On the Complexity of Halfspace Volume Queries

Erik D. Demaine* Jeff Erickson† Stefan Langerman‡

Abstract

Given a polyhedron P in $\mathbb{R}^d$ with n vertices, a halfspace volume query asks for the volume of $P \cap H$ for a given halfspace H. We show that, for $d \geq 3$, such queries can require $\Omega(n)$ operations even if the polyhedron P is convex and can be preprocessed arbitrarily.

1 Introduction

A typical *range query problem* can be formulated as follows: Preprocess a set S of n points in $\mathbb{R}^d$ so that, given an arbitrary *query range* $r \subseteq \mathbb{R}^d$ of some fixed type, the number of points in $r \cap S$ can be computed efficiently. There is extensive literature on this class of problems [1], but little has been done to generalize it to a more continuous setting.

We consider range queries on (solid) polyhedra in $\mathbb{R}^d$, where the ranges are halfspaces. We denote the halfspaces above and below a hyperplane h by h^+ and h^-, respectively. Let P be a fixed polyhedron. A *halfspace volume query* asks, given a query hyperplane h, to compute the volume of the intersection $P \cap h^-$ (or equivalenty, of $P \cap h^+$).

Czyzowicz, Contreras-Alcalá, and Urrutia [3, 4] studied the problem of halfplane-area queries, in the special case where P is a convex polygon. In that case, an $O(n)$-space data structure can be constructed to find the two edges intersected by the query line h in $O(\log n)$ time. Given those two edges, they show a simple technique to compute the area of $P \cap h^-$ in $O(1)$ time. Boland and Urrutia [2] observe that the same method also works for non-convex polygons as long as h intersects exactly two edges of P. If h intersects k edges of P, these edges can be found in $O(k \log n)$ time using standard ray-shooting techniques. Then, given those k edges, the algorithm of Czyzowicz *et al.* can be generalized to compute the area of $P \cap h^-$ in $O(k)$ time.

In light of results in discrete range searching, where most queries can be performed in sublinear time aftre suitable preprocessing, it is natural to ask whether halfplane-area queries can be performed in $o(k)$ time. Recently, Langerman [6] gave a negative answer, showing that any straight-line program requires $\Omega(k)$ operations to answer arbitrary halfplane area queries, even if the k edges intersecting h are known in advance, and regardless of preprocessing time and storage space.

Iacono and Langerman [5] generalized the data structures for $\mathbb{R}^2$ to simply connected polyhedra P in $\mathbb{R}^3$. As in the planar case, the k edges of P that intersect h can be found in $O(k \log n)$ time; given those k edges, the volume of $P \cap h^-$ can be computed in $O(k)$ time with a data structure using $O(n)$ space and preprocessing. Langerman's lower bound [6] implies that the $O(k)$ time bound is worst-case optimal when P is not convex, but this lower bound does not apply when P is convex.

Our main result is that Iacono and Langerman's algorithm is optimal even when P is convex.

Main Theorem. *For any $d \geq 3$, any straight-line program that answers halfspace-volume queries for a fixed convex polyhedron in $\mathbb{R}^d$ requires $\Omega(k)$ time in the worst case, where k is the number of edges intersecting the query hyperplane, regardless of preprocessing and storage space, even if the k intersected edges are known at preprocessing time.*

Like all lower bounds in the straight-line-program model, including Langerman's earlier result [6], our bound also holds in more general models of computation such as algebraic computation trees and the real RAM.

2 Proof

We prove our lower bound for a specific class of queries to be performed on a particular convex polyhedron P in $\mathbb{R}^3$. We first define a planar polygon Q with vertices $v_0, v_1, \ldots, v_n$, where $v_i = (a_i, a_i^2, 1)$ and $0 = a_0 < a_1 < \cdots < a_n$. This polygon is clearly convex. Our polyhedron P is the unbounded cone whose apex is the origin $(0, 0, 0)$ and whose intersection with the plane $z = 1$ is the polygon Q.

For any query hyperplane h, the polygon $P \cap h$ is a projective transformation of the base polygon Q, and computing the volume of $P \cap h^-$ clearly reduces to computing the area of this transformed polygon. To prove the lower bound, we consider the following more general problem. Let π denote the plane $z = 1$. A *projective area query* asks, given an arbitrary linear transformation $T : \mathbb{R}^3 \to \mathbb{R}^3$, represented by a 3×3 matrix, to compute the area of $T(P) \cap \pi$. (We can equivalently

*MIT Laboratory for Computer Science, edemaine@mit.edu

†University of Illinois at Urbana-Champaign, jeffe@cs.uiuc.edu, http://www.cs.uiuc.edu/~jeffe. Partially supported by NSF CAREER award CCR-0093348 and NSF ITR grants DMR-0121695 and CCR-0219594.

‡Chargé de recherches du FNRS, Université Libre de Bruxelles, stefan.langerman@ulb.ac.be

view T as a planar projective transformation from π to itself that maps Q to $T(P) \cap \pi$.) We easily observe that

$$\begin{aligned} \mathrm{vol}(T(P) \cap \pi^-) &= \det(T) \cdot \mathrm{vol}(P \cap T^{-1}(\pi^-)) \\ &= \frac{\det(T)}{3} \cdot \mathrm{area}(P \cap T^{-1}(\pi)). \end{aligned}$$

Both $\det(T)$ and the plane $T^{-1}(\pi)$ can be computed in constant time. Thus, to prove our main theorem, it suffices to show that answering an arbitrary projective area query for P requires $\Omega(n)$ time.

