Optimization Algorithms for Faster Computational Geometry*

Zeyuan Allen-Zhu[1], Zhenyu Liao[2], and Yang Yuan[3]

1 **Princeton University, Princeton, USA**
`zeyuan@csail.mit.edu`

2 **Boston University, Boston, USA**
`zhenyul@bu.edu`

3 **Cornell University, Ithaka, USA**
`yangyuan@cs.cornell.edu`

Abstract

We study two fundamental problems in computational geometry: finding the maximum inscribed ball (MaxIB) inside a bounded polyhedron defined by m hyperplanes, and the minimum enclosing ball (MinEB) of a set of n points, both in d-dimensional space. We improve the running time of iterative algorithms on

- MaxIB from $\widetilde{O}(md\alpha^3/\varepsilon^3)$ to $\widetilde{O}(md + m\sqrt{d}\alpha/\varepsilon)$, a speed-up up to $\widetilde{O}(\sqrt{d}\alpha^2/\varepsilon^2)$, and[1]
- MinEB from $\widetilde{O}(nd/\sqrt{\varepsilon})$ to $\widetilde{O}(nd + n\sqrt{d}/\sqrt{\varepsilon})$, a speed-up up to $\widetilde{O}(\sqrt{d})$.

Our improvements are based on a novel saddle-point optimization framework. We propose a new algorithm `L1L2SPSolver` for solving a class of regularized saddle-point problems, and apply a randomized Hadamard space rotation which is a technique borrowed from compressive sensing. Interestingly, the motivation of using Hadamard rotation solely comes from our optimization view but not the original geometry problem: indeed, it is not immediately clear why MaxIB or MinEB, as a geometric problem, should be easier to solve if we rotate the space by a unitary matrix. We hope that our optimization perspective sheds lights on solving other geometric problems as well.

1998 ACM Subject Classification F.2 Analysis of Algorithms and Problem Complexity – geometrical problems and computations

Keywords and phrases maximum inscribed balls, minimum enclosing balls, approximation algorithms

Digital Object Identifier 10.4230/LIPIcs.ICALP.2016.53

1 Introduction

The goal of this paper is to bridge the fields of optimization and computational geometry using a simple unified saddle-point framework. As two immediate products of this new connection, we obtain faster iterative algorithms to approximately solve two fundamental problems in computational geometry: the maximum inscribed ball problem (MaxIB) and the

* This is an abstract of our full paper at `http://arxiv.org/abs/1412.1001` [3]. The first version of this paper appeared in December 2014 but contains only the smooth convex optimization based algorithms. The second version of this paper appeared in December 2015 and already contains all the technical details of this present paper.

[1] $\alpha \geq 1$ is the aspect ratio of the polyhedron. Throughout this paper we use the $\widetilde{O}$ notation to hide logarithm factors such as $\log m, \log d, \log \alpha$, and $\log(1/\varepsilon)$.

43rd International Colloquium on Automata, Languages, and Programming (ICALP 2016).
Editors: Ioannis Chatzigiannakis, Michael Mitzenmacher, Yuval Rabani, and Davide Sangiorgi;
Article No. 53; pp. 53:1–53:6
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

minimum enclosing ball problem (MinEB). Our methods are composed of simple updating rules on vectors and therefore do not require geometric operations that are found in classical algorithms. This is another example of surprisingly good results obtained using optimization insights following the current trend of theoretical computer science.

In the rest of this introduction, we describe the definitions of the MaxIB and MinEB problems and review prior work. In the next three sections, we describe our saddle-point formulation and algorithms for MaxIB and MinEB.

Maximum Inscribed Ball (MaxIB). In the MaxIB problem, we are given a polyhedron P in $\mathbb{R}^d$ defined by m halfspaces $\{H_1, \dots, H_m\}$. Each halfspace H_j is characterized by a linear constraint $\langle A_j, x \rangle + b_j \geq 0$. As in prior work [20], we assume that P is bounded (so $m \geq d$) and a common point is known to be contained in P – without loss of generality, let it be the origin O. Let $\alpha \geq 1$ be an upper bound on the aspect ratio of P, i.e., the ratio between the radii of the minimum enclosing ball and the maximum inscribed ball of P, and $\varepsilon > 0$ be a desired error bound.

The goal of MaxIB is to find a point $x \in P$ such that its minimum distance to all the bounding hyperplanes H_j is at least $(1 - \varepsilon) r_{\mathsf{opt}}$, where r_{opt} is the radius of a maximum inscribed ball of P.

Besides the applications in computational geometry, MaxIB has also been used in the column generation method [13] and the sphere method [14] for linear programming, and the central cutting-plane method for convex programming [9].

When the dimension is a constant, the ε-kernel technique (see the survey [1]) yields a linear-time approximation algorithm for MaxIB based on core-set construction. However, its running time is proportional to $\varepsilon^{-\Omega(d)}$. In high dimensions, finding the maximum inscribed ball remains a challenging problem in theoretical computer science and operations research. One can reduce this problem to a linear program [9] and rely on existing LP solvers, however, the so-obtained algorithm can be too slow for practical purposes (although still in polynomial time).

