Hausdorff Approximation of 3D Convex Polytopes

Mario A. Lopez
Department of Mathematics
University of Denver
Denver, CO 80208, U.S.A.
mlopez@cs.du.edu

Shlomo Reisner
Department of Mathematics
University of Haifa
Haifa 31905, Israel
reisner@math.haifa.ac.il

Abstract

Let P be a convex polytope in $\mathbb{R}^d$, $d = 3$ or 2, with n vertices. We present linear time algorithms for approximating P by simpler polytopes. For instance, one such algorithm selects $k < n$ vertices of P whose convex hull is the approximating polytope. The rate of approximation, in the Hausdorff distance sense, is best possible in the worst case. An analogous algorithm, where the role of vertices is taken by facets, is presented.

1 Introduction

Let P be a convex polytope in $\mathbb{R}^3$ (or a convex polygon in $\mathbb{R}^2$), having n vertices. Given a positive integer $k < n$ (k must be big enough but there is no constraint on the relative size of k with respect to n) we present an $O(n)$ time algorithm that selects k of the vertices of P, in such a way that the convex hull Q of the selected vertices satisfies

$$(1) \qquad d_H(P, Q) \leq \frac{cR}{k} \qquad \left(\frac{cR}{k^2} \text{ in the 2D case} \right),$$

where c is a fixed constant and R is the minimal radius of a ball containing P. Here, d_H denotes the Hausdorff distance. The significance of estimate (1) is in the fact that, up to the value of the constant c, this is the best possible worst-case estimate, as examples of polytopes P that approximate closely the Euclidean ball show (see [10] or [17], for example).

In a completely analogous way (that we shall not elaborate upon, except for hints), a similar algorithm produces a polytope W containing a polytope P with n facets, W has $k < n$ facets that are contained in k of the hyperplanes supporting facets of P and an inequality similar to (1) is satisfied (with Q replaced by W). We call the first type of approximation *inner approximation*, and the second one, *outer approximation*.

The approach presented here is similar in nature and in the methods to the one of [15], where analogous results are obtained for the symmetric distance (i.e. volume difference). There is, however, a basic difference between the approaches. In [15] the algorithm for symmetric difference approximation could be extended to dimensions higher than 3 in polynomial (though not linear) time (the power increasing with the dimension d). The algorithms presented here do not extend polynomially to dimensions higher than 3. What we do prove here is the *existence* in $\mathbb{R}^d$ of a polytope $Q \subset P$ with k vertices selected from the n vertices of P, such that

$$(2) \qquad d_H(P, Q) \leq \frac{cR}{k^{2/(d-1)}}\,.$$

This is the best possible worst case bound on the rate of such approximation. The existence of such an approximation rate for any convex body in $\mathbb{R}^d$ (with constant c independent of d) is well known (see, e.g. [2, 8]), but the fact that, for a polytope P, it can be achieved with vertices(Q) $\subset$ vertices(P) is a new result. Let us mention here the recent algorithm of Chan [4], which solves a related problem for arbitrary d. Given P and $\varepsilon = d_H(P, Q)$, the algorithm returns a polytope Q whose k vertices are selected from those of P and which satisfies $k \le c(d)/\varepsilon^{(d-1)/2}$, where the constant $c(d)$ depends on the dimension d. For dimensions 2 and 3, our algorithms take k and P as input and return a polytope Q of size k, containing or contained in P, which satisfies (2). They are also simpler and easier to implement.

Methods for the approximation of general convex bodies in $\mathbb{R}^d$ by convex polytopes can be found in the mathematical literature (see [11, 12] for surveys, as well as [9]). Refinements of these methods into algorithms are, to mention two, [14] and [15]. Algorithmic results which can be interpreted as approximation algorithms were obtained in [6] where a randomized algorithm is given which, for 3D polytopes, runs in $O(k^2 n \log k \cdot \log(n/k))$ time in the worst case (n and k as above). In [1], a deterministic algorithm based on similar ideas is presented, but its running time is significantly higher. These are slower and harder to implement than our algorithms, and their precision rate (in the Hausdorff distance sense) is (almost) a multiple of the best one for the specific polytope. Algorithms for 2D Hausdorff approximation are presented in [16] which give precision almost equal to the best possible for the *specific* polygon P involved. These algorithms run in $O(n \log n)$ or $O(n)$ time (inner or outer case).

