Theoretical Computer Science 421 (2012) 56–61

Contents lists available at SciVerse ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Complexity of approximating the vertex centroid of a polyhedron☆

Khaled Elbassioni [a,*], Hans Raj Tiwary [b]

[a] Max-Planck-Institut für Informatik, Saarbrücken, Germany

[b] Université Libre de Bruxelles (ULB), Département de Mathématique, CP 216, Brussels, Belgium

ARTICLE INFO

Article history:
Received 22 July 2010
Received in revised form 21 March 2011
Accepted 18 November 2011
Communicated by S. Sen

ABSTRACT

Let $\mathcal{P}$ be an $\mathcal{H}$-polytope in $\mathbb{R}^d$ with vertex set V. The vertex centroid is defined as the average of the vertices in V. We first prove that computing the vertex centroid of an $\mathcal{H}$-polytope, or even just checking whether it lies in a given halfspace, is #P-hard. We also consider the problem of approximating the vertex centroid by finding a point within an ϵ distance from it and prove this problem to be #P-easy in the sense that it can be solved efficiently using an oracle for some #P-complete problem. In particular, we show that given an oracle for counting the number of vertices of an $\mathcal{H}$-polytope, one can approximate the vertex centroid in polynomial time. Counting the number of vertices of a polytope defined as the intersection of halfspaces is known to be #P-complete. We also show that any algorithm approximating the vertex centroid to *any* "sufficiently" non-trivial (for example constant) distance, can be used to construct a fully polynomial-time approximation scheme for approximating the centroid and also an output-sensitive polynomial algorithm for the Vertex Enumeration problem. Finally, we show that for unbounded polyhedra the vertex centroid cannot be approximated to a distance of $d^{\frac{1}{2}-\delta}$ for any fixed constant $\delta > 0$ unless $P = NP$.

1. Introduction

An intersection of a finite number of closed halfspaces in $\mathbb{R}^d$ defines a polyhedron. A polyhedron can also be represented as $conv(V) + cone(Y)$, the Minkowski sum of the convex hull of a finite set of points V and the cone of a finite set of rays. A bounded polyhedron is called a polytope. In what follows, we will discuss mostly polytopes for simplicity and refer to the unbounded case explicitly only toward the end. We call a polytope (polyhedron resp.) defined by a set of inequalities an $\mathcal{H}$-polytope ($\mathcal{H}$-polyhedron resp.) and a polytope (polyhedron resp.) defined by vertices (and extreme rays) a $\mathcal{V}$-polytope (polyhedron resp.).

Let $\mathcal{P}$ be an $\mathcal{H}$-polytope in the ambient space $\mathbb{R}^d$ with vertex set V. Various notions try to capture the essence of a "center" of a polytope. Perhaps the most popular notion is that of the center of gravity of $\mathcal{P}$. Recently, Rademacher has proved that computing the center of gravity of a polytope is #P-hard [8]. The proof essentially relies on the fact that the center of gravity captures the volume of a polytope perfectly and that computing the volume of a polytope is #P-hard [4]. Note that, randomized polynomial algorithms exist that approximate the volume of a polytope within any arbitrary

☆ During part of this work the second author was supported by Graduiertenkolleg fellowship for Ph.D. studies provided by Deutsche Forschungsgemeinschaft.

* Corresponding author. Tel.: +49 6819325107.
E-mail addresses: elbassio@mpi-inf.mpg.de (K. Elbassioni), hans.raj.tiwary@ulb.ac.be (H.R. Tiwary).

0304-3975/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2011.11.017

factor [5]. It is also easy to see that the center of gravity can be approximated by simply sampling random points from the polytope, the number of samples depending polynomially on the desired approximation (see Algorithm 5.8 of [5]).

In this paper, we study a variant of the notion of "center" defined as the centroid (average) of the vertices of P. Despite being quite a natural feature of polytopes, this variant seems to have received very little attention both from theoretical and computational perspectives. Throughout this paper, we will refer to the vertex centroid just as centroid. The reader should note that in popular literature the word centroid refers more commonly to the center of gravity. We nevertheless use the same terminology for simplicity of language. Our motivation for studying the centroid stems from the fact that the centroid encodes the number of vertices of a polytope. As we will see, this also makes computing the centroid hard.

