution for a degree, diploma or other qualification.

Fung Ping Yuen
July 2001

Acknowledgements

It is probably needless to say that there are too many people to thank for their assistance in the course of my graduate work and the preparation of this thesis. Nevertheless I will try to acknowledge all of them.

On top of all, I would like to thank my supervisor, Prof. Francis Y. L. Chin for his supervision, guidance and encouragement, and in particular pointing me to this topic. He often has stunning ideas that lead a new way out when it seems there can be no further progress.

I would also like to express my gratitude to the examiners, Prof. J. A. De Loera and Prof. C. A. Wang, who spent their valuable time to carefully read this thesis, and to give many suggestions on improving it. Prof. Wang has also collaborated with us for some time; in particular, he pointed out a fatal flaw in an earlier version of an upper bound proof, which eventually lead to a whole series of new ideas and results.

I would also like to thank the Department of Computer Science and Information Systems, the University of Hong Kong (HKU), for the financial support in the form of postgraduate studentships; and the Committee on Research and Conference Grants (CRCG) of HKU for its support in the form of conference grants for attending a conference where many of the results are presented.

Finally, I must give my deepest thanks to my parents, whose support although being 'invisible' are of utmost importance.

Contents

List of Figures

Chapter 1

Introduction

1.1 The Problem

One of the most fundamental problems in Computational Geometry [32] concerns the decomposition of polyhedra, which has been studied extensively for over two decades. *Polyhedron decomposition* is the problem of partitioning a d-dimensional polyhedron into simpler polyhedra. These polyhedra should be non-overlapping except at the common vertices, edges, faces, etc. shared by them. *Polyhedron triangulation* is the decomposition of a polyhedron into simplices. In two dimensions, polyhedron triangulation is reduced to triangulation of simple plane polygons, which has already been extensively studied. The three dimensional counterpart, however, has received little attention until quite recently. In this research, we are interested in three-dimensional polyhedron triangulation, which means partitioning a polyhedron into a set of tetrahedra.

Many different problems about three-dimensional triangulations have been studied, including Delaunay triangulations [7], triangulating with good shape parameters, e.g., aspect ratios [12, 29], etc. Our focus is on finding a triangulation that consists of a small number of tetrahedra.

1.2 Motivation

Triangulation has fundamental theoretical significance as well as important applications. On the theoretical side, triangulations of three-dimensional polyhedra are a natural generalization of triangulations of two-dimensional polygons, which have been well studied (see e.g. [7, 10]). However, many results in two dimensions become invalid in three dimensions; we shall see some of these in Section 1.4. The problem also has far-fetching connections to different areas of theoretical computer science, such as rotation distance of binary trees in data structures [38]; and in mathematics, including the topology of 3-manifolds, interpolation of three-variable functions [6], oriented matroids [3], etc. It is also of foundational importance in its own right.

On the applied side, triangulations have important applications in numerical computing (e.g., solution of partial differential equations by the finite element method [39]), mesh generation, computer-aided design, tool design, computer graphics, and Geographic Information Systems (GIS) [18], to name a few. In these applications, the objects (e.g., the parts in manufacturing a tool, or a region where a PDE is to be solved) are required to be modelled using polygons and polyhedra. There are few or even no efficient algorithms to solve some of the problems concerning these polyhedra. However, when they are decomposed into tetrahedra, efficient algorithms can be obtained more easily. Therefore many computations on polyhedra involve decomposition as a first step. Developing efficient algorithms for polyhedron decomposition thus has important applications.

Finding a small triangulation of polyhedra helps to save the running time and storage space for algorithms processing a lot of polyhedra. This is the case when modelling a complicated real-world object, e.g., an aircraft. This problem also arises in Computer Algebra and Mathematical Programming, polyhedral techniques in Algebraic Geometry, and the characterization of f-vectors of triangulations of polytopes [5].

1.3 Terminology and Definitions

To begin with, we state some terminology and definitions regarding polyhedra and their triangulations. Most of the following are adapted from [43], [25] and [23].

1.3.1 Polyhedra

A (convex) *polyhedron* is an intersection of finitely many closed halfspaces in $\Re^d$. A *polytope* is a bounded polyhedron. Equivalently, a polytope is the convex hull of a finite set of points in $\Re^d$. This distinction of boundedness is not important for our purpose, and in three dimensions 'polyhedron' is more commonly used, while 'polytope' is more commonly used in higher dimensions. Thus we will use the term 'polyhedron' in this thesis. Non-convex polyhedra can be regarded as unions of convex polyhedra [9] although we will not consider them in this thesis.

A *face* of a polyhedron P is the intersection of P with a supporting hyperplane (i.e., a hyperplane that has non-empty intersection with P and P lies completely in one of the closed halfspaces defined by the hyperplane). The dimension of a face is the dimension of its affine hull. So vertices and edges are 0-dimensional and 1-dimensional faces, respectively. $(d-1)$-dimensional faces are called *facets*. Since we only treat 3-dimensional objects, without confusion we simply call the 2-dimensional facets 'faces'.

A set of vertices in 3-D are in *general position* if no three of them are collinear and no four of them are coplanar. We assume the vertices of a polyhedron are always in general position in this thesis. Vertices not in general position can be perturbed by various means; see [37, 20, 1] etc. The general position assumption implies that the polyhedron is strictly convex (no coplanar faces), and also *simplicial*, i.e., all faces are triangles.

A *simplex* is the convex hull of $d+1$ affinely independent points in $\Re^d$. It is the simplest shape in d-dimension, and decompositions terminate when all components become simplices. In three dimensions a simplex is a *tetrahedron*.

1.3.2 Triangulations

A *subdivision* of a polyhedron P is a finite collection $T = \{P_1, ..., P_m\}$ of d-dimensional polyhedra such that [25]:

1. The vertices of each P_i are drawn from the vertices of P;
2. P is the union of $P_1, ..., P_m$; and
3. If $i \neq j$ then $P_i \cap P_j$ is a common (possibly empty) face of the boundaries of P_i and P_j.

A *triangulation* is a subdivision in which each P_i is a simplex. The *size* of a triangulation T is the number of tetrahedra in the triangulation.

We note that the terminology in the literature is quite confusing; the word *triangulation* is usually used in two dimensions and for the general d-dimensional problem, but also sometimes for three dimensions. Another word *tetrahedralization* is commonly used for three dimensions, but sometimes the words *triangulation* and *tetrahedralization* are used interchangeably in three dimensions. (The word 'triangulation' may have another meaning in three dimensions, meaning the subdivision of the surface of a three-dimensional object into triangles, but this is not our concern.) Since we are only interested in three dimensions in this thesis, we shall use the term 'triangulation' for brevity. Unless otherwise stated the word 'triangulation' is used in the three-dimensional sense. We shall not discuss the general d-dimensional problem in this thesis.

The *surface graph* of a polyhedron P is its 1-skeleton (i.e., the set of all faces of P of dimension at most one). An edge in a triangulation is called an *interior edge* if it does not lie on the surface of the polyhedron. A *cap* of a vertex v in a polyhedron P is the polyhedron formed by the closure of $P \setminus CH(P \setminus \{v\})$ where $CH(\,)$ denotes the convex hull.

Throughout this thesis, n denotes the number of vertices of a given polyhedron, e_m the minimum number of interior edges required among all triangulations of the polyhedron, and Δ the maximum vertex degree on the surface graph.

1.4 Related Work

In view of the importance of triangulations, much research has been done on the polyhedron triangulation problem. Before we state the results we achieve, we give in this section a brief overview of some related previous work.

1.4.1 Properties of Polyhedra

Some properties of polyhedra are useful to us. These are well-known and we only state the results below. [43, 32]

Proposition 1.1 (Euler's theorem) *For a polyhedron with n vertices, e edges and f faces, we have $n - e + f = 2$.*

Proposition 1.2 *For a maximal (i.e., triangulated) planar graph with n, e and f as in Proposition 1.1, we have $e = 3n - 6$, $f = 2n - 4$.*

Proposition 1.3 (Steinitz' Theorem) *A graph G is the surface graph of a 3-dimensional polyhedron if and only if G is simple, planar, and 3-connected. (A graph is d-connected if the removal of any $d - 1$ or fewer vertices leaves a connected graph.)*

As a result of Propositions 1.2 and 1.3, the average vertex degree of a simplicial polyhedron is $6 - \frac{12}{n}$. This implies:

Proposition 1.4 *In any polyhedron, there exists at least one vertex of degree at most five. If $n > 12$, $\Delta \geq 6$.*

1.4.1.1 Combinatorial Structure

We can partition all polyhedra into equivalent classes by considering their *combinatorial structure*. This refers the inclusion relation between the faces (of any dimension) of the polyhedron, as represented by the *face lattice* [43]. Two polyhedra are *combinatorially equivalent* (i.e., have the same combinatorial structure) if there is a bijection between the faces that preserves the inclusion relation. It is same as saying that their face lattices are the same (isomorphic). It is known that, in three dimensions, two polyhedra are combinatorially equivalent if and only if their surface graphs are isomorphic [43]. Two combinatorially equivalent polyhedra may have different *coordinatizations*, i.e., different vertex coordinates or relative vertex positioning.

Operations on polyhedra are called *combinatorial invariant* if they do not depend on any non-combinatorial properties. Other operations may require the exact vertex coordinates or the relative vertex positioning to be known. Combinatorial invariant operations give the same results/effects when applied to all combinatorially equivalent polyhedra.

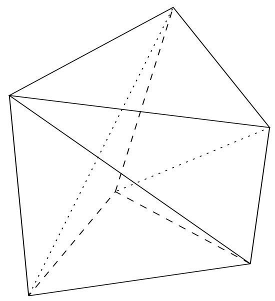

Figure 1.1: A Schönhardt polyhedron.

1.4.2 Properties of 3-D Triangulations

While the two-dimensional triangulation problem has been thoroughly studied and has nice properties, the three-dimensional analogue is in general much more difficult, but also interesting.

For instance, while triangulation is always possible for a given simple polygon, the same is not true in three dimensions: there exist polyhedra that cannot be triangulated, as proved by Lennes [26] as early as in 1911. We call polyhedra that admit some triangulations *triangulable*. The most famous non-triangulable polyhedron is Schönhardt's polyhedron [36](Fig. 1.1). It is obtained by 'twisting' the top face of a triangular prism slightly, and allowing the rectangular side faces to fold and bend in. The convex hull of any four vertices of the resulting polyhedron has parts outside the polyhedron, thus not triangulable. This is the smallest non-triangulable polyhedron. Bagemihl [2] gave a general construction for n-vertex non-triangulable polyhedron. Even worse, the problem of deciding whether a given non-convex polyhedron is triangulable is NP-complete [35].

Non-triangulable polyhedra can always be triangulated by using additional vertices, called *Steiner points*, but determining how many Steiner points are required for a particular polyhedron is also NP-hard [35]. Steiner points cause some problems like complications in data representation and round-off errors due to numerical inaccuracy [11], although it was noted that using Steiner points can reduce the size of triangulations, even for convex polyhedra [3]. As we shall see, Steiner points are not necessary for convex polyhedra, so we do not use Steiner points for triangulation in this thesis.

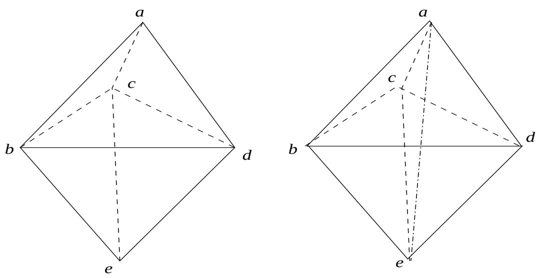

Figure 1.2: Two triangulations of a bipyramid. The tetrahedra are (left) $abcd, bcde$; and (right) $abce, acde, abde$.

In two dimensions, any triangulation of a n-vertex polygon gives the same number $(n-2)$ of triangles. This is again not true in three dimensions. Different ways of triangulating the same polyhedron can result in different sizes of triangulations. In fact, it can vary from $O(n)$ to $O(n^2)$. One example is the *cyclic polyhedron* [7, 43], formed by the convex hull of the vertices lying on the *moment curve* $\{v_i = (i, i^2, i^3), i = 0, 1, ..., n\}$. It can be triangulated into $\binom{n-2}{2}$ tetrahedra of the form $\{v_i v_{i+1} v_j v_{j+1}\}$. This is the maximum possible number of tetrahedra for any triangulation [20]. However, using the 'pulling' approach (to be discussed in Section 1.4.4) the same polyhedron gives only $2n-7$ tetrahedra. In fact, there exists an operation, called *flipping* [7], involving a set of five vertices, that exchange a group of two tetrahedra into three, and vice versa, by using interior edges (Fig. 1.2).

Flipping suggests that the size of a triangulation may be related to the number of interior edges. This is in fact true: for any polyhedron, the number of interior edges e in a triangulation is directly related to the number of tetrahedra t by the following formula, which was shown in e.g. [20, 3, 33]:

Lemma 1.5 ([20, 3, 33]) *For a polyhedron with n vertices admitting a triangulation having e interior edges and of size t, we have $t = e + n - 3$.*

This formula has two important implications:

1. In order to minimize the size of triangulation, it is equivalent to minimize the number of interior edges of the triangulation.