We now present some terminology and notation. A *convex polytope* P in the Euclidean space $\mathbb{R}^d$ is the convex hull of a finite set of points. In the present paper we always assume that P has a non-empty interior. An extreme point of P is called *a vertex*. The set of vertices of P is vert(P). Equivalently, P is a convex polytope in $\mathbb{R}^d$ if and only if it is a bounded set (with non-empty interior) which is the intersection of a finite set of half-spaces bounded by hyperplanes. The $(d-1)$-dimensional faces of a d-dimensional polytope are its *facets*. For a set A we denote by $|A|$ the cardinality of A. The convex hull of a set M is conv(M). Let B_2^d be the Euclidean unit ball in $\mathbb{R}^d$, centered at 0. For bounded subsets A and B of $\mathbb{R}^d$ the Hausdorff distance between A and B is: $d_H(A, B) = \inf\{\varepsilon > 0\,;\, A \subset B + \varepsilon B_2^d \text{ and } B \subset A + \varepsilon B_2^d\}$, where $+$ denotes the Minkowski sum.

2 Inner Approximation

The following lemma appears in [20, Lemma 3.3]. (The assumption made there, that P is simplicial, is not needed for the result to be true, and is made only to simplify the proof).

Lemma 2.1 *There exist constants $c_0, c_1 > 0$ such that for any $\varepsilon > 0$ and any positive integers d, n with $n > (c_0)^d/\varepsilon$ the following holds: Let P be a convex polytope with n vertices in $\mathbb{R}^d$ contained in the Euclidean unit ball B_2^d of $\mathbb{R}^d$. For a vertex v of P we denote by h_v the distance from v to the convex hull of all the vertices of P other than v. Then the set*

$$A_\varepsilon = \left\{ v \;\middle|\; v \text{ is a vertex of } P \text{ and } h_v \le \frac{c_1}{(\varepsilon n)^{\frac{2}{d-1}}} \right\}$$

has at least $(1-\varepsilon)n$ elements.

Note that h_v is the Hausdorff distance $d_H(P, \text{conv}(\text{vert}(P) \setminus \{v\}))$, and that members of A_ε are good candidates for removal, when trying to approximate P with a polytope or polygon with fewer vertices. We formalize this idea as follows

Definition 2.1 *Let c_0 and c_1 satisfy the requirements of Lemma 2.1, let $\varepsilon > 0$ be given, and let P be a convex polytope in $\mathbb{R}^d$ which is contained in B_2^d. We say that a vertex v of P is* (ε, c_1)-useful *(or* ε-useful *if c_1 is accepted as given) if $v \in A_\varepsilon$, i.e., if $h_v \leq c_1/(\varepsilon n)^{2/(d-1)}$.*

In the sequel, if ε is clear from the context, we shall simply say that v is useful.

2.1 Dimension 3

In this section we present an algorithm for approximating a convex polytope P in $\mathbb{R}^3$, having n vertices, by a convex polytope Q contained in P, which has k vertices constituting a subset of the vertices of P. The algorithm runs in $O(n)$ time. The constant involved in the $O(n)$ estimate is strongly influenced by the desired degree of precision of the approximation.

We assume that the polytope P is given with its "convex hull", i.e., that all the adjacency relations between vertices, edges and facets of P are given. Computation of the convex hull of P from its vertices (in 2D and 3D) requires $O(n \log n)$ time (see [19, 5, 3], for example).

The algorithm operates by repeatedly removing a carefully chosen vertex from the current polytope P_i. Thus, the basic step selects a vertex v of P_i, and replaces P_i by $\text{conv}(\text{vert}(P_i) \setminus \{v\})$, the convex hull of all vertices of P_i except v. When repeated $n-k$ times, vertex removal yields a polytope of the desired size. We refer to the part removed, $P_i \setminus \text{conv}(\text{vert}(P_i) \setminus \{v\})$, as the *cap* of v.