In an influential paper, Xie, Snoeyink, and Xu [20] obtained an approximation algorithm for MaxIB with running time $O(md\alpha^3/\varepsilon^3 + md\alpha \log \alpha) = \widetilde{O}(md\alpha^3/\varepsilon^3)$. Their algorithm is based on a number of interesting geometric observations, as well as a dual transformation to reduce the MaxIB problem to a sequence of minimum enclosing ball (MinEB) instances, which they solve by applying known core-set techniques [6, 12]. Unfortunately, their cubic dependence on α and $1/\varepsilon$ undermines the practical applicability of their algorithm.

In Section 3, we use saddle-point optimization techniques to obtain an algorithm `MaxIBSPSolver` with running time $\widetilde{O}\big(md + m\sqrt{d}\alpha/\varepsilon\big)$. In other words, we reduce the dependence on both α and $1/\varepsilon$ from cubic to linear, and improve the running time by a factor up to $\sqrt{d}\alpha^2/\varepsilon^2$. We emphasize that our improvement could be significant in the views of theoretical computer scientists, operations researchers, as well as experimentalists:

- In theoretical computer science, one usually views α and ε as large constants so our improvement can be seen as $\widetilde{\Omega}(\sqrt{d})$ if one ignores the input reading time $O(md)$.
- In operations research or statistics, one usually concentrate on the convergence rate which is the ε dependence (recall that the seminal work of Nesterov is only to reduce $1/\varepsilon$ to $1/\sqrt{\varepsilon}$ [15]). Our improvement in this paper is from $1/\varepsilon^3$ to $1/\varepsilon$.
- In practice, if α is 10 for the polyhedron, ε is 10%, and the dimension $d = 100$, our method could potentially be 10^5 times faster than that of [20]. We leave it a future work to inspect the practical performance of our method on real-life datasets.

In the full version of this paper [3], we also apply convex (rather than saddle-point) optimization and obtain a *parallel* algorithm `MaxIBConvexSolver` with slightly slower *total* running time $\widetilde{O}(md\alpha/\varepsilon)$. However, in terms of *parallel* running time (i.e., the number of parallelizable iterations, a classical benchmark used by iterative solvers [4]), `MaxIBConvexSolver` improves the result of [20] by a factor $\widetilde{\Omega}(\alpha^2/\varepsilon^2)$.

Minimum Enclosing Ball (MinEB). In the MinEB problem, we are given a set $\{a_1, a_2, \ldots, a_n\} \subseteq \mathbb{R}^d$ of points in the d-dimensional space and are asked to find a point $x \in \mathbb{R}^d$ so that its maximum distance to all the n points is at least $(1+\varepsilon)R_{\mathsf{opt}}$, where R_{opt} is the radius of a minimum enclosing ball that contains all the points in this set.

As originally studied by Sylvester in [18], the problem of MinEB has found numerous applications in fields such as data mining, learning, statistics, and computer graphics. In particular, the relationship between MinEB and support vector machines (SVMs) has been recently emphasized by [11, 10, 7, 17]. Efficient algorithms for this problem are both of theoretical and practical importance.

If the dimension d is constant, the algorithm of Welzl [19] solves MinEB exactly in linear time. Unfortunately, its dependency on d is exponential.

For large dimensions, a sequence of works based on the core-set technique [6, 12, 5, 21, 7] has given algorithms whose best known running time is $O(nd/\varepsilon)$. This running time is tight for the core-set technique, as, in the worst-case, the size of a coreset of MinEB is at least $\Omega(1/\varepsilon)$ [5]. Another type of algorithm due to Clarkson, Hazan, and Woodruff [8] achieves a running time of $\widetilde{O}(n/\varepsilon^2 + d/\varepsilon)$. This algorithm is fast for large values of ε, but may not be suitable for very small ε. All these cited algorithms converge at best in $O(1/\varepsilon)$ iterations.

Recently, Saha, Vishwanathan, and Zhang [17] designed two algorithms for MinEB that successfully overcame this $1/\varepsilon$ barrier. Using our ε-notation for multiplicative error, they give one algorithm which works in the ℓ_2-norm and achieves a running time of $O(ndQ/\sqrt{\varepsilon})$, and another algorithm which works in the ℓ_1-norm and achieves a running time of $O(nd\sqrt{\log n}L/\sqrt{\varepsilon})$. While the values of Q and L depend on the input structure, we observe that Q can be as large as $O(\sqrt{n})$, while L is never larger than a constant. In other words, their proposed algorithms have worst-case running times $O(n^{1.5}d/\sqrt{\varepsilon})$ and $O(nd\sqrt{\log n}/\sqrt{\varepsilon})$. The key component behind the result of Saha, Vishwanathan, and Zhang is the excessive gap framework of Nesterov [16], which is a primal-dual first-order approach for structured non-smooth optimization problems.

In Section 4, we rewrite MinEB as a saddle-point optimization problem, and obtain an algorithm `MinEBSPSolver` that runs in $\widetilde{O}(nd + n\sqrt{d}/\sqrt{\varepsilon})$. This is faster than the previous algorithm [17] by a factor up to $\sqrt{d}$, and faster than the popular core-set algorithm by a factor up to $\sqrt{d}/\sqrt{\varepsilon}$.

As an additional result, in the full version of this paper [3], we also observe that MinEB can be directly formulated as a convex (rather than saddle-point) optimization problem, and get