We prove this lower bound by considering transformations of the form

$$T_x = \begin{bmatrix} 1 & 0 & x \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

for some real value $x > 0$. The transformed polygon $Q'_x = T_x(P) \cap \pi$ has vertices $v'_0, v'_1, \ldots, v'_n$, where

$$v'_i = \left(\frac{a_i}{a_i x + 1}, \frac{a_i^2}{a_i x + 1}, 1 \right).$$

The area of Q'_x can be expressed as the sum of the signed areas of all triangles of the form $\triangle v'_0 v'_{i-1} v'_i$; recall that $v'_0 = v_0 = (0, 0, 1)$.

$$\begin{aligned} F(x) &= \mathrm{area}(Q'_x) \\ &= \sum_{i=2}^{n} \mathrm{area}(\triangle v'_0 v'_{i-1} v'_i) \\ &= \sum_{i=2}^{n} \frac{\mathrm{area}(\triangle v_0 v_{i-1} v_i)}{(a_i x + 1)(a_{i-1} x + 1)} \\ &= \frac{1}{2} \sum_{i=2}^{n} \frac{a_i^2 a_{i-1} - a_{i-1}^2 a_i}{(a_i x + 1)(a_{i-1} x + 1)} \\ &= \frac{1}{2} \sum_{i=2}^{n} \frac{(a_i^2 a_{i-1})(a_{i-1} x + 1) - (a_{i-1}^2 a_i)(a_i x + 1)}{(a_i x + 1)(a_{i-1} x + 1)} \\ &= \frac{1}{2} \sum_{i=2}^{n} \left(\frac{a_i^2 a_{i-1}}{a_i x + 1} - \frac{a_{i-1}^2 a_i}{a_{i-1} x + 1} \right) \\ &= \frac{1}{2} \left(\sum_{i=2}^{n} \frac{a_i^2 a_{i-1}}{a_i x + 1} - \sum_{i=1}^{n-1} \frac{a_i^2 a_{i+1}}{a_i x + 1} \right) \\ &= \frac{1}{2} \left(\frac{a_1^2 a_2}{a_1 x + 1} + \sum_{i=2}^{n-1} \frac{a_i^2 (a_{i-1} - a_{i+1})}{a_i x + 1} + \frac{a_n^2 a_{n-1}}{a_n x + 1} \right) \end{aligned}$$

$F(x)$ is a rational function in x, parameterized by the values $a_1, \ldots, a_n$. To prove a lower bound on the complexity of computing this function, we use the following theorem of Motzkin [7]:

Motzkin's Theorem. *Let K be an infinite field. If $u, v \in K[x]$ are relatively prime and the leading coefficient of v is 1, then*

$$L_+(u/v) \geq T(u, v) - 1, \quad L_*(u/v) \geq \frac{1}{2}(T(u, v) - 1)$$

where $L_+(f)$ is the minimum number of additions and subtractions, and $L_*(f)$ the minimum number of multiplications and divisions, required to evaluate f, where operations not involving x are regarded as costless. $T(u, v)$ is the degree of transcendence of the set of coefficients of u and v over the primefield of K.

To compute $F(x)$ over some primefield $\mathbb{K}$ (for example, $\mathbb{R}$ or $\mathbb{Q}$), we enlarge $\mathbb{K}$ to the extension field $K = \mathbb{K}(a_1, \ldots, a_n)$. If we write $F(x) \in K(x)$ as a quotient of two polynomials, the denominator $\prod_{i=1}^{n}(a_i x + 1)$ has n algebraically independent roots $-1/a_i$, and thus the set of its coefficients has degree of transcendence n over $\mathbb{K}$. Our lower bound now follows immediately from Motzkin's theorem.

Acknowledgments. This work was initiated during the Workshop on Geometric and Computational Aspects of Instance-Based Learning, held at the Bellairs Research Institute, Barbados, January 31–February 7, 2003, organized by Godfried Toussaint.

References

[1] P. K. Agarwal and J. Erickson. Geometric range searching and its relatives. *Advances in Discrete and Computational Geometry*, 1–56, 1999. Contemporary Mathematics 223, American Mathematical Society.

[2] R. Boland and J. Urrutia. Polygon area problems. *Proc. 12th Canad. Conf. Comput. Geom.*, 159–162, 2000. ⟨http://www.cccg.ca/proceedings/2000/⟩.

[3] F. Contreras-Alcalá. Cutting polygons and a problem on illumination of stages. M.Sc. thesis, Dept. Comp. Sci. University of Ottawa, Ottawa, ON, Canada, 1998. ⟨http://www.csi.uottawa.ca/~fhca/thesis/⟩.

[4] J. Czyzowicz, F. Contreras-Alcalá, and J. Urrutia. On measuring areas of polygons. *Proc. 10th Canad. Conf. Comput. Geom.*, 1998. ⟨http://www.cccg.ca/proceedings/1998/⟩.

[5] J. Iacono and S. Langerman. Volume queries in polyhedra. *Proc. Japan Conf. Discrete Comput. Geom.*, 156–159, 2000. Lecture Notes Comput. Sci. 2098, Springer-Verlag.

[6] S. Langerman. On the complexity of halfspace area queries. *Proc. 17th Annu ACM Sympos. Comput. Geom.*, 207–211, 2001. ACM Press.

[7] V. Strassen. Algebraic complexity theory. *Algorithms and Complexity*, chapter 11, 633–672, 1990. Handbook of Theoretical Computer Science A, MIT Press.