2. This gives a lower bound of $n-3$ tetrahedra for any triangulation [20, 34].

1.4.3 General Polyhedra

We have seen that not all polyhedra are triangulable. In fact, all (strictly) convex polyhedra are triangulable while non-convex polyhedra may not. Convex polyhedra are our focus of research and will be discussed in the next subsection. Although we are not going to deal with non-convex polyhedra in our results, it is useful to look at some properties of their triangulations.

For general (non-convex) polyhedra, Chazelle studied the problem of partitioning a (non-convex) polyhedron into convex parts, and showed that there exist polyhedra requiring $\Omega(n^2)$ convex pieces [8]. This is also a lower bound for triangulation. This is in fact sufficient; polyhedron can always be triangulated into $O(n^2)$ tetrahedra [8, 9].

Actually, the size of triangulation depends on some measure of non-convexity; in [9] Chazelle and Palios gave an algorithm that produces $O(n+r^2)$ tetrahedra using $O(n+r^2)$ Steiner points and runs in time $O((n+r^2)\log r)$, where r is the number of reflex edges of the polyhedron.

Only a few special types of non-convex polyhedra are known to be triangulable without Steiner points (Steiner points cause problems as we have seen above). One such type is the region between two convex polyhedra (the convex hull of their union, minus the two polyhedra), of which there is a series of results for them [22, 6, 11, 41].

1.4.4 Some Triangulation Algorithms for Convex Polyhedra

Convex polyhedra can always be triangulated, and there are two major ways to perform the triangulation. From these we know that $e_m = O(n)$. Below we describe two algorithms (see [25] for a nice explanation of canonical ways of triangulating a polytope).

1.4.4.1 Pulling

There is a simple and well-known algorithm for triangulating any convex polyhedron called 'pulling' [25]. This is also known as 'starring' [6], 'coning' [5], 'fanning' [14] or 'emission' [40]. Pulling triangulates a polyhedron by picking a vertex v of the polyhedron, and then forms tetrahedra from v to

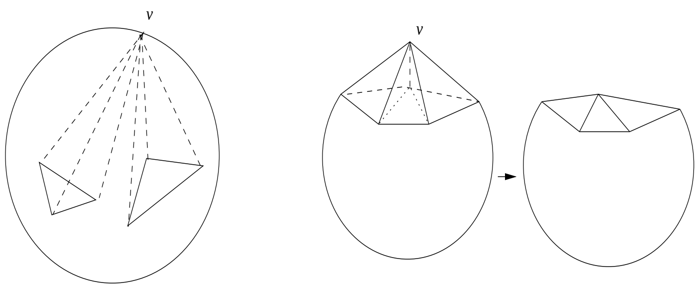

Figure 1.3: (a) pulling. (b) peeling.

all other non-adjacent triangular faces. Among the $2n - 4$ faces, $deg(v)$ (the degree of v) of which are adjacent to v. Thus this algorithm gives $2n - 4 - deg(v) \leq 2n - 7$ tetrahedra. Obviously, one would pick the vertex of highest degree as v to minimize the size of the triangulation. Since $\Delta \geq 6$ for $n > 12$ (Proposition 1.4), this heuristic gives no more than $2n - 4 - \Delta \leq 2n - 10$ tetrahedra.

Pulling is a combinatorial invariant operation, and takes $O(n)$ time with the face information given, e.g. the input is the surface graph.

1.4.4.2 Peeling

Another algorithm is called 'peeling' or 'cap-removal' [6, 9, 20]. In each stage, a vertex v is picked from the polyhedron P, and the cap of the vertex is removed. The cap itself can easily be triangulated (by pulling from v). This results in a smaller convex polyhedron $CH(P \setminus \{v\})$, and the next stage repeats this process until four vertices are left, forming a simplex. Peeling a vertex of degree $deg(v)$ gives $deg(v) - 2$ tetrahedra. Each time we pick a vertex of minimum degree to peel (known as 'greedy peeling'). Since there must exist a vertex having degree at most five (Proposition 1.4), it follows easily [20] that the number of tetrahedra produced is no more than $3n - 11$. If there are many low-degree vertices, the size of triangulation produced may be smaller than pulling.

Peeling can be done in $O(n \log n)$ time for convex polyhedron [6].

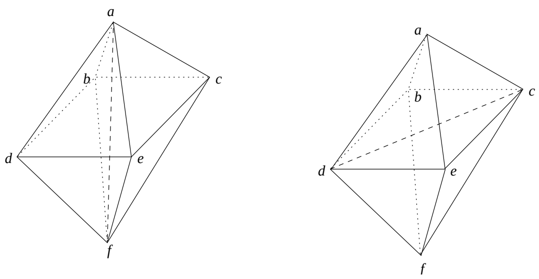

Figure 1.4: Two minimum triangulations of an octahedron. The tetrahedra are (left) *bdaf, deaf, ecaf, cbaf*, and (right) *abcd, acde, bcdf, cdef.*

1.4.5 Minimum Triangulations

We noticed in Section 1.4.2 that different triangulations of the same polyhedron may have different size. One of the problems being studied is to find a triangulation having a small size. A triangulation of a polyhedron is called a *minimum triangulation* if it has the minimum size over all the possible triangulations. Minimum triangulations of a polyhedron may not be unique, i.e., there may be more than one triangulation that gives the same minimum size. An example is an octahedron, that has two triangulations both of size four (Fig. 1.4). These triangulations are indeed minimum, since each tetrahedron can only have two faces on the surface of the polyhedron (there are no degree-3 vertices) and an octahedron has eight faces.

Although an arbitrary triangulation of convex polyhedra is easy to find, the problem of triangulating a convex polyhedron into the minimum number of tetrahedra was a long-standing open problem. The only known lower bound is that any polyhedron requires at least $n-3$ tetrahedra [7], this being a consequence of Lemma 1.5. Comparing with the $2n-7$ bound by pulling, it seems that there is a lot of room for improvement. However, there has not been too much work in the literature on trying to give a smaller size triangulation. Previous research on triangulations tends to optimize other qualities of triangulation, such as the shape of the tetrahedra produced, etc. [12, 29, 7], but not the number of simplices.

One reason for the little progress in this area is due to the lower bound result of Sleator et al. [38], which states that there exist polyhedra requiring at least $2n-10$ tetrahedra for any triangulation. For $n > 12$, the pulling heuristic gives no more than $2n - 10$ tetrahedra (see Section 1.4.4.1). Thus the pulling heuristic is worst-case optimal, in terms of the absolute number of tetrahedra produced. We shall also see in Chapter 2 that some polyhedra only need $n - 3$ tetrahedra in triangulation. Thus the $n - 3$ lower bound is indeed tight for some polyhedra.

To summarize, $n-3$ tetrahedra are necessary for any polyhedron, and for some polyhedra this much is sufficient. $2n - 10$ tetrahedra are sufficient for any polyhedron, and for some this much is indeed necessary. It seems that nothing can be done.

The way out of this is to look at the approximation ratio of the algorithm instead of the absolute size of the triangulation. Only quite recently, the problem of finding minimum triangulations for convex polyhedra is proved to be NP-hard [4, 5]. Due to this result, it is very unlikely that efficient algorithms for this problem exist. As with other NP-complete problems, the usual approach is to develop good approximation algorithms.

However, there are no results specifically addressing this in the literature. The 'pulling' and 'peeling' heuristics mentioned in Section 1.4.4 can be used, but these two methods are not designed specifically to address the minimum triangulation problem, and do not give good approximation ratios. Considering the $n-3$ lower bound, it can be seen that peeling yields an approximation ratio of $3 - \Omega(\frac{1}{n})$ while pulling gives an approximation ratio of $2 - \Omega(\frac{1}{n})$. The ratio $2 - \Omega(\frac{1}{n})$ is in fact tight for the pulling heuristic; see Section 2.3. This thesis aims to address the problem of finding a closer approximation to minimum triangulations.

1.5 Our Results

In view of the NP-hardness of finding minimum triangulations (Section 1.4.5), designing good approximation algorithms for this problem becomes significant. This research is concerned with this problem. The following summarizes the major results that we present in this thesis:

- We establish an efficient algorithm for finding a close approximation to minimum triangulations of convex polyhedra. We give a new approximation algorithm with approximation ratio $2 - \Omega(\frac{1}{\sqrt{n}})$. This improves upon the previous best $2 - \Omega(\frac{1}{n})$ approximation ratio, and is the first algorithm to achieve a nontrivial approximation ratio. This is possible by discovering new relationships between the size of minimum triangulations and the combinatorial properties of polyhedra.

- We obtain lower bounds on the approximability of this problem. We prove a matching lower bound, $2 - O(\frac{1}{\sqrt{n}})$, on the approximability of this problem, under the restriction that only the combinatorial structure of the polyhedron is considered. This shows that our algorithm is optimal in this sense. This is achieved by observing the sizes of minimum triangulations of two convex polyhedra can differ greatly even if they have the same combinatorial structure.

- We give better approximation ratios for special classes of polyhedra using our algorithm. (1) For polyhedra with constant maximum degree Δ we achieve an asymptotic tight approximation ratio $2 - \Omega(\frac{1}{\Delta}) - \Omega(\frac{1}{n})$. (2) For polyhedra without 3-cycles and degree-4 vertices we achieve a tight approximation ratio 3/2. (3) For polyhedra without degree-3 and degree-4 vertices, we achieve an upper bound $2 - \frac{1}{12}$ on the approximation ratio.

Most of the results in this thesis appeared in various conference proceedings and journals [14, 15].

1.6 Organization of the Thesis

The rest of this thesis is organized in the following manner. In Chapter 2 we study the relationships between the size of minimum triangulations and various parameters of a polyhedron, and also introduce some key structures to be used in the chapters to follow. In Chapter 3, we modify the pulling algorithm to obtain an algorithm with a better approximation ratio. In Chapter 4 we show that any algorithm that only considers the combinatorial structure of the polyhedra cannot give an approximation ratio better than what our algorithm achieved. Chapter 5 discusses the approximation ratio of our algorithm applied to some special classes of polyhedra. The final chapter concludes this thesis.

Chapter 2

Basic Structures

In this chapter, we give some preliminary results that we will need in later parts of this thesis. In Section 2.1 we introduce the vertex-edge chain structure, which has a very important property regarding the size of triangulations. In Section 2.2 we discuss 3-cycles and stacked polyhedra. In Section 2.3 we give some relationships between the size of minimum triangulation of a polyhedron and some of its combinatorial properties.

2.1 Vertex-Edge Chain Structure

A key structure that first appeared in [3] will be used extensively in this thesis. This so-called '*vertex-edge chain structure*' (VECS) (also called 'BP-net' in [42]) 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 segment q_0q_{m+1} goes through the interior of the polyhedron formed by tetrahedra abq_iq_{i+1} for all i. This polyhedron is called a *wedge* (Fig. 2.1(a)). All the $2m+2$ faces form a convex surface. We say this VECS is of *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. Note that this definition does not count all tetrahedra in a triangulation, but makes sure that each tetrahedron is counted at most once.

A VECS of size m can be triangulated using $m+1$ incident tetrahedra abq_iq_{i+1} $(0 \leq i \leq m)$ by one interior edge ab. However, if the main diagonal is not allowed to appear in the triangulation, the situation is different. The

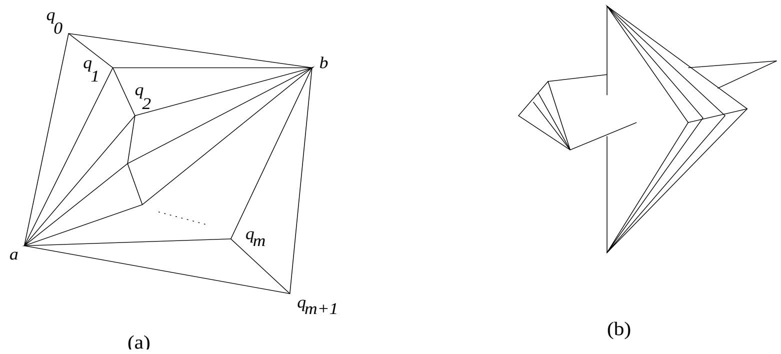

Figure 2.1: (a) A VECS of size m. (b) Two interlocked VECSs.

following lemma was proved in [3]:

Lemma 2.1 (Below et al.) *In a polyhedron P containing a VECS of size m as a substructure, if in a triangulation of P the edge ab is not present, then at least m other interior edges must be present.*

The large number of interior edges implies a large number of tetrahedra (Lemma 1.5). However Lemma 2.1 is not strong enough for some of our results in Chapter 4. We now extend this lemma and state some slightly stronger results concerning VECSs.