We say that distinct vertices u and v of P_i are *adjacent* if they are connected by an edge of P_i. Let $N_i(v)$ denote the set of vertices adjacent to v in P_i. Thus, the degree of v, denoted by $\deg(v)$ is simply $|N_i(v)|$. A facet g of $\text{conv}(N_i(v))$ is said to be *visible* from v if $\text{conv}(N_i(v))$ and v are on opposite sides of the supporting plane of g (or if $\text{conv}(N_i(v))$ is 2-dimensional, in which case it is considered a visible face).

Lemma 2.2 *Each facet of conv(vert(P_i) $\setminus \{v\}$) is of one of the following three kinds:*

a) Facets f of P_i not incident to v.

b) Facets g which are the convex hull of the vertices, other than v, of a facet f of P_i such that f contains v and has more than three vertices (note that $g \subset f$).

c) Facets h of $\text{conv}(N_i(v))$ which are visible from v.

The convex hull of $\text{conv}(\text{vert}(P_i) \setminus \{v\})$ can be computed easily. Each facet g of type b) can be computed in constant time by removing v from the corresponding facet f of P_i. This is done by replacing the two edges of f incident on v with a single edge that connects the two vertices of f that are adjacent to v. Of course, there are at most $r(v) := \deg(v)$ such facets. Computing the convex hull of $N_i(v)$ can be done in $O(r(v) \log r(v))$ time. All facets of type c) can then be found in $O(r(v))$ additional time from $\text{conv}(N_i(v))$. Thus the removal of a vertex v from P_i can be done in $O(r(v) \log r(v))$ time.

The proof of the following lemma makes a simple use of Euler's relation and is presented in [15, Lemma 2.2].

Lemma 2.3 *Let P be a polytope in $\mathbb{R}^3$ with n vertices. For any $0 < \beta < 1$ there are at least βn vertices of P with degree less than $\frac{3(2-\beta)}{(1-\beta)}$.*

Lemma 2.4 *Let P be a polytope in $\mathbb{R}^3$ with $n > 6$ vertices, such that $P \subset B_2^3$. Then for any $6 < r < n$ and $\varepsilon < \frac{r-6}{r-3}$, if $n \geq c_0^3/\varepsilon$ then there exist at least σn ε-useful vertices of degree less than r, where $\sigma = \frac{r-6}{r-3} - \varepsilon$ (note that $\sigma > 0$).*

Proof. Let S_r denote the set of vertices of P with degree less than r, and S_u, the set of vertices that are ε-useful. For $r > 6$ Lemma 2.3 implies $|S_r| \geq \frac{r-6}{r-3}n$. Similarly, from Lemma 2.1 we know that $|S_u| \geq (1-\varepsilon)n$. Now, since $n \geq |S_r \cup S_u| = |S_r| + |S_u| - |S_r \cap S_u|$, it follows that $|S_r \cap S_u| \geq (\frac{r-6}{r-3} - \varepsilon)n = \sigma n$, thus the claim is established. ■

Note that the required relation between r and ε which is stated in Lemma 2.4, can be rewritten to give the same σ provided $0 < \varepsilon < 1$ and $r > \frac{3(2-\varepsilon)}{1-\varepsilon}$.

We now describe a deterministic algorithm that computes the approximating polytope in linear time. The idea is to repeatedly remove as many vertices as possible without having to update the caps of vertices adjacent to the removed vertices. To this end, we repeatedly identify and eliminate a set of useful vertices which constitute an independent set in the 1-skeleton of the current polytope. During this process we are essentially computing a hierarchical representation of the input polytope, an idea proposed in [13, 7] (see also Section 7.10 of [18]), in the context of point location, polytope separation and other problems.