The parallels between centroid and the center of gravity of a polytope mimic the parallels between the number of vertices and the volume of a polytope. Computing the volume is #P-complete [4] but it can be approximated quite well [5]. Accordingly, the problem of computing the corresponding centroid is hard [8, Theorem 1] but the volume centroid can be approximated quite well [5]. On the other hand, computing the number of vertices is not only #P-complete [3,7], it cannot be approximated within any factor polynomial in the number of facets and the dimension. As we will see in this paper, computing the vertex centroid of an $\mathcal{H}$-polytope exactly is #P-hard. Even approximating the vertex centroid for unbounded $\mathcal{H}$-polyhedra turns out to be NP-hard. We do not know the complexity of approximating the vertex centroid of an $\mathcal{H}$-polytope (bounded case).

The problem of enumerating vertices of an $\mathcal{H}$-polytope has been studied for a long time. However, in spite of years of research it is neither known to be hard nor is there an output sensitive polynomial algorithm for it. Note that the problem of enumerating all vertices of an $\mathcal{H}$-polytope is different from the problem of counting the number of vertices. While the latter problem is known to be #P-complete [3,7], the complexity status of Vertex Enumeration is open [1]. A problem that is polynomially equivalent to the Vertex Enumeration problem is to decide if a given list of vertices of an $\mathcal{H}$-polytope is complete [1]. In this paper, we show that any algorithm that approximates the centroid of an arbitrary polytope to any "sufficiently" non-trivial distance can be used to obtain an output sensitive polynomial algorithm for the Vertex Enumeration problem.

The main results of this paper are the following.

(I) Computing the centroid of an $\mathcal{H}$-polytope is #P-hard, and it remains #P-hard even just to decide whether the centroid lies in a halfspace.

(II) Approximating the centroid of an $\mathcal{H}$-polytope is #P-easy.

(III) Any algorithm approximating the centroid of an arbitrary polytope within a distance $d^{\frac{1}{2}-\delta}$ for any fixed constant $\delta > 0$ can be used to obtain a fully polynomial-time approximation scheme for the centroid approximation problem and also an output sensitive polynomial algorithm for the Vertex Enumeration problem.

(IV) There is no polynomial algorithm that approximates the vertex centroid of an arbitrary $\mathcal{H}$-polyhedron within a distance $d^{\frac{1}{2}-\delta}$ for any fixed constant $\delta > 0$, unless $P = NP$.

The first two results in (I) follow easily from the hardness of counting the number of vertices of an $\mathcal{H}$-polytope. The next result is obtained by repeatedly slicing the given polytope, in a way somewhat similar to the one used to prove that computing the center of gravity is #P-hard [8]. The bootstrapping result in (III) is obtained by taking the product of the polytope with itself sufficiently many times. Using this result, and building on a construction in [6], we prove (IV). Namely, we use a modified version of the construction in [6] to show that it is NP-hard to approximate the centroid within a distance of $1/d$, then we use the result in (III) to bootstrap the hardness threshold to $d^{\frac{1}{2}-\delta}$ for any fixed constant $\delta > 0$.

We should remark that for the approximation of the centroid, we only consider polytopes (and polyhedra) whose vertices lie inside a unit hypercube. To see how this assumption can easily be satisfied, notice that a halfspace h can be added to a polyhedron P such that $P \cap h$ is bounded and the vertices of P are preserved in $P \cap h$. Suppose for simplicity that the polyhedron is defined as the set of inequalities $Ax \leq \mathbf{1}$. Then it is easy to see that a halfspace whose normal is any vector lying in cone(A) when added to the set of inequalities makes the polyhedron bounded if the defining hyperplane has a large enough distance from the origin. The distance required for this only requires a number of bits polynomial in the size of the input A. For details, see [9] where the same issue is discussed in a different context.

Once we have a polytope in $\mathbb{R}^d$, solving $2d$ linear programs gives us the width along each coordinate axis. The polytope can be scaled by a factor depending on the width along each axis to obtain a polytope all whose vertices lie inside a unit hypercube. In case we started with a polyhedron P, the scaled counterpart of the halfspace h that was added can be thrown to get back a polyhedron that is a scaled version of P and all whose vertices lie inside the unit hypercube. In Section 2.2, we provide justification for this assumption.

Since all the vertices of the polytope (or polyhedron) lie inside a unit hypercube, any arbitrary point from inside this hypercube is at a distance of at most $d^{\frac{1}{2}}$ from the vertex centroid. Thus, our last result above should be contrasted to the fact that approximating the vertex centroid within a distance of $d^{\frac{1}{2}}$ is trivial. Also, even though we discuss only polytopes *i.e.* bounded polyhedra in Sections 2.1 and 2.2, the results and the proofs are valid for the unbounded case as well. We discuss the unbounded case explicitly only in Section 2.3.