Lemma 2.2 *(i) In a polyhedron 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 polyhedron contains k mutually disjoint VECSs each of size m as substructures, all without the main diagonal in a triangulation, then in this triangulation at least $2mk$ tetrahedra are 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. There are only two cases where these faces may share a tetrahedron. 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 with another

neighboring 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 on 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 at least $2mk$ 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} (Fig. 2.1(b)). More generally, $k > 2$ VECSs are said to be *pairwise-interlocked* (or simply 'interlocked') if every pair of them are interlocked.

Lemma 2.3 *In any triangulation of a polyhedron 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 tetrahedron with the main diagonal (see proof of Lemma 2.2). 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. □

2.2 3-cycles and Stacked Polyhedra

A *3-cycle* is a closed path of three edges on the surface graph of a polyhedron such that each side of the cycle contains at least one other vertex (Fig. 2.2(a)). This is the same as what is commonly known in graph theory literature (see e.g. [19]) except that we disallow minimal 3-cycles (i.e., no other vertices inside), since this is simply a face of the polyhedron.

A polyhedron can be partitioned, or 'cut', along a 3-cycle into two parts. Several important properties of this cut are noted below.

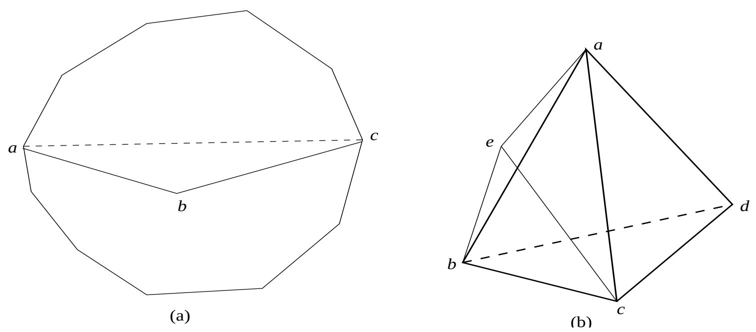

Figure 2.2: (a) a 3-cycle abc. (b) A stacked polyhedron. Vertex e is stacked on the tetrahedron $abcd$. A new vertex may then stack on e.g. face bce.

Observation 2.4 *A cut along a 3-cycle has the following properties:*
(i) each resulting part is convex.
(ii) this operation is combinatorial invariant.
(iii) no new 3-cycles are created in this process, since no new edges are created.
(iv) no other 3-cycles are 'broken' by this cut, since other 3-cycles do not 'cross' this cut. This is because any two 3-cycles do not cross each other, i.e., vertices of a 3-cycle C_1 will not be on different sides of another 3-cycle C_2.

Note that all the above properties are not enjoyed by cycles of greater length.

Observation 2.5 *Suppose v is a vertex of a polyhedron P, and w, w' are any two non-adjacent vertices in the cap of v. If v does not lie on a 3-cycle, then ww' must not be a surface edge. Note that this may not be true if v lies on 3-cycles.*

Observation 2.6 *A degree-3 vertex implies a 3-cycle (formed by the three neighbors).*

A *stacked polyhedron* [34, 5] (Fig. 2.2(b)) is a polyhedron that has a triangulation without interior edges [25]. Thus its size of triangulation is $n-3$, the minimum possible (Section 1.4.2). Equivalently, this means it admits a triangulation whose dual graph is a tree [5]. Stacked polyhedra can be built from simplices one at a time, by 'gluing' a tetrahedron to a face of the polyhedron, maintaining convexity [24]. The reverse of this operation is a triangulation of the polyhedron by peeling off degree-3 vertices one at a time.

Observation 2.7 *A polyhedron can have at most $n - 4$ 3-cycles, achieved when it is a stacked polyhedron.*

In later parts of this thesis, we shall make use of these properties of 3-cycles in proving the bounds of our new algorithm.

2.3 Relationships between the Size of Triangulations and Various Parameters

In this section, we give some relationships between the size of minimum triangulations of a polyhedron and its combinatorial properties, such as its maximum vertex degree and the number of 3-cycles. In Chapter 3, these relationships will be used to prove our new algorithm has the bound as claimed.

Lemma 2.8 *For a polyhedron P with no 3-cycles and $n > 4$, $e_m\Delta \geq n - 2$, and this is tight.*

Proof. We claim that each face of P must have at least one interior edge incident to it. Assume this is not so. Then there is a face $v_0v_1v_2$ of P without incident interior edges; it is in some tetrahedron with fourth vertex v_3, and v_0v_3, v_1v_3, v_2v_3 have to be surface edges of P. Therefore the three triangles $v_0v_1v_3, v_1v_2v_3, v_2v_0v_3$ are either 3-cycles or faces. But 3-cycles are forbidden. If all three triangles are faces, then P is simply a tetrahedron. Therefore our claim holds. Since there are $2n - 4$ faces, there are at least $2n - 4$ interior edges, but each is counted at most 2Δ times since each of the endpoints can be incident to at most Δ faces. Thus $e_m(2\Delta) \geq 2n - 4$, i.e. $e_m\Delta \geq n - 2$. This bound can be achieved by considering a bipyramid [42] (see Fig. 2.3), in which $e_m = 1, \Delta = n - 2$. □

The above tightness construction is clean and simple, but it involves vertices with linear degrees and only a constant number (actually only one) of interior edge. We can construct another example that is almost tight, and having arbitrary degrees (e.g., we may set the degree to be constant, as in Section 5.1) below. Consider a polyhedron consisting of a single wedge $(a, b, q_0, ..., q_{m+1})$ of size m. On top of the faces aq_jq_{j+1} and bq_jq_{j+1}, we build

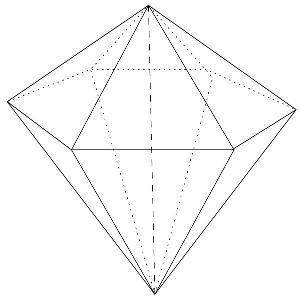

Figure 2.3: A bipyramid with $n-2$ vertices in the middle chain; here $n=8$.

a new VECS also of size m, treating q_jq_{j+1} as the main diagonal (now it becomes an interior edge), with vertices $(q_j, q_{j+1}, r_0 = a, r_1, ..., r_m, r_{m+1} = b)$. Here $j = \lfloor m/2 \rfloor$. Then we recursively build another VECS on top of the faces $q_jr_jr_{j+1}$ and $q_{j+1}r_jr_{j+1}$, and so on and so forth, for k levels, maintaining convexity. See Fig. 2.4. Note that this constructed polyhedron has no 3-cycles. A minimum triangulation of this polyhedron consists of triangulating each of the VECSs using the corresponding main diagonals layer by layer, forming roughly one tetrahedron per vertex. We have

$$n = \underbrace{(m+4) + m + m + ...}_{k \text{ levels}} = km + 4, \Delta = m + 4, e_m = k - 1$$

Thus $e_i\Delta = (k-1)(m+4) \approx km = n - 4 \approx n - 2$. We may pick any k or m (but not both) as we wish. For example, if we want constant degrees, we may pick m to be constant.

Note that Lemma 2.8 depends on the non-existence of 3-cycles. In fact, without this restriction, such a bound is impossible; we can only achieve $e_m\Delta =$ constant. This can be demonstrated by constructing a 'nearly-stacked' polyhedron with e_m and Δ both constant. (We can also use a stacked polyhedron, but then $e_m = 0$ which is uninteresting.) See Fig. 2.5. First consider a tetrahedron $abcd$. On face bcd we stack a new vertex v_1, and then v_2 and v_3 as shown. At any time after we stacked v_1, there is exactly one degree-3 vertex. We now continue to stack inside $v_1v_2v_3$, but notice that we no longer affect the degrees of a, b, c and d. Thus we can repeat this stacking process for any number of times while keeping the maximum degree constant. To the final face, we build the structure in Fig. 2.5(b) on top of it. For this polyhedron, a minimum triangulation consists of triangulating the final block (Fig. 2.5(b))

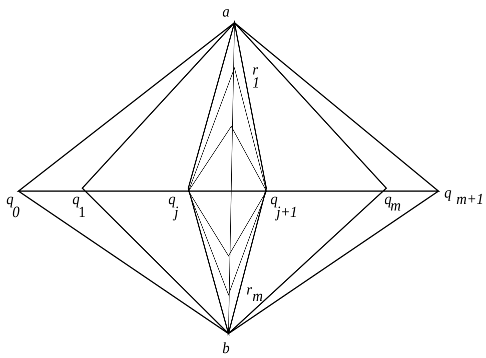

Figure 2.4: A VECS $(q_j, q_{j+1}, r_0, ..., r_{m+1})$ on top of another VECS $(a, b, q_0, ..., q_{m+1})$. A third level can now be constructed on faces $q_j r_j r_{j+1}$ and $q_{j+1} r_j r_{j+1}$.

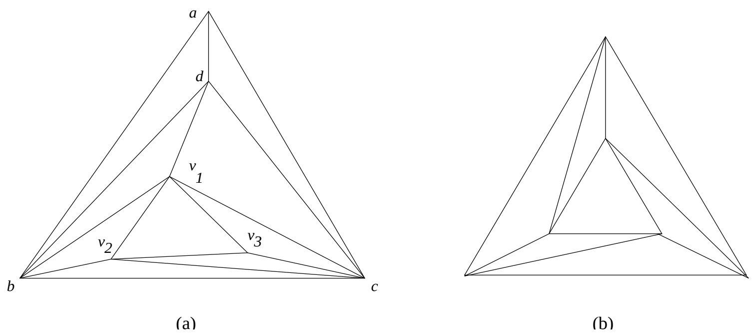

Figure 2.5: Construction showing e_m and Δ both constant.

using a constant number of interior edges, then peel the resulting polyhedron without using interior edges. Thus both e_m and Δ are constant.

Note that this is also a worst-case example on the approximation ratio $2-\Omega(\frac{1}{n})$ of the pulling heuristic. Since both e_m and Δ are constant, the pulling heuristic applied to this polyhedron gives only a ratio of $\frac{2n-4-\Delta}{e_m+n-3} = 2 - \Theta(\frac{1}{n})$. In Section 3.1 we shall see that we can improve this when there are no 3-cycles, using Lemma 2.8.

We can generalize Lemma 2.8 to polyhedra having k 3-cycles:

Lemma 2.9 *For a polyhedron with k 3-cycles and $n > 4$, $e_m\Delta + 3k \geq n - 2$.*

Proof. As in Lemma 2.8, for each of the $2n - 4$ faces, there should be at least one incident interior edge, unless, among the three bounding edges of the face, at least one are on a 3-cycle. Each 3-cycle can share an edge with at most six faces (on both sides of the three edges). Thus there remain at least $2n-4-6k$ faces having incident interior edges. With the same argument as in Lemma 2.8, $e_m(2\Delta) \geq 2n-4-6k$, and the result follows. □

Intuitively, Lemma 2.9 leads to the following important fact: either $e_m\Delta = \Omega(n)$, or $k = \Omega(n)$. In the former case we can follow Lemma 2.8 to show improved lower bound in the size of triangulations, thus improving the approximation ratio. In the latter case, the existence of many 3-cycles implies that the polyhedron is 'stack-like' and we may follow the algorithm to triangulate stacked polyhedra, i.e., cutting along 3-cycles. This gives the idea of the algorithm in the next chapter.

Chapter 3

A New Triangulation Algorithm

In this chapter, we give a new triangulation algorithm that computes an approximate minimum triangulation of a convex polyhedron with approximation ratio $2 - \Omega(\frac{1}{\sqrt{n}})$. We supplement the pulling heuristic with the idea of cutting 3-cycles. We obtain better approximation ratios by producing smaller triangulations in the case when the number of 3-cycles k is large, and raising the lower bound on the size of triangulations when k is small.

3.1 A Restricted Case: No 3-cycles

We first consider the case when the given polyhedron has no 3-cycles. This restriction will be removed in the next subsection. For this special case, we use the pulling heuristic to perform the triangulation. Actually, as we shall see in the next section, our new algorithm for the general case reduces to the pulling heuristic when there are no 3-cycles.

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

Lemma 3.1 *Using the pulling heuristic, the approximation ratio r for polyhedra without 3-cycles is bounded above by $\frac{2n-2\sqrt{n-2}-3}{n-2}$.*

Proof. The approximation ratio is bounded by $\frac{2n-4-\Delta}{n-3+e_m}$. By Lemma 2.8 we have $e_m\Delta \geq n-2$. Hence, using the Arithmetic-Geometric-Mean Inequality,

$$e_m + \Delta \geq 2\sqrt{e_m\Delta} \geq 2\sqrt{n-2}$$

Thus

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

since $e_m \geq 1$ (by Lemma 2.8, $e_m \geq \lceil \frac{n-2}{\Delta} \rceil \geq 1$). □

This lemma shows that the pulling heuristic achieves an approximation ratio of $2-\Omega(\frac{1}{\sqrt{n}})$ in this case, as compared with the worst-case ratio $2-\Omega(\frac{1}{n})$.