We start with an auxiliary algorithm that identifies and removes *one* independent set of vertices. Assume, initially, that k satisfies the following precondition:

$$k > \left(1 + \frac{\varepsilon}{r} - \frac{r-6}{r(r-3)}\right) n \tag{3}$$

(with ε and r are as above). Clearly, the removal of a vertex v does not alter the cap of a vertex w if v and w are not adjacent (this can be concluded with the help of Lemma 2.2). Thus, we identify a subset $R \subset \text{vert}(P)$ of size $n-k$ such that for all $v \in R$: v is useful, $\deg(v) < r$, and v is not adjacent to w for any other $w \in R$. The following algorithm, based on this simple idea, finds an approximating polytope Q, provided that k satisfies (3) above.

Auxiliary(P, n, k, ε, r)

Input: a polytope P ($P \subset B_2^3$) with n vertices, $c_0^3/\varepsilon \leq k < n$, k satisfies (3).

Output: a polytope $Q \subset P$ with k vertices.

1. Compute a list L of all useful vertices of P of degree $< r$
2. Mark all vertices of L as clean
3. Let $P_n = P$
4. **for** $i \leftarrow n$ **downto** $k+1$ **do**
5. find the next clean vertex v in L
6. mark v and all vertices adjacent to it in L as dirty
7. Let $P_{i-1} \leftarrow \text{conv}(\text{vert}(P_i) \setminus \{v\})$
8. **return** $Q = P_k$

Analysis. For a vertex v of P_i, the time needed to compute the convex hull of $\text{vert}(P_i) \setminus \{v\}$ is, as remarked above, $O(r(v) \log r(v))$, where $r(v)$ is the degree of v. Thus Step 1 requires

$O(nr\log r)$ time. Since each vertex in the list is scanned at most once, the total time spent in Step 5 is $O(n)$. Each execution of Steps 6 and 7 requires $O(r)$ and $O(r\log r)$ time, respectively. Thus, the total time is bounded by $O(nr\log r)$.

Clearly, if $n-k$ is too large then R, as specified above, may not exist. The existence of $n-k$ clean vertices is deduced from (3) by establishing a lower bound on the maximum possible value of $n-k$ as follows: By Lemma 2.4, at least $(\frac{r-6}{r-3}-\varepsilon)n$ vertices are both useful and have small degree. Since the removal of a (clean) vertex produces at most r dirty vertices and we want to remove $n-k$ (clean) vertices, to be on the safe side, we require

$$\left(\frac{r-6}{r-3}-\varepsilon\right)n > (n-k)r \tag{4}$$

This is equivalent to the condition (3) above.

Let γ be any positive number less than $\frac{(r-6)-\varepsilon(r-3)}{r(r-3)}$ and $\delta<\gamma$. In the following algorithm we shall be interested in the quantity $1/(\gamma-\delta)$, note that we can get this quantity to be, for example, as small as $\frac{2r(r-3)}{(r-6)-\varepsilon(r-3)}$.

Approx(P, n, k, ε, r)
Input: a polytope P ($P\subset B_2^3$) with n vertices, $\max(c_0^3/\varepsilon,\ 1/(\gamma-\delta))\le k<n$.
Output: a polytope $Q\subset P$ with k vertices.
1. $n_0\leftarrow n$
2. $Q_0\leftarrow P$
3. $i\leftarrow 0$
4. **repeat**
5. $\quad i\leftarrow i+1$
6. $\quad n_i\leftarrow \max(k,\lceil(1-\gamma)n_{i-1}\rceil)$
7. $\quad Q_i\leftarrow \text{Auxiliary}(Q_{i-1},n_{i-1},n_i,\varepsilon,r)$
8. **until** $n_i=k$
9. **return** $Q=Q_i$

Analysis. As long as n_i, the number of vertices in the polytope Q_i, computed in the i-th iteration, has not reached the value k, we have $(1-\gamma)^i n\le n_i\le(1-\delta)^i n$. In fact, the left hand side of the last inequality is clear, while the right hand side is derived as follows:

$$n_i=\lceil(1-\gamma)n_{i-1}\rceil\le(1-\delta)n_{i-1}\,.$$

This is because $(1-\delta)n_{i-1}-(1-\gamma)n_{i-1}\ge(\gamma-\delta)k\ge 1$.

Now, for $i_0=\left\lceil\frac{\log(n/k)}{\log(1/(1-\delta))}\right\rceil$ we get $(1-\delta)^{i_0}n\le k$, hence there are at most i_0 steps of the iteration. The $(i+1)$-th step (steps 5-7) requires $O(n_i r\log r)$ time. Summing up, the total time is bounded by

$$r\log r\sum_{i=0}^{i_0-1}n_i<r\log r\cdot n\sum_{i=0}^{\infty}(1-\delta)^i<nr\log r\cdot\frac{1}{\delta}\,.$$

Thus, by the choice of δ, the running time is $O(\frac{r^2\log r}{1-\varepsilon}n)$.

We now estimate the Hausdorff distance $d_H(P,Q)$. At the i-th step of the algorithm, ε-useful vertices of the n_{i-1}-polytope Q_{i-1} are removed. The caps of these vertices are disjoint. Thus, by the definition of usefulness, we get

$$d_H(Q_{i-1}, Q_i) \leq \frac{c_1}{\varepsilon n_{i-1}}\,. \tag{5}$$

Let i_1 be the biggest i such that $n_i > k$ (clearly $i_1 < i_0$). We conclude from (5):

$$d_H(P,Q) \leq \frac{c_1}{\varepsilon}\left(\sum_{i=0}^{i_1}\frac{1}{n_i}\right) \leq \frac{c_1}{\varepsilon}\left(\frac{1}{n}\sum_{i=0}^{i_1}\frac{1}{(1-\gamma)^i}\right)$$

$$< \frac{c_1}{\varepsilon}\left(\frac{1}{\gamma(1-\gamma)^{i_1}n}\right)\,.$$

Now, since $n_i \leq (1-\gamma)n_{i-1}+1$, we get

$$(1-\gamma)^{i_1}n = (1-\gamma)^{i_1-1}(1-\gamma)n \geq (1-\gamma)^{i_1-1}(n_1-1) \geq (1-\gamma)^{i_1-2}(n_2-1-(1-\gamma))\ldots$$

$$> n_{i_1} - \sum_{i=0}^{\infty}(1-\gamma)^i \geq k - \frac{1}{\gamma}\,.$$

Thus we get

$$d_H(P,Q) \leq \frac{c_1}{\varepsilon}\left(\frac{1}{\gamma\left(k-\frac{1}{\gamma}\right)}\right) \leq \frac{c_2(\varepsilon,r)}{k}\,, \tag{6}$$

for some constant $c_2(\varepsilon,r)$ that depends on ε and r alone and, for fixed r, grows like ε^{-1} as ε tends to zero.

We summarize the above in the following theorem (by scaling we avoid the assumption that P is contained in the unit ball).

Theorem 2.1 *Let $r > 6$ and $0 < \varepsilon < \frac{r-6}{r-3}$ be user defined constants. Then there exist constants $c(\varepsilon,r)$ and $\alpha(\varepsilon,r)$ and an algorithm that runs in $O(\frac{r^2\log r}{1-\varepsilon}n)$ time, which, given a convex polytope P in $\mathbb{R}^3$ with n vertices and k such that $\alpha(\varepsilon,r) \leq k \leq n$, finds a convex polytope $Q \subset P$ with k vertices, chosen from the original vertices of P, such that*

$$d_H(P,Q) \leq \frac{c(\varepsilon,r)}{k}R\,.$$

R here is the minimal radius of a Euclidean ball containing P. For fixed r, $c(\varepsilon,r)$ and $\alpha(\varepsilon,r)$ grow like ε^{-1} as ε tends to zero.

2.2 Dimension 2

We now consider the problem of approximating a 2-dimensional polytope P. The algorithm of Section 2.1 adapts easily to 2D in a simpler way than in 3D, as all vertices have degree 2. The result is a linear time approximation algorithm for polygons. Using Lemma 2.1 for $d = 2$ and the algorithmic ideas used before, we get the following theorem.