3.2 The General Case Algorithm

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

- **Step 1.** Find all 3-cycles, using algorithms such as those in [31] and [13].
- **Step 2.** Partition the polyhedron along all these 3-cycles to produce a set of subpolyhedra, each is free of 3-cycles.
- **Step 3.** Apply the pulling heuristic to each of the resulting subpolyhedra.

The detailed algorithm is shown in the next page.

Algorithm *CutPull*(P)
Input: A convex polyhedron P with n vertices.
Output: T, a set of tetrahedra that triangulates P.

/* Step 1: Find all 3-cycles */
$\mathcal{C} \leftarrow$ Enumerate-triangles(P)
/* finds all 3-cycles, but also include faces of P. See [31, 13]. */
remove from $\mathcal{C}$ all faces of P
/* Step 2: Partition the polyhedron into 3-cycle-free components */
$\mathcal{P} \leftarrow \emptyset$
If $\mathcal{C} = \emptyset$ **Then** $\mathcal{P} \leftarrow P$
Else pick a 3-cycle C from $\mathcal{C}$
    ProcessCycle(C)
    ProcessCycle$'$(C)
    /* same as ProcessCycle( ) but change inside to outside */
/* Step 3: Apply pulling to each subpolyhedron */
$T \leftarrow \emptyset$
for each polyhedron Q in $\mathcal{P}$
    pick a vertex v of highest degree in Q
    $T \leftarrow T \cup \{$ the set of tetrahedra of Q pulled from v $\}$
End.

Procedure ProcessCycle(C)
Traverse for all vertices inside C
If during the traversal, any vertices/edges are found to be on another 3-cycle C', recursively call ProcessCycle(C'), then continue with this traversal
Let R be the polyhedron formed by the searched subgraph
$\mathcal{P} \leftarrow \mathcal{P} \cup R$
remove the searched edges from the graph
remove C from $\mathcal{C}$
End.

Obviously this algorithm produces a correct triangulation of P. The following subsections look at the running time and the approximation ratio attained by this algorithm.

3.2.1 Running Time

We need a data structure for representing polyhedra. We can use any data structure suitable for representing a planar graph, e.g. the Doubly-Connected Edge List (DCEL) [32]. In particular, it should be able to enumerate, say, all edges incident to a vertex in a counterclockwise manner.

Finding all 3-cycles takes linear time [31, 13]. Although these algorithms also return the faces of the polyhedron, we can eliminate them easily, by checking what are the faces on the two sides on each edge of the (candidate) 3-cycle. This only takes linear time in total.

Consider the following simple approach to implement our idea of cutting 3-cycles (Step 2): find a 3-cycle, construct the two subpolyhedra, and then recursively partition the two subpolyhedra. However this may yield an $O(n^2)$ time algorithm due to many unnecessary repetitions in traversing the graphs. We can devise a linear time algorithm for this by making use of the 'nesting' behaviour of 3-cycles.

We perform Step 2 (cutting 3-cycles) 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 recursively first. In this depth-first search manner, we shall first produce the deepest-nested subpolyhedron. We remove this subpolyhedron from the graph and continue to visit the vertices of the current subpolyhedron, until we finish processing 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 being processed, thus they take $O(n)$ time in total.

Finally, picking maximum degree vertices and pulling also take $O(n)$ time for all subpolyhedra. This is because each cut increases the total number of vertices by at most three, and thus after k cuts the total number of vertices in all subpolyhedra becomes at most $n + 3k \leq n + 3(n-4) = O(n)$ (Observation 2.7). Both finding maximum degrees and pulling takes time linear to the number of vertices of each subpolyhedron. Thus the total running time is $O(n)$. Obviously, the problem takes $\Omega(n)$ time. Therefore we have the following.

Theorem 3.2 *Algorithm* **CutPull** *takes* $\Theta(n)$ *time.*

Note that we assume our algorithm is given inputs in the form of combinatorial relationships, e.g. the surface graph. If the surface graph is not given, e.g., we are given only the vertex coordinates, then we need to run a convex hull algorithm first, taking $O(n \log n)$ time instead of just $O(n)$ in above. In this case $O(n \log n)$ is still optimal since a triangulation essentially produces the convex hull of the polyhedron, which takes $\Omega(n \log n)$ time [32].

3.2.2 Approximation Ratio Analysis

We show in this subsection that the approximation ratio of our algorithm is $2 - \Omega(\frac{1}{\sqrt{n}})$ in the general case. We start with a few lemmas.

Lemma 3.3 *Let P be a convex polyhedron with maximum degree Δ and has a 3-cycle. Suppose we cut through the 3-cycle to produce two subpolyhedra 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. At least one vertex with degree Δ is not on the 3-cycle being cut. Then they are on one side (or both sides) of P_1 and P_2, say $\Delta_1 = \Delta$. Then $\Delta_1 + \Delta_2 - 2 \geq \Delta + 3 - 2 > \Delta$.
Case 2. All vertices with degree Δ are on the 3-cycle being cut. Let the degrees of the two new vertices at P_1 and P_2, cut from a vertex with degree Δ, be d_1 and d_2 respectively. Then $\Delta = d_1 + d_2 - 2$. 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 3.4 *Let P, P_1 and P_2 be the same polyhedra as defined in Lemma 3.3, 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

$$\begin{aligned} t_1 + t_2 &= 2(n_1 + n_2) - 8 - (\Delta_1 + \Delta_2) \\ &= 2(n + 3) - 8 - (\Delta_1 + \Delta_2) \\ &= 2n - 4 - (\Delta_1 + \Delta_2 - 2) \\ &\leq 2n - 4 - \Delta \\ &= t \end{aligned}$$

The last inequality follows from Lemma 3.3. □

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 polyhedron. Note also that any cut will not create new 3-cycles, and two 3-cycles will never 'cross' each other (Observation 2.4). Therefore the lemma is also true for multiple cuts.

A closer look on the lemmas reveals that each cut on a 3-cycle will most likely reduce the size of triangulation by at least one, except when the cut lies on maximum-degree vertices. This gives the intuition behind the following lemma.

Lemma 3.5 *For a polyhedron 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 polyhedron with k 3-cycles, the algorithm picks a 3-cycle and partitions the polyhedron into two subpolyhedra. Suppose they have n_1 and n_2 vertices, and k_1 and k_2 3-cycles, respectively. We have the following:

$$0 \leq k_1, k_2 < k$$

$$\begin{aligned} n &= n_1 + n_2 - 3 \\ k &= k_1 + k_2 + 1 \end{aligned}$$

Since each subpolyhedra have less than k 3-cycles, by induction assumption their triangulations have sizes at most $2n_1 - 7 - k_1$ and $2n_2 - 7 - k_2$ respectively. So the size of triangulation of the polyhedron is at most

$$\begin{aligned} t &\leq (2n_1 - 7 - k_1) + (2n_2 - 7 - k_2) \\ &= 2(n+3) - 14 - (k_1 + k_2) \\ &= 2n - 7 - (k_1 + k_2 + 1) \\ &= 2n - 7 - k \end{aligned}$$

Thus the claim holds. □

Lemma 3.4 shows that our algorithm is no worse than simple pulling, i.e., produces at most $2n - 4 - \Delta$ tetrahedra, while Lemma 3.5 guarantees our algorithm gives at most $2n - 7 - k$ tetrahedra. Note that there are polyhedra having small k but large Δ, or vice versa. The following examples illustrate the four different cases (Fig. 3.1):

(i) A polyhedron with a linear number of 3-cycles but constant degrees (Fig. 3.1(a)). This is a multi-layered triangular prism, made by connecting several copies of a triangular prism (with triangulated faces), that somewhat looks like a long 'tube'. The degrees are constant but there are $O(n)$ 3-cycles. Figure 2.5 illustrates another polyhedron with constant degrees and a linear number of 3-cycles.

(ii) A polyhedron with linear maximum degree but no 3-cycles (Fig. 3.1(b)). This is a 'ball' consisting of a 'North Pole', a 'South Pole' and three 'rings' each containing the same number of vertices. The maximum-degree vertices are the poles whose degrees depend on the number of vertices in each ring, thus linear. There are no 3-cycles.

(iii) A polyhedron with constant degrees and no 3-cycles (Fig. 3.1(c)). This is a multi-layered (non-triangular) prism, a non-triangular version of (i). This may also be regarded as similar to (ii) but having a constant number of vertices per ring, a linear number of rings and without the poles.

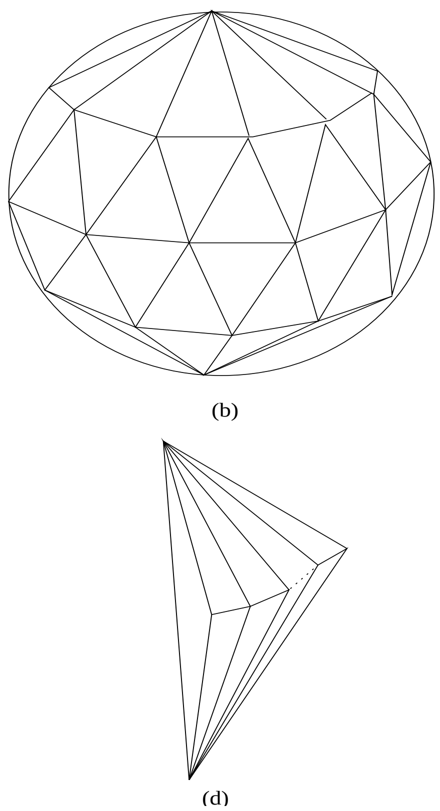

Figure 3.1: Four examples. (a):$k = O(n), \Delta = O(1)$. (b):$k = O(1), \Delta = O(n)$. (c):$k = O(1), \Delta = O(1)$. (d):$k = O(n), \Delta = O(n)$.

(iv) A polyhedron with linear maximum degree and a linear number of 3-cycles (Fig. 3.1(d)). This is a stacked polyhedron with $n-4$ 3-cycles and $\Delta = n-2$; it is in fact a wedge.

Note that for (i) and (iii) the polyhedron needs to be slightly perturbed to become strictly convex. From these examples we know that neither $2n-4-\Delta$ nor $2n-7-k$ always dominates the other term. Thus we have the following:

Lemma 3.6 *The size of triangulation produced by our algorithm is at most min(*$2n-4-\Delta$*,* $2n-7-k$*).*

With all the above lemmas, we are ready to prove the bound. The following theorem gives the analysis of the approximation ratio of our algorithm.

Theorem 3.7 *Algorithm* **CutPull** *gives an approximation ratio of* $2-\Omega(\frac{1}{\sqrt{n}})$.

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. Consider the following two cases:
Case 1. $k = o(n)$. There are at least $2n-4-6k$ faces sharing no edges with 3-cycles. For these faces, at least one interior edge is incident to them for any triangulation. By Lemma 2.8, we have $e_m\Delta \geq n-2-3k = \Theta(n)$, giving $e_m + \Delta \geq \Theta(\sqrt{n})$. Using a proof similar to that of Lemma 3.1, we have [1]

$$r \leq \frac{2n-4-\Delta}{e_m+n-3} \leq \frac{2n-4-\Theta(\sqrt{n})}{n-3} \leq 2-\Omega(\frac{1}{\sqrt{n}})$$

This means even if we apply the pulling heuristic directly to the original polyhedron (without cutting 3-cycles), we still have a bound of $2-\Omega(\frac{1}{\sqrt{n}})$. By Lemma 3.4, partitioning the polyhedron 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.
Case 2. $k = \Theta(n)$. By Lemma 3.5, 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 $c > 0$. □

3.3 Discussions

Some points concerning our algorithm worth further discussions.

3.3.1 Effect of 3-cycles

It is claimed that the pulling heuristic seems to give a minimum triangulation in many cases [38, 3], although its worst-case approximation ratio is only $2-\Omega(\frac{1}{n})$, as noted in Section 2.3, Fig. 2.5. In [4, 5] it is identified that VECSs are obstacles in finding minimum triangulations. We have now identified 3-cycles as another structure that makes the pulling heuristic non-optimal.

Our algorithm can be viewed as a combination of pulling heuristic and algorithms for optimally triangulating stacked polyhedra. In fact, our algorithm gives optimal triangulation for stacked polyhedra. Note that the

[1] We are somewhat abusing the notations $\Theta()$ and $\Omega()$ here.

peeling heuristic (Section 1.4.4.2) also gives optimal triangulation for stacked polyhedra. The 'cut-3-cycle' operation may be viewed as a generalization of peeling.

Although cutting 3-cycles is an important step in our algorithm, it does not always produce minimum triangulations. This also answers a question in [21] in which it was asked whether peeling a degree-3 vertex is always 'safe', i.e., is part of a minimum triangulation. The following construction shows the contrary.