Theorem 2.2 *Let $0 < \varepsilon < 1$ be an arbitrary constant. There exists an algorithm that runs in $O(\frac{n}{1-\varepsilon})$ time which, given a convex polygon P in $\mathbb{R}^2$ with n vertices and k such that $\alpha(\varepsilon) \leq k < n$, finds a convex polygon $Q \subset P$ with k vertices, chosen from the original vertices of P, such that*

$$d_H(P, Q) \leq \frac{c(\varepsilon)}{k^2} R\,.$$

R here is the minimal radius of a Euclidean ball containing P. $c(\varepsilon)$ and $\alpha(\varepsilon)$ grow, respectively, like ε^{-2} and ε^{-1} as ε tends to zero.

The error estimate in Theorem 2.2 is, again, of best possible worst-case order in k. We note here that in [16] we presented efficient algorithms which provide approximations of convex polygons with precision within a multiple of the *best rate for the specific polygon treated*, by any constant greater than 1. The present algorithm does less, in general, in terms of the precision. It is, however, simpler to implement and, unlike [16], uses the vertices of the original polygon.

2.3 Higher dimensions

For dimension $d > 3$ the best worst-case order of the error in inner or outer Hausdorff approximation is $cRk^{-2/(d-1)}$ (k is the number of vertices). Unlike in the volume difference approximation (see [15]), our Hausdorff algorithm can not be generalized to these dimensions, even by increasing the complexity. This is because of the existence, in dimension higher than 3, of neighborly polytopes. In such polytopes all the vertices are neighbors of each other, thus we can not find useful vertices whose sets of neighbors are disjoint. In our volume-difference algorithm, this only increased the complexity. Here, this problem breaks the proof of correctness of the algorithm.

Theorem 2.1, and its proof, beside their algorithmic content, present also a theoretical result: convex polytopes in $\mathbb{R}^2$ and $\mathbb{R}^3$ can be approximated, in the Hausdorff distance sense, by polytopes with fewer vertices inscribed in them, so that a best possible worst-case order of the error is achieved and *the vertices of the approximating polytope constitute a subset of the vertices of the approximated polytope*. It turns out that this last feature is available in higher dimensions as well. This is the subject of the reminder of this section.

Lemma 2.5 *There exist constants c_1, $c_2 > 0$ such that for any $0 < \varepsilon < 1$, any integers $d > 0$ and $n > (c_2)^d/\varepsilon$ and any convex polytope $P \subset B_2^d$ having vertex set V with $|V| = n$, there exists a subset $U \subset V$, $|U| \geq (1-\varepsilon)n$ such that*

$$d_H(P, \mathrm{conv}(V \setminus U)) < \frac{c_1}{(\varepsilon n)^{2/(d-1)}}\,.$$

Proof. We may assume for simplicity (w.l.o.g) that P is simplicial. It is a known fact (see [2, 8] and also [20]) that for $m \geq c_0^{(d-1)/2}$, with $c_0 > 0$ an appropriate constant, we can find m points $\{z_1, \dots, z_m\}$ on the boundary of P (in fact, P can be replaced here by any convex body contained in B_2^d), so that $Q = \mathrm{conv}(\{z_1, \dots, z_m\})$ satisfies

$$d_H(P, Q) < \frac{c_0}{m^{2/(d-1)}}\,.$$

Let us take $m = \lceil \varepsilon n/d \rceil$. Then, for a constant c_2 that satisfies $c_2 \geq (dc_0^{(d-1)/2})^{1/d}$ for all d, we get: If $n > c_2^d/\varepsilon$ then $m > c_0^{(d-1)/2}$ as required above. Hence, for such m and for Q as above, we have

$$d_H(P,Q) < \frac{c_0 d^{2/(d-1)}}{(\varepsilon n)^{2/(d-1)}} < \frac{c_1}{(\varepsilon n)^{2/(d-1)}}$$

for some constant c_1. We may assume, again for simplicity, that all the vertices of Q lie in the relative interior of facets of P. We claim that for a subset $U \subset V$, $Q \subset \text{conv}(V \setminus U)$ if and only if, for every $x \in U$, no vertex z of Q lies in a facet of P adjacent to x. This is, in fact, the case that all the vertices of Q lie in facets of P that are also facets of $\text{conv}(V \setminus U)$.