Claim 3.8 *Cutting 3-cycles is not always a correct step in finding minimum triangulations.*

Proof. A counterexample can be constructed using a 'cupola' [5], that is, three VECSs of size m attached to the non-convex sides of a Schönhardt polyhedron. 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 polyhedron remains convex (Fig. 3.2). Now the original bottom face becomes a 3-cycle. The polyhedron has $3m+7$ vertices. If we cut through the 3-cycle, the three main diagonals cannot be used together, because then the Schönhardt polyhedron 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 faces of the Schönhardt polyhedron from inside, we can triangulate the Schönhardt polyhedron by pulling from the added vertex, while the three VECSs are triangulated using their own main diagonals. This gives only $3m+10$ tetrahedra. Therefore cutting 3-cycles does not always produce minimum triangulations. □

The above construction actually shows something stronger: cutting all 3-cycles is not a correct step for any algorithms that achieve an approximation ratio of 4/3 or better (if there is any at all).

3.3.2 Non-simplicial polyhedra

The above algorithm only handles polyhedron with vertices in general position, i.e., the polyhedron is simplicial. The case of non-simplicial polyhedra

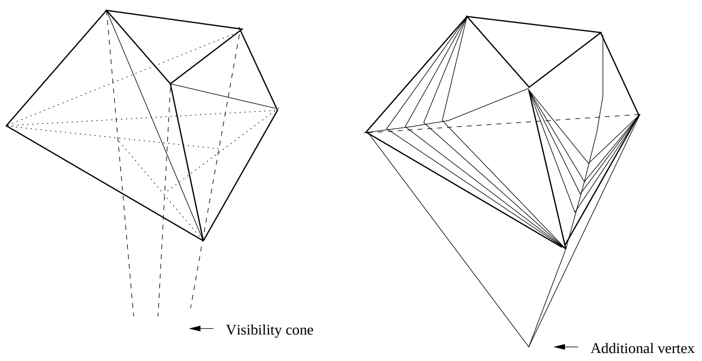

Figure 3.2: A cupola with a 3-cycle.

is quite interesting, but it seems that our algorithm has no easy generalization to handle them. If the surface of the polyhedron is pre-triangulated, there may or may not be a triangulation of the interior compatible with the surface triangulation. An example is a triangular prism: if the three rectangular faces are triangulated in a cyclic order as in Schönhardt's polyhedron, then no triangulation of the prism exists, since any tetrahedron forms a bad intersection with one of these diagonals. It is not known how to determine efficiently whether there is a triangulation compatible with the surface triangulation, and the complexity of this decision problem is yet unknown [6] and may well be NP-complete. If the surface is not pre-triangulated, we can triangulate it in such a way so that the pulling algorithm is possible, but we cannot guarantee the same bound on the approximation ratio, since different surface triangulations may generate different numbers of 3-cycles.

3.3.3 Combinatorial Invariance

Note that finding 3-cycles, cutting them, and pulling on the resulting polyhedra, are all combinatorial invariant operations. Thus our algorithm is also combinatorial invariant, which means that it only takes the surface graph (or other equivalent combinatorial information e.g. the face lattice) of the polyhedron as input. In the next chapter, we shall show that this restriction implies a lower bound on the approximability.

Chapter 4

Lower Bound of Approximation

In the previous chapter, we give a new triangulation algorithm for convex polyhedra with approximation ratio $2 - \Omega(\frac{1}{\sqrt{n}})$. In this chapter, we shall give a lower bound on the approximability of this problem. We show that, given only the combinatorial structure but without the vertex coordinates, no approximation algorithms for finding minimum triangulation of convex polyhedra can have an approximation ratio better than $2 - O(\frac{1}{\sqrt{n}})$. This matches the upper bound given by our algorithm in Chapter 3.

In Section 4.1 we state the key lemma which relates variable-size triangulations to lower bound in approximations. Sections 4.2 and 4.3 contain the constructions to demonstrate the lower bound. In Section 4.4 we discuss some related problems.

4.1 A Non-approximability Lemma

The problem of whether the size of a minimum triangulation of a polyhedron is an invariant of its face lattice (i.e., whether it depends only on the combinatorial structure) was open, until recently Below et al. [3] gave a simple example showing that it is not true when $n > 9$. We shall extend this idea to show that minimum triangulations for two different, but combinatorially equivalent, convex polyhedra can indeed have a very large difference in size. The following simple lemma illustrates that such a difference can be utilized to show a lower bound on approximability, given the combinatorial restriction.

Lemma 4.1 *Suppose two combinatorially equivalent polyhedra A and B have minimum triangulations of sizes t_A and t_B respectively, where without loss of generality assume $t_A \leq t_B$. Then for any algorithm that approximates a minimum triangulation by considering only the combinatorial structure, it cannot give an approximation ratio better than t_B/t_A.*

Proof. Any combinatorial approximation algorithm can only produce the same output for A and B, since from the viewpoint of a combinatorial algorithm they are exactly the same. An approximation algorithm that considers the combinatorial structure only must produce a triangulation no smaller than t_B. Otherwise, this algorithm would produce a triangulation with less than optimal number of tetrahedra when B is the input, which is absurd. Now there exists a polyhedron A with minimum triangulation containing only t_A tetrahedra. When A is the input, the approximation algorithm can only give a triangulation of size t_B. Thus the approximation ratio in this case is bounded by $r \geq t_B/t_A$. □

With this lemma, our task of proving lower bounds is reduced to finding combinatorial equivalent polyhedra with a large difference in the size of minimum triangulations. The following sections are devoted to this task.

4.2 Two Simple Constructions

Before we go straight to our intended lower bound, it is useful to consider some simpler constructions first. The following construction is modified from an example in [3] used to show that minimum triangulations are not an invariant of the face lattice. Consider four vertices a, b, c and d that are almost coplanar in the shape of a rectangle. On top of the line segment cd we put a parabolic arc with vertices $c = p_0, p_1, ..., p_m, p_{m+1} = d$ on it. Below line segment ab we put another parabolic arc and the vertices $a = q_0, q_1, ..., q_m, q_{m+1} = b$. Note that the two arcs are oriented in different directions. We obtain a polyhedron by taking the convex hull of all the $2m + 4$ vertices. See Fig. 4.1. The polyhedron is actually two VECSs attached together.

We can adjust the coordinates of a, b, c and d such that ab may lie above or below cd slightly, without affecting the surface graph, and hence the com-

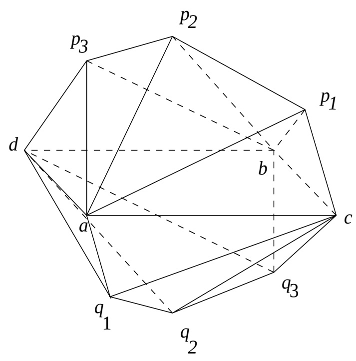

Figure 4.1: Example for 3/2 lower bound.

binatorial structure, of the polyhedron. This allows us to alter the size of minimum triangulations by adjusting the relative positioning of ab and cd. When ab is above cd, the two VECSs do not interlock, and we have a triangulation consisting of the tetrahedra abp_ip_{i+1}, $cdq_iq_{i+1}(0 \leq i \leq m+1)$, and also $abcd$. This gives a triangulation of size $2(m+1)+1 = 2m+3$. However, when ab is below cd, the two VECSs are interlocked, thus at least one of them cannot use the main diagonal for triangulation. Therefore there are at least m interior edges (Lemma 2.1), giving a triangulation of size at least $m + (2m+4) - 3 = 3m+1$ (Lemma 1.5). It follows from Lemma 4.1 that our example gives a bound of $r \geq \frac{3m+1}{2m+3} \approx 3/2$.

The paper [3] contained another example we can modify to give a better bound. First consider a triangular prism with three VECSs attached to the rectangular faces. The top triangular face is smaller than the bottom face; this is not entirely necessary and is just for clarity in the drawings (Fig. 4.2). Then we twist the top face by an angle α to make the prism non-convex, like an Schönhardt polyhedron. When $\alpha = \pi/3$, the three main diagonals touch each other. When $\alpha = \pi/3 + \epsilon$ for small $\epsilon > 0$, the three VECSs are interlocked. Thus at most one of the three main diagonals can be used, giving $2m$ tetrahedra to each of the two blocked VECS, and $m+1$ tetrahedra for the unblocked one. In total there are at least $5m+1$ tetrahedra. But when $\alpha = -\epsilon$, we have twisted in the other direction, making the triangular prism slightly convex. The three VECSs do not block each other (actually they are not VECS at all now since the main diagonal does not go through its interior), thus can be triangulated using only $3(m+1)$ tetrahedra. The

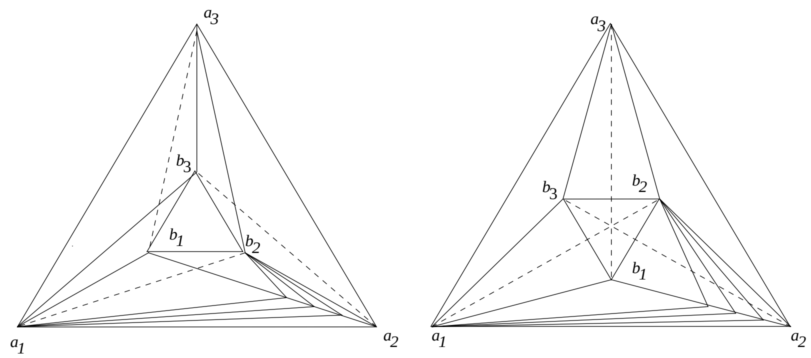

Figure 4.2: Example for 5/3 lower bound, top view. (a): $\alpha = 0$; (b) $\alpha = \pi/3$.

remaining convex triangular prism has six vertices that takes four tetrahedra to triangulate. This gives a total of $3m + 7$ tetrahedra. By Lemma 4.1 we immediately have a non-approximability example of ratio roughly 5/3. Actually, the pulling heuristic does achieve a ratio 5/3 for this problem since $\Delta \approx n/3$, $r \leq \frac{2n-4-n/3}{n-3} \approx 5/3$.

Note that the three VECSs we placed here are not completely disjoint, thus Lemma 2.3 cannot be applied directly; however this can easily be accounted for to achieve the same result.

The above two examples suggest that increasing the number of interlocked wedges may lead to a tighter lower bound, and may eventually reach the upper bound we proved in Chapter 3. However, straightforward extensions such as using an m-sided twisted prism with VECSs attached do not work. In fact, placing a lot of wedges with all of them intersecting while maintaining convexity requires careful analysis. We show in the next section how to achieve this.

4.3 A $2 - O(\frac{1}{\sqrt{n}})$ Lower Bound

We could improve the bound on the approximation ratio if we could make more VECSs interlocking each other. However the constructions in the previous section do not seem to generalize to more VECSs. In this section we present another construction that generalizes to an arbitrary number of VECSs. We then show that this gives the lower bound we desire.

We construct two polyhedra, P1 and P2, each consisting of m thin wedges,

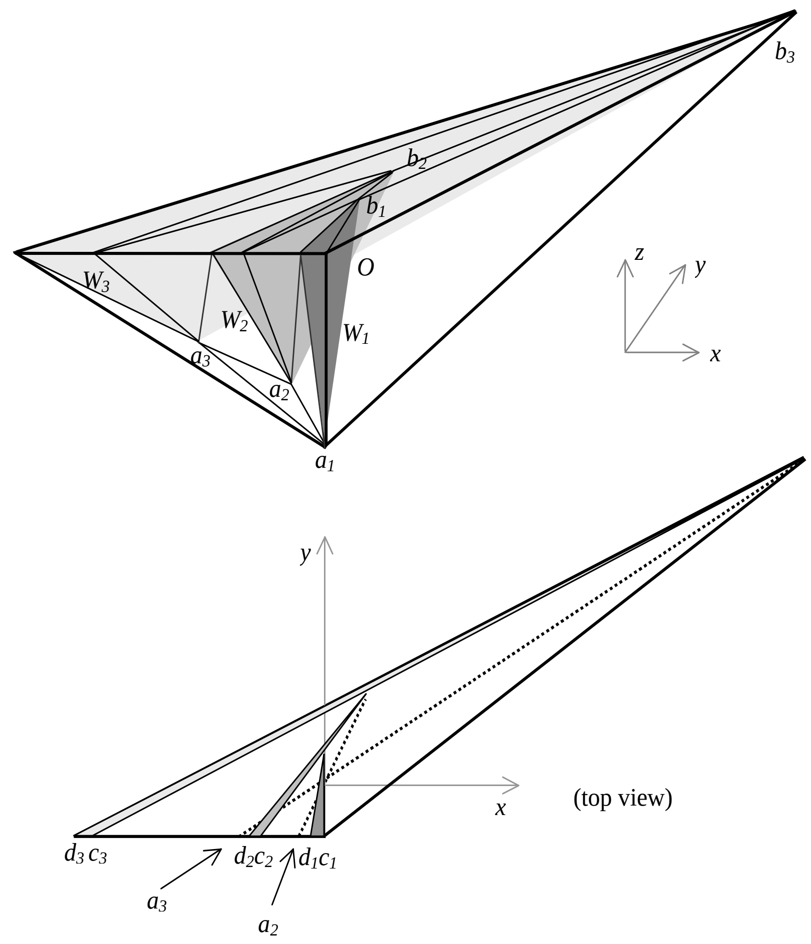

Figure 4.3: The placement of three wedges. q^i's are not shown.

each wedge being a VECS of size m. The two polyhedra 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 size of triangulation. P2 is constructed similarly but the wedges are not interlocked, and thus all m main diagonals can be present in a triangulation, resulting in a much smaller size of triangulation.

4.3.1 The Detailed Construction

Figure 4.3 shows a rough idea of the construction of P1 and P2. We first arrange m wedges in an xyz-coordinate system. Wedge W_k ($1 \le k \le m$) 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 to W_k have been constructed, W_{k+1} is constructed as follows (Fig. 4.3 also shows a top view from the positive z-axis):

Pick a_{k+1} so that its x-coordinate is smaller than that of 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 $(\infty,\ 0,\ 1)$.

Next, m points $q_k^1, q_k^2, ..., q_k^m$ are created between each interval (c_k, d_k) and edges $a_k q_k^i, b_k q_k^i$ are added so that they form m VECSs each 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 along the edges of the polyhedron so that the polyhedron is convex, while keeping the orientations of all the main diagonals unchanged, i.e., 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 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 polyhedron will not be changed if δ is 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 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. 4.3 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 4.2 *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 $b_k q_k^i q_k^{i+1}$ appear on the surface of P1 and P2.
(iv) For P1, all wedges are interlocked. For P2, all wedges do not intersect one another.

4.3.2 Size of the Triangulations

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

Lemma 4.3 *Any minimum triangulation of P1 contains at least $2m^2 - m + 1$ tetrahedra.*

Proof. By Lemmas 2.2 and 2.3, 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 be 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 4.4 *Any minimum triangulation of P2 contains at most $m^2 + 8m - 8$ tetrahedra.*

Proof. We triangulate each wedge by using the main diagonal of its corresponding VECS, producing $m+1$ tetrahedra each. Now we have to triangulate

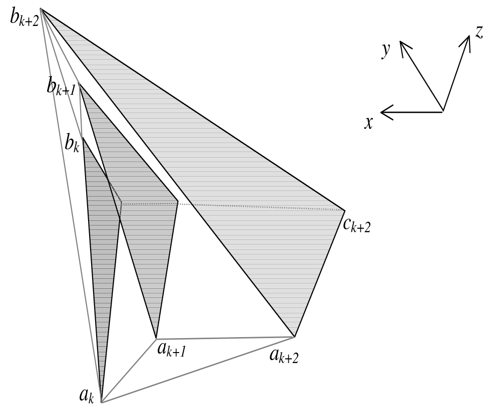

Figure 4.4: The space between three wedges, viewing from 'the back'. Thickness of each wedge is not shown.

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. 4.4). 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 $W_1 W_2 W_3$, then $W_1 W_3 W_4$ 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 $a_1 b_1 c_1 b_m$ and $a_m b_m d_m a_1$ 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 four 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 $m(m+1) + 4(m-1) + 3(m-2) + 2 = m^2 + 8m - 8$. $\square$

Theorem 4.5 *Any approximation algorithm for finding minimum triangulations of convex polyhedra, provided with only the combinatorial structure, cannot have approximation ratio better than* $2 - O(\frac{1}{\sqrt{n}})$.

Proof. In the above discussion we constructed two combinatorially equivalent polyhedra P1 and P2. By Lemmas 4.3 and 4.4, 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. By Lemma 4.1 with $t_A \leq n + \Theta(\sqrt{n})$ and $t_B \geq 2n - \Theta(\sqrt{n})$, we have [1]

$$r \geq \frac{t_B}{t_A} \geq \frac{2n - \Theta(\sqrt{n})}{n + \Theta(\sqrt{n})} \geq 2 - O(\frac{1}{\sqrt{n}})$$

□

4.4 Discussions

Unlike most other non-approximability results for NP-hard problems (see e.g. [27, 16]), our result does not depend on the P/NP question: this lower bound for approximation is valid no matter P=NP or not. It is because the bound is derived by observing the ambiguity in the size of triangulations due to combinatorial restrictions. However, this lower bound proof may not hold if we allow non-combinatorial information to be used. It is of interest to see whether this bound remains valid without the combinatorial restriction, in which case we have to assume P $\neq$ NP.

It was noted that the size of minimum triangulation is not an invariant of the face lattice. This opens the problem of 'how much' can the size of minimum triangulations be varied with the same face lattice. The polyhedra P1 and P2 we constructed demonstrates that the difference can be as much as twice. Our combinatorial invariant triangulation algorithm actually implies that this is the best possible. Thus we have:

Corollary 4.6 *For two combinatorially equivalent polyhedra, their sizes of minimum triangulations can differ by at most a factor of* $2 - \Omega(\frac{1}{\sqrt{n}})$*, and this is tight.*

[1] Again we are abusing the notations $\Theta()$ and $O()$ here.

Chapter 5

Some Special Cases

The bounds we gave in Chapters 3 and 4, though tight, is only a small improvement compared with previous algorithms. Asymptotically, it still tends to 2 as n tends to infinity, which is the same as the pulling heuristic. It is natural to consider whether there are special polyhedra with approximation ratio smaller than 2. In this section we try to identify certain classes of polyhedra that can have better approximation ratios using our new algorithm. The two cases that are considered are those with constant degrees, and those without VECSs and 3-cycles. They are described in Sections 5.1 and 5.2 respectively.

5.1 Polyhedra with Constant Degrees

The NP-hardness proof of minimum polyhedron triangulation [4, 5] involves constructing polyhedra having maximum degrees as a function of n. It is interesting to consider the case when the maximum degree is bounded by a constant. This occurs frequently for randomly-generated polyhedra, or in Delaunay triangulation / mesh generation. In this section we give improved analysis using our new approximation algorithm for this problem.

5.1.1 The Upper Bound

We first prove an upper bound on the approximation ratio, using Lemmas 2.9 and 3.5:

Theorem 5.1 *When Δ is constant, the* **CutPull** *algorithm gives an approximation ratio bounded above by $2 - \frac{2}{\Delta+1} - \Omega(\frac{1}{n})$.*

Proof. Suppose there are k 3-cycles in the polyhedron. By Lemma 3.5 we have

$$r \leq \frac{2n-7-k}{n-3+e_m} < \frac{2n-1-k}{n+e_m}$$

By Lemma 2.9, $e_m\Delta + 3k \geq n-2$, thus $e_m\Delta + (\frac{\Delta+1}{2})k \geq n-2$ (since $\Delta \geq 6$ for $n > 12$, Proposition 1.4). Rearranging gives

$$k \geq \frac{2}{\Delta+1}n - \frac{4}{\Delta+1} - \frac{2\Delta}{\Delta+1}e_m$$

Substituting into the previous inequality gives

$$\begin{aligned} r &\leq \frac{2n-1-\frac{2}{\Delta+1}n+\frac{4}{\Delta+1}+\frac{2\Delta}{\Delta+1}e_m}{n+e_m} \\ &= \frac{(2-\frac{2}{\Delta+1})n+(2-\frac{2}{\Delta+1})e_m-(1-\frac{4}{\Delta+1})}{n+e_m} \\ &= 2-\frac{2}{\Delta+1}-\frac{1-\frac{4}{\Delta+1}}{n+e_m} \\ &= 2-\frac{2}{\Delta+1}-\Omega(\frac{1}{n}) \end{aligned}$$

since $e_m = O(n)$ (Section 1.4.4). □

This bound improves substantially from 2, especially when Δ is small. For example when $\Delta = 6$, $r \leq 12/7 \approx 1.714$; when $\Delta = 7$, $r \leq 7/4 = 1.75$.

In fact, we can prove further, along the lines of Theorem 3.7, that for general (non-constant) Δ, the approximation ratio is bounded by $r \leq 2 - \Omega(\frac{1}{\Delta}) - \Omega(\frac{\Delta}{n})$. This is more general than both the bound here and that in Theorem 3.7; the worst case is when $\Delta = \Theta(\sqrt{n})$ in which the bound reduces to the $2 - \Omega(\frac{1}{\sqrt{n}})$ bound.

5.1.2 The Lower Bound

It is proved in Chapter 4 that no combinatorial algorithms can give an approximation ratio better than $2 - O(\frac{1}{\sqrt{n}})$ for the minimum triangulation problem. The proof is based on constructing two polyhedra P1 and P2 with the same combinatorial structure but having different sizes in minimum triangulations. In this subsection we prove similar results for the constant degree case. This

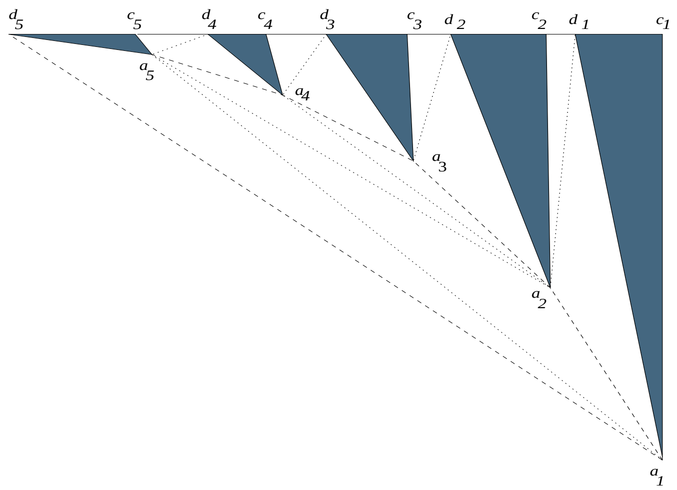

Figure 5.1: The wedges (shaded) with $m = 5$, showing the vertical plane only.

shows that our upper bound proved in the previous subsection is asymptotically tight when only combinatorial information is considered.

Similar to Section 4.3, we construct two polyhedra Q1 and Q2, that consist of a set of wedges, which are long VECSs. The polyhedra consist of m wedges, each wedge having size s which is a constant. Thus the number of vertices $n = m(s+4)$. The placement of wedges and the construction of Q1 and Q2 are almost identical to those of P1 and P2 in Section 4.3, with one exception: since we need constant degrees, the vertices a_1 and b_1 in the original construction (Fig. 4.3) do not satisfy this requirement. We remove the degeneracy of the vertical and horizontal planes in another way: we use zig-zags paths, e.g. $a_1 a_n a_2 a_{n-1}...$, $b_1 b_n b_2 b_{n-1}...$ (See Fig. 5.1). Such placement is always possible by using sufficiently small perturbations. The maximum degree $\Delta = s + 7$ (e.g. a_2 in Fig. 5.1) is thus constant.

We now bound the size of the triangulations.

Lemma 5.2 *Any minimum triangulation of Q1 has at least* $(\frac{2\Delta-14}{\Delta-3})n - \Delta + 8$ *tetrahedra, while any minimum triangulation of Q2 has at most* $(\frac{\Delta+1}{\Delta-3})n$ *tetrahedra.*

Proof. For Q1, at most one main diagonal of these m wedges can be present in any triangulation (Lemma 4.3), and each of the remaining wedges needs

$2s$ tetrahedra to triangulate. Thus

$$\begin{aligned} t_{Q1} &\geq (m-1)(2s)+(s+1) \\ &= 2(\frac{n}{s+4})s - s + 1 \\ &= 2(\frac{\Delta-7}{\Delta-3})n - (\Delta-7)+1 \\ &= (\frac{2\Delta-14}{\Delta-3})n - \Delta + 8 \end{aligned}$$

For Q2, all wedges can use their respective main diagonals to triangulate simultaneously. The remaining non-convex region can be triangulated into $4(m-1)+3(m-2)+2 = 7m-8$ tetrahedra, using the 'shielding' argument same as in Lemma 4.4. Note that for this to work the a_i's and b_i's have to be 'zig-zagged' in a matching manner. So

$$\begin{aligned} t_{Q2} &\leq m(s+1)+7m-8 \\ &= (\frac{n}{s+4})(s+1+7)-8 \\ &= (\frac{n}{\Delta-3})(\Delta+1)-8 \\ &< (\frac{\Delta+1}{\Delta-3})n \end{aligned}$$

□

Lemma 5.2 immediately gives the following theorem:

Theorem 5.3 *When Δ is constant, any algorithm that only considers the combinatorial structure of the polyhedron cannot give an approximation ratio better than $2 - O(\frac{1}{\Delta+1}) - O(\frac{1}{n})$.*

Proof. By Lemmas 5.2 and 4.1, the approximation ratio

$$\begin{aligned} r &\geq \frac{(\frac{2\Delta-14}{\Delta-3})n - \Delta + 8}{(\frac{\Delta+1}{\Delta-3})n} = \frac{n(2\Delta-14)-(\Delta-8)(\Delta-3)}{n(\Delta+1)} \\ &= \frac{2\Delta-14}{\Delta+1} - O(\frac{1}{n}) = 2 - O(\frac{1}{\Delta+1}) - O(\frac{1}{n}) \end{aligned}$$

□

Note that this proof immediately generalizes to a lower bound proof of $r \geq 2 - O(\frac{1}{\Delta}) - O(\frac{\Delta}{n})$ for non-constant Δ. Thus the bound mentioned at the end of the previous subsection is also tight up to combinatorial considerations.