That is, for $Z \subset V$, $Q \not\subset \text{conv}(V \setminus Z)$ if and only if there is a vertex z of Q that lies in a facet of P that contains a vertex $x \in Z$. Now, if

$$Z = \{x \in V \,;\, \exists z_j \in \text{vert}(Q),\ z_j \text{ is in a facet of } P \text{ containing } x\},$$

then

$$|Z| \leq \sum_{j=1}^{m} |\{x \in V \,;\, x \text{ is in a facet of } P \text{ containing } z_j\}| = d \cdot m = \varepsilon n$$

Thus, taking $U = V \setminus Z$, we have $|U| \geq (1-\varepsilon)n$ and $Q \subset \text{conv}(V \setminus U)$. Hence

$$d_H(P, \text{conv}(V \setminus U)) \leq d_H(P,Q) < \frac{c_1}{(\varepsilon n)^{2/(d-1)}}.$$

■

By setting $k = \lceil \varepsilon n \rceil$, we can rewrite Lemma 2.5 in the following way (we underlined the sentence that provides new information)

Corollary 2.6 *There exist constants $c_1, c_2 > 0$ such that for any positive integer d, any integers $n > k > (c_2)^d$ and any convex polytope P in $\mathbb{R}^d$ with n vertices, there exists a polytope Q, which is the convex hull of k of the vertices of P, that satisfies*

$$d_H(P,Q) < \frac{c_1 R}{k^{2/(d-1)}}.$$

R above is the minimal radius of a Euclidean ball containing P.

3 Outer approximation and constant estimation

The algorithm presented earlier works in a complete analogy for outer approximation. Here we consider a convex polytope P in $\mathbb{R}^3$, having n *facets* and the result of the algorithm is a convex polytope W containing P, which has k facets ($k < n$) and approximates P, in the Hausdorff sense, with best possible order of precision in general, in terms of k. The new algorithms are again based on a mathematical result from [20] whose 3-dimensional case is dual to Lemma 2.1. Using this result instead of Lemma 2.1 we derive an algorithm which is "dual" to the one of Section 2. In this algorithm the roles of vertices and facets are interchanged and the removal of a facet is accomplished by replacing P with the intersection of the remaining facet half-spaces. (Note that facets whose removal results in an unbounded polyhedron are, by definition, not useful.) This results in the following theorem which is analogous to Theorem 2.1. An outer approximation theorem analogous to theorem 2.2 in dimension 2 is true as well.

Theorem 3.1 *Let $r > 6$ and $0 < \varepsilon < \frac{r-6}{r-3}$ be user defined constants. Then there exist constants $c(\varepsilon, r)$ and $\alpha(\varepsilon, r)$ and an algorithm that runs in $O(\frac{r^2 \log r}{1-\varepsilon} n)$ time, which, given a convex polytope P in $\mathbb{R}^3$ with n facets and k such that $\alpha(\varepsilon, r) \leq k \leq n$, finds a convex polytope $W \supset P$ with k facets, all of whose facets are contained in planes supporting facets of P, such that*

$$d_H(P, W) \leq \frac{c(\varepsilon, r)}{k} R\,.$$

R here is the minimal radius of a Euclidean ball containing P. For fixed r, $c(\varepsilon, r)$ and $\alpha(\varepsilon, r)$ grow like ε^{-1} as ε tends to zero.

Remark A result analogous to Corollary 2.6, with *vertices* replaced by *facets*, is true as well (with a dual proof).