5.2 Polyhedra Without Degree-4 Vertices and 3-cycles

From the results in [4] and Chapter 3, it can be seen that the major problems in finding minimum triangulations appear in 3-cycles and VECSs. In this section we first analyze the special case in which the polyhedra concerned have no 3-cycles and degree-4 vertices (thus no VECSs). Note that the non-existence of 3-cycles implies that there are no degree-3 vertices (Observation 2.6), and thus all vertices have degrees at least five. We show that in this case the approximation ratio of the **CutPull** algorithm is at most $3/2$, better than the general case ratio $2 - \Omega(\frac{1}{\sqrt{n}})$. Moreover our analysis is tight: we construct polyhedra having an approximation ratio no better than $3/2 - \epsilon$ using **CutPull** for any $\epsilon > 0$. We then consider the case when 3-cycles are present.

Empirically, it has been observed that 3-cycles are not very common in polyhedra, in particular those not induced by degree-3 vertices. Moreover, polyhedra usually have some degree-3 and degree-4 vertices, although not too many. However, our results still have the following significance:

(i) as far as we know this is one of the very few classes of polyhedra that is known to have approximation ratio at most $2-\epsilon$ for constant $\epsilon > 0$. For example, stacked polyhedra (Section 2.2) can be triangulated optimally in linear time, or the 'k-opt polyhedra' [40].

(ii) the existence of 3-cycles and degree-4 vertices can be checked in linear time, in contrast to k-opt polyhedra where no algorithm is known to check whether a polyhedron is k-opt.

(iii) they may arise as intermediate polyhedra in the processing of other triangulation algorithms, e.g. peeling (Section 1.4.4.2). If there are few degree-4 vertices, we might peel those degree-4 vertices first and may leave behind a polyhedron with degrees at least five (but note that new degree-4 vertices may appear in peeling).

(iv) Certain classes of polyhedra, such as prisms, antiprisms, etc. [17] have no 3-cycles and degrees at least five (provided that the coplanar points

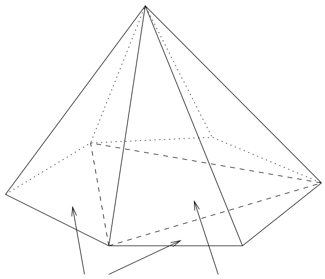

Figure 5.2: Ear and internal triangles.

are perturbed so that the faces are suitably triangulated, and if necessary with simple modification/replication).

We classify all vertices of a polyhedron P into two types (w.r.t. a particular triangulation): a vertex is called *type-I* if some interior edge is directly incident to it; otherwise it is called *type-II*. For any vertex v, we define the *neighborhood* $N(v)$ of v to be the set of vertices directly connected to v on the surface graph, i.e. $N(v) = \{u | (u, v) \in \text{surface edges of } P\}$. $N(v)$ forms a 3-dimensional polygon. Consider any triangulation of the polygon $N(v)$. (Note that this is slightly different from the definition of 'cap' or 'dome' [9] in that a triangulation of $N(v)$ may not yield a convex patch of triangular faces.) Triangles with two edges on the polygon $N(v)$ are called *'ear' triangles*, and all others are called *'internal' triangles*. See Fig. 5.2.

We present some observations about type-II vertices in the lemma below:

Lemma 5.4 *Suppose v is a type-II vertex of degree d in a polyhedron P with respect to a triangulation of P.*
(i) All tetrahedra incident to v form a triangulation of the region bounded by the 3-D polygon $N(v)$ and the faces of P around v. There are $d-2$ tetrahedra in this part of the triangulation. Their bases triangulate the polygon $N(v)$.
(ii) For any triangulation of $N(v)$, if $d \geq 5$ and v is not lying on any 3-cycles, there is at least one vertex in $N(v)$ having two or more incident interior edges. The triangulation of $N(v)$ consists of at least two 'ear' triangles and at least one 'internal' triangle.

Proof. (i) Consider a face vv_1v_2 on P incident to v. There is a tetrahedron incident to this face. For this tetrahedron, its fourth vertex v_3 must be in $N(v)$

(otherwise, vv_3 is an interior edge, contradicting v being a type-II vertex). Thus $v_1v_2v_3$ form a triangle within $N(v)$, i.e., triangulating the 3-D polygon $N(v)$. Now this triangle divides the polygon $N(v)$ into two parts (one part may be empty if v_3 is next to either v_1 or v_2). The two parts are blocked by the tetrahedron $vv_1v_2v_3$, so each part is then recursively triangulated as argued above. Thus the region between the 3-D polygon $N(v)$ and v must be triangulated by a set of tetrahedra such that their bases triangulate $N(v)$. Since triangulating this polygon takes $d-2$ triangles, there are $d-2$ tetrahedra in this part of the triangulation.
(ii) The existence of two ears is well-known [28]. There can be at most $\lfloor d/2 \rfloor$ ear triangles, leaving at least $d - 2 - \lfloor d/2 \rfloor \geq 1$ internal triangles. The existence of vertices with multiple incident interior edges follows from the existence of internal triangles. □

Lemma 5.5 *For a polyhedron P having no 3-cycles and with vertex degrees at least five, there are at least $4n/3 - 3$ tetrahedra in any triangulation of P.*

Proof. Suppose we have n_1 type-I vertices and n_2 type-II vertices, and $n_1 + n_2 = n$. We consider the two cases, each using a different method to bound the size of triangulation:

Case 1: $n_1 \geq \alpha n$, where α $(0 \leq \alpha \leq 1)$ is a constant to be specified later. We want to count the number of interior edge endpoints incident to these vertices (each interior edge having two endpoints). By definition, for each type-I vertex there is at least one interior edge endpoint incident to it. This gives n_1 edge endpoints. In addition, for each of the n_2 type-II vertices, there is at least one type-I vertex in its neighborhood that has two or more edge endpoints incident to it (Lemma 5.4). But the previous step did not count the extra endpoints (only one endpoint was counted for each type-I vertex). Thus there are at least n_2 additional edge endpoints, if all of them are distinct. Note that these must not be surface edges of the polyhedron (Observation 2.5). It can be shown (see Claim 5.6 appearing immediately after this lemma) that at most two type-II vertices share such an additional endpoint; the worst case is as in Fig. 5.4 where two type-II vertices (v_1 and v_9) sharing a type-I vertex (v_3) that only has two interior edge endpoints. Thus at least $n_2/2$ edge endpoints are added. Since each interior edge has two endpoints to be counted,

$$e_m \geq \frac{1}{2}(n_1 + \frac{n_2}{2}) = \frac{n_1}{2} + \frac{n_2}{4} = \frac{n + n_1}{4} \geq \frac{(1+\alpha)n}{4}$$

Thus

$$t = e_m + n - 3 \geq \frac{(1+\alpha)n}{4} + n - 3 = \frac{(5+\alpha)n}{4} - 3$$

Case 2: $n_1 < \alpha n$. Then $n_2 > (1-\alpha)n$. For each type-II vertex v, all tetrahedra incident to v constitute a triangulation of $N(v)$ (Lemma 5.4)(Fig. 5.2). Consider any triangulation of the 3-D polygon $N(v)$, with each triangle corresponding to a tetrahedron having v as a vertex. We count the number of tetrahedra incident to the type-II $N(v)$'s. All 'internal' tetrahedra of a type-II vertex v will not be counted by other type-II vertices (since the other three vertices of the tetrahedron are type-I). The 'ear' tetrahedra may be counted twice. For example in Fig. 5.4 tetrahedra $v_1v_2v_3v_4$ and $v_1v_3v_5v_6$ are 'ear' tetrahedra of a type-II vertex v_1, but the tetrahedron $v_1v_2v_3v_4$ (resp. $v_1v_3v_5v_6$) may also be counted by v_2 (resp. v_6) if v_2 (resp. v_6) are type-II. It cannot be counted more than twice since the other two vertices of the tetrahedron are type-I. Since $d \geq 5$, there is at least one 'internal' tetrahedron and at least two 'ear' tetrahedra, giving a total of at least two tetrahedra (each 'ear' counted as 0.5 for this vertex to avoid double counting). Thus the total number of tetrahedra incident to these type-II vertices is at least $2n_2 \geq 2(1-\alpha)n > 2(1-\alpha)n - 3$.

Considering both cases, the number of tetrahedra is at least $\min(\frac{(5+\alpha)n}{4} - 3, 2(1-\alpha)n - 3)$. It is easy to show that this is maximized when $\alpha = 1/3$, thus $t \geq 4n/3 - 3$. □

Claim 5.6 *There are at least $n_2/2$ extra interior edge endpoints.*

Proof. First note that any triangulation of $N(v)$ of a type-II vertex v gives at least one 'internal' face (i.e. the internal triangle). Such a face may be shared by at most two type-II-vertex's $N(v)$'s, on both sides of the face (see Fig. 5.4). Thus n_2 type-II vertices produce at least $n_2/2$ such interior faces. Note that for such a sharing to occur, the polyhedron would be divided into two parts (the two 'gaps' in Fig. 5.4), which are disjoint except touching at a single vertex.

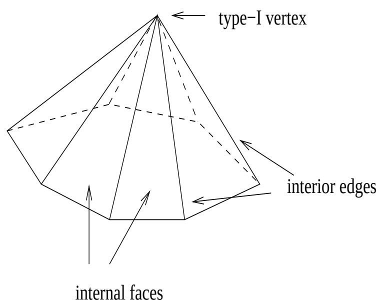

Figure 5.3: A loop of 7 adjacent internal faces giving 7 interior edges, 6 of which are 'extra'.

Now, observe that each such internal face gives at least one extra interior edge endpoint. Consider all the f internal faces incident to a vertex, they induce at least f interior edge endpoints. (If these faces are not adjacent (sharing an edge), there are two edges per face; if they share an edge along the way, there is a one-to-one mapping between internal faces and internal edges.) All these f endpoints are 'extra', unless the faces wrap back to a loop (Fig. 5.3). But in this case it is impossible for two type-II $N(v)$'s to share any of these faces, since as noted above sharing makes the polyhedron separated into two parts, so the loop cannot be completed. In this case every loop of $f \geq 3$ faces produces $f - 1 \geq f/2$ extra interior edge endpoints.

Thus in any case at least $n_2/2$ extra interior edge endpoints are present.

□

The bound in Lemma 5.5 is tight (up to a constant additive factor) as shown by the lemma below:

Lemma 5.7 *There exist polyhedra, without 3-cycles and degree-4 vertices, whose sizes of minimum triangulation are at most* $4n/3 + 8/3$.

Proof. The construction proceeds as follows.

Step 1. First consider the nine vertices $v_1, ..., v_9$ as in Fig. 5.4; $N(v_1) = \{v_2, v_3, v_4, v_5, v_6\}$, $N(v_9) = \{v_3, v_4, v_5, v_7, v_8\}$. $N(v_1)$ and $N(v_9)$ share the triangle $v_3v_4v_5$. This may be regarded as a (non-convex) polyhedron with v_3v_4, v_3v_5 being the nonconvex edges, that will become interior edges when we further construct the polyhedron. We now have a (partial) triangulation

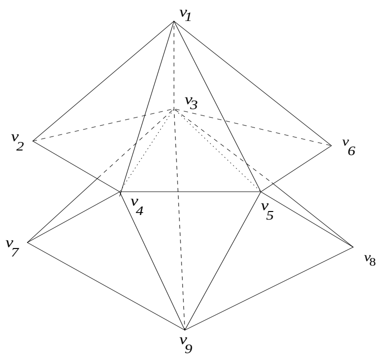

Figure 5.4: Two type-II vertices sharing an internal triangle. Also Step 1 of the construction.

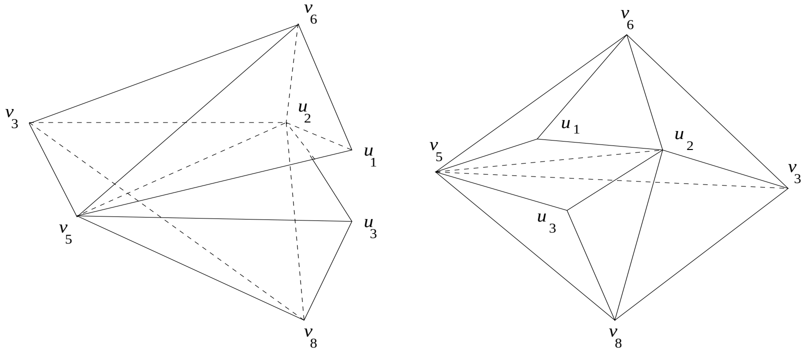

Figure 5.5: Step 2, showing 3-D view (left) and view from right (right).

consisting of six tetrahedra $v_1v_2v_3v_4, v_1v_3v_4v_5,\ v_1v_3v_5v_6,\ v_9v_7v_3v_4,\ v_9v_3v_4v_5$ and $v_9v_3v_5v_8$.

Step 2. We patch each of the two non-convex 'gaps' by the following structure. For the right-side gap, we add three vertices u_1, u_2 and u_3 and form the polyhedron as in Fig. 5.5, producing a new non-convex gap. The u_i's are placed carefully so that convexity is maintained (except at the new non-convex gap). The new region can be triangulated by the tetrahedra $v_5v_6u_1u_2, v_3v_5v_6u_2, u_2u_3v_5v_8$ and $v_3v_5u_2v_8$. The left-side gap is treated similarly. Each application of step 2 replaces the two old gaps by two new ones, using six more vertices and eight more tetrahedra. Note that the new gap and the old gap share a vertex. Step 2 is recursively applied to the new gaps. This step can be applied any number of times. We can keep the maximum degree of the polyhedron constant by alternating which vertex the original

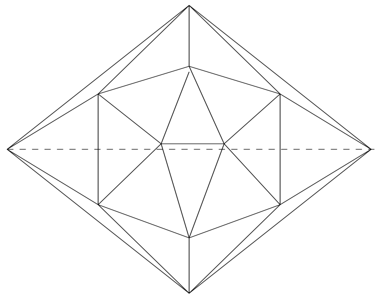

Figure 5.6: Final patch for Step 3. The two triangles at the back (as indicated by the dashed line) are to be attached to the gap.

gap touches with the new gap.

Step 3. Finally, we patch each of the two gaps by the structure shown in Fig. 5.6. This structure has 12 vertices (8 of them are new vertices when patched), all are of degree at least five, and needs at most $2n - 4 - \Delta = 2(12) - 9 = 15$ additional tetrahedra to triangulate (by pulling). The structure is patched so that the final polyhedron is convex.

It can be seen that this convex polyhedron has no 3-cycles, and all vertices have degrees at least five.

Note that we constructed a triangulation of this polyhedron along with the above steps. This may not be a minimum triangulation, but a minimum triangulation must have same or smaller size. If Step 2 is applied p times, then $n = 9 + 6p + 16$ and $t \leq 6 + 8p + 30$, giving $t \leq 4n/3 + 8/3$. □

The tight bound on the size of triangulations gives a tight bound on the approximation ratio of **CutPull**:

Theorem 5.8 *The approximation ratio of* **CutPull** *algorithm for polyhedron without 3-cycles and all vertices having degrees at least five is at most 3/2, and this is tight.*

Proof. The approximation ratio follows from Lemmas 3.5 and 5.5: $r \leq \frac{2n-7}{4n/3-3} < 3/2$. That the bound is tight follows from the constructed polyhedron in Lemma 5.7, having constant degrees, no 3-cycles, and $t \leq 4n/3 + 8/3$. Thus for those polyhedra, **CutPull** gives $r \geq \frac{2n-4-\Delta}{4n/3+8/3} = 3/2 - \epsilon$ where $\epsilon = \Theta(\frac{1}{n})$ tends to 0 as n tends to infinity. □

With the presence of 3-cycles (but still without degree-3 and degree-4 vertices), the argument in Lemma 5.5 works for vertices not lying on any 3-cycles. Thus if there are n' vertices not lying on 3-cycles, we have $t \geq 4n'/3-3$. Note that we have $n' \geq n - 3k$, thus with Lemma 3.5 we have:

Theorem 5.9 *For polyhedra with k 3-cycles and having degrees at least five, the approximation ratio $r \leq \frac{2n-7-k}{max(n-3,4n/3-4k-3)}$. In particular, it can be shown to be less than $2 - \frac{1}{12} = 1.9166...$ for any k.*

Chapter 6

Conclusion

Triangulation in three dimensions is an important problem in Computational Geometry. It has practical applications and is also of theoretical interest. Recently, the problem of finding a minimum triangulation of a convex polyhedron was shown to be NP-hard. In this thesis, we studied the problem of approximating a minimum triangulation for convex polyhedra.

We discovered new relationships between the size of minimum triangulations of a convex polyhedron and its combinatorial parameters. This is proved by simple counting techniques. Using this, we gave a new approximation algorithm for finding minimum triangulations of convex polyhedra with approximation ratio $2 - \Omega(\frac{1}{\sqrt{n}})$. We then showed a matching $2 - O(\frac{1}{\sqrt{n}})$ lower bound on the approximability of the minimum triangulation problem, by observing polyhedra with the same combinatorial structure can have large differences in their sizes of minimum triangulations. Thus there is no better constant-ratio approximation algorithm without considering the actual coordinates of the vertices, i.e., our algorithm is optimal in this sense. We also gave improved approximation ratios for some special classes of polyhedra, including polyhedra with bounded maximum degrees, and polyhedra without 3-cycles and degree-4 vertices.

The results in this thesis point out some open questions. Possible future research directions include:

- Are there algorithms with better approximation ratios ($r < 2 - \epsilon$ where $\epsilon > 0$ is constant, does not depend on n) when the actual coordinatizations are taken into account, instead of just looking at the combinatorial structure? We suspect that may not be possible.

- Where is the exact place of the 'borderline' of the NP-completeness of the minimum polyhedron triangulation problem? For example, when Δ is constant, it is not known whether the problem remains to be NP-complete. It is also unclear whether the bounds we gave in Section 5.1 can be improved.

- Are there other kinds of polyhedra that have better approximation ratios using our algorithm? Are there other algorithms for other types of polyhedra? Can we identify some polyhedra that can be triangulated optimally or near-optimally in polynomial time?

- More specifically, for the case without 3-cycles and degree-4 vertices, are there better algorithms for this class of polyhedra? Can it be optimally triangulated in polynomial time? How important is the effect of 3-cycles? It seems quite possible to improve the $2-\frac{1}{12}$ bound for polyhedra having no degree-3 and degree-4 vertices, and our results may also be generalized to polyhedra having few (but nonzero) degree-4 vertices. Our results also show that even when both VECSs and 3-cycles are absent, pulling may not give a minimum triangulation. Are there more restricted classes of polyhedra that can be optimally triangulated by **CutPull** or pulling?

- What about non-simplicial polyhedra? Are there good approximation algorithms for them? It seems nontrivial to extend results for simplicial polyhedra to them; see the discussion in Section 4.4.

- What about maximum triangulations? Are they of the same complexity to obtain as minimum triangulations? O'Rourke [30] raises this problem and points out some motivations.

- What about higher dimensions? It is not at all clear how can we generalize our algorithm to higher dimensions, although pulling can be applied in higher dimensions. A lower bound of $n-d$ simplices is known for triangulating any d-dimensional n-vertex polyhedron. Can we improve this in a way analogous to the 3-D case? Corollary 4.6 also raises the question of finding an example in higher dimensions where the difference in triangulation size between two combinatorial equivalent polyhedra grows unbounded.

This research looked at the old triangulation algorithm from a new perspective, using approximation ratios instead of counting the actual number of tetrahedra. We combine heuristics for polyhedron triangulation and give completely new techniques for proving stronger lower bounds on the size of triangulation. We hope that this piece of research will generate interest in this area and we look for better results in the future.

Bibliography

[1] Avis D. and ElGindy H., Triangulating Point Sets in Space, *Discrete and Computational Geometry* 2 (1987), 99–111.

[2] Bagemihl F., On Indecomposable Polyhedra, *American Mathematical Monthly* 55(7) (1948), 411–413.

[3] Below A., Brehm U., De Loera J. A., and Richter-Gebert J., Minimal Simplicial Dissections and Triangulations of Convex 3-Polytopes, *Discrete and Computational Geometry* 24(1) (2000), 35–48.

[4] Below A., De Loera J. A., 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), 65–66.

[5] Below A., De Loera J. A., and Richter-Gebert J., The Complexity of Finding Small Triangulations of Convex 3-Polytopes, available at http://www.math.ucdavis.edu/~deloera/RECENT_WORK/recent.html

[6] Bern M., Compatible Tetrahedralizations, *Proceedings of 9th Annual ACM Symposium on Computational Geometry* (1993), 281–288.

[7] Bern M. and Eppstein D., Mesh Generation and Optimal Triangulation, in *Computing in Euclidean Geometry* (Hwang and Du eds.), World Scientific, Singapore (1995).

[8] Chazelle B., Convex Partitions of Polyhedra: a Lower Bound and Worst-case Optimal Algorithm, *SIAM Journal on Computing* 13(3) (1984), 488–507.

[9] Chazelle B. and Palios L., Triangulating a Non-convex Polytope, *Discrete and Computational Geometry* 5 (1990), 505–526.

[10] Chazelle B., Triangulating a Simple Polygon in Linear Time, *Discrete and Computational Geometry* 6 (1991), 485–524.

[11] Chazelle B. and Shouraboura N., Bounds on the Size of Tetrahedralizations, *Proceedings of 10th Annual ACM Symposium on Computational Geometry* (1994), 231–239.

[12] Cheng S. W., Dey T. K., Edelsbrunner H., Facello M. A. and Teng S. H., Sliver Exudation, *Journal of ACM* 47 (2000), 883–904.

[13] Chiba N. and Nishizeki T., Arboricity and Subgraph Listing Algorithms, *SIAM Journal on Computing* 14(1) (1985), 210–223.

[14] Chin F. Y. L., Fung S. P. Y. and Wang C. A., Approximation for Minimum Triangulation of Convex Polyhedra, *Proceedings of 12th Annual ACM-SIAM Symposium on Discrete Algorithms* (2001), 128–137.

[15] Chin F. Y. L., Fung S. P. Y. and Wang C. A., Approximation for Minimum Triangulations of Simplicial Convex 3-Polytopes, *Discrete and Computational Geometry*, to appear.

[16] Crescenzi P. and Kann V., A Compendium of NP Optimization Problems, at http://www.nada.kth.se/theory/compendium/wwwcompendium.html

[17] Cromwell P. R., *Polyhedra*, Cambridge University Press (1997).

[18] De Floriani L., Puppo E. and Magillo P., Applications of Computational Geometry to Geographic Information Systems, in *Handbook of Computational Geometry* (Sack and Urrutia eds.), Elsevier, North-Holland (2000).

[19] Diestel R., *Graph Theory*, Graduate Text in Mathematics 173, Springer-Verlag, New Tork (2000).

[20] Edelsbrunner H., Preparata F. P., and West D. B., Tetrahedrizing Point Sets in Three Dimensions, *Journal of Symbolic Computation* 10 (1990), 335–347.

[21] Eppstein D., http://www.ics.uci.edu/~eppstein/projects/tetra/

[22] Goodman J. E. and Pach J., Cell Decompositions of Polytopes by Bending, *Israel Journal of Mathematics* 64 (1988), 129–138.

[23] Henk M., Richter-Gebert J. and Ziegler G. M., Basic Properties of Convex Polytopes, in *Handbook of Discrete and Computational Geometry* (Goodman J. E. and O'Rourke J. eds.), CRC Press (1997).

[24] Kalai G., Polytope Skeleton and Paths, in *Handbook of Discrete and Computational Geometry* (Goodman J. E. and O'Rourke J. eds.), CRC Press (1997).

[25] 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).

[26] Lennes N. J., Theorems on the Simple Finite Polygon and Polyhedron, *American Journal of Mathematics* 33 (1911), 37–62.

[27] Mayr E. W., Prömel H. J. and Steger A. eds., Lectures on Proof Verification and Approximation Algorithms, *Lecture Notes in Computer Science* 1367, Springer-Verlag (1998).

[28] Meisters G. H., Polygons Have Ears, *American Mathematical Monthly* 82 (1975), 648–651.

[29] Mitchell S. A. and Vavasis S. A., Quality Mesh Generation in Three Dimensions, *Proceedings of 8th ACM Symposium on Computational Geometry* (1992), 212–221.

[30] O'Rourke J., Computational Geometry Column 40, *International Journal of Computational Geometry and Applications* 10(6) (2000), 649–651.

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

[32] Preparata F. P. and Shamos M. I., *Computational Geometry: an Introduction*, Springer-Verlag, New York (1985).

[33] Richter-Gebert J., Finding Small Triangulations of Polytope Boundaries is Hard, *Discrete and Computational Geometry* 24 (2000), 503–517.

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

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

[36] Schönhardt E., Uber die Zerlegung von Dreieckspolyedern in Tetraeder, *Math. Annalen* 98 (1928), 309–312.

[37] Seidel R., The Nature and Meaning of Perturbations in Geometric Computing, *Discrete and Computational Geometry* 19 (1998), 1–17.

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

[39] Strang G. and Fix G. J., *An Analysis of the Finite Element Method*, Prentice Hall, Englewood Cliffs, New Jersey (1973).

[40] Wang C. A. and Yang B., Optimal Tetrahedralizations of Some Convex Polyhedra, *16th European Workshop on Computational Geometry* (2000), 5 pages.

[41] Wang C. A. and Yang B., Tetrahedralization of Two Nested Convex Polyhedra, *Proceedings of 6th Annual International Computing and Combinatorics Conference* (2000), 291–298.

[42] Wang C. A. and Yang B., Minimal Tetrahedralizations of a Class of Polyhedra, *Proceedings of 12th Canadian Conference on Computational Geometry* (2000), 81–90.

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