The constants The constants c_0, c_1 of Lemma 2.1 and their counterparts for outer approximation, are important in order to decide for which values of n (number of vertices or facets) the algorithms are guaranteed to work. These constants also determine the precision rate of the algorithms. Moreover, both algorithms require verifying whether a vertex (or facet) is useful. The definition of usefulness involves the constant c_1. Hence we need estimates on the constants for the implementation of the algorithms. Upper bounds for these values have been found in [15] (the notations c_0 and c_1 were used there in a different sense). Basically the estimate was based on estimating the cardinality of an ε-net on the Euclidean sphere in $\mathbb{R}^3$. It follows from those computations that an upper bound of 183 for both c_0^3 and c_1 will work (in [15] the value of of the upper bound for c_1 was larger, due to the need for additional volume estimate, which is not needed here).

References

[1] H. Brönnimann and M. Goodrich, Almost optimal set covers in finite VC-dimensions. Discr. Comput. Geom. 14 (1995), 463–479.

[2] E. M. Bronshtein and L. D. Ivanov, The approximation of convex sets by polyhedra. Siberian Math. J. 16 (1975) (English translation), 852–853.

[3] T. M. Chan, Optimal output-sensitive convex hull algorithms in two and three dimensions. Discr. Comput. Geom. 16 (1996), 361–368.

[4] T. M. Chan, Faster core-set constructions and data-stream algorithms in fixed dimensions. Comput. Geom. 35 (2006) 20-35.

[5] B. Chazelle and J. Matoušek, Derandomizing an output-sensitive convex hull algorithm in three dimensions. Comput. Geom. 5 (1995), 27–32.

[6] K. L. Clarkson, Algorithms for polytope covering and approximation. Proceedings of the 3rd Workshop on Algorithms and Data Structures. Lecture Notes in Computer Science 709. Springer-Verlag 1993, 246–252.

[7] D. P. Dobkin and D. G. Kirkpatrick, Determining the separation of preprocessed polyhedra - a unified approach. Proc. 17-th Internat. Colloq. Automata Lang. Program. Lecture Notes in Computer Science 443. Springer-Verlag 1990, 400–413.

[8] R. M. Dudley, Metric entropy of some classes of sets with differentiable boundaries. J. Approximation Th. 10 (1974), 227–236.

[9] Y. Gordon, M. Meyer and S. Reisner, Constructing a polytope to approximate a convex body. Geometriae Dedicata 57 (1995), 217–222.

[10] Y. Gordon, S. Reisner and C. Schütt, Umbrellas and polytopal approximation of the Euclidean ball. J. Approximation Th. 90 (1997), 9-22.

[11] P. M. Gruber, Aspects of approximation of convex bodies. Handbook of Convex Geometry, edited by P. M. Gruber and J. M. Wills. North-Holland 1993, 319–345.

[12] P. M. Gruber, Approximation by convex polytopes. Polytopes: Abstract, Convex and Computational, edited by T. Bistriczky et al. Kluwer Academic Publishers 1994, 173–203.

[13] D. G. Kirkpatrick, Optimal search in planar subdivisions. SIAM J. Comput. 12 (1983). 28–35.

[14] M. A. Lopez and S. Reisner, Algorithms for polyhedral approximation of multidimensional ellipsoids. J. Algorithms 33 (1999), 140–165.

[15] M. A. Lopez and S. Reisner, Linear time approximation of 3D convex polytopes. Comput. Geom. 23 (2002), 291–301.

[16] M. A. Lopez and S. Reisner, Hausdorff approximation of convex polygons, Comput. Geom. 32 (2005), 139-158.

[17] P. Mankiewicz and C. Schütt, On the Delone triangulation numbers. J. Approximation Th. 111 (2001), 139-142.

[18] J. O'Rourke. Computational geometry in C, 2nd Edition. Cambrigde University Press 1998.

[19] F. P. Preparata and S. J. Hong, Convex hulls of finite sets of points in two and three dimensions. Comm. ACM 20 (1977), 87–93.

[20] S. Reisner, C. Schütt and E. Werner, Dropping a vertex or a facet from a convex polytope. Forum Math. 13 (2001), 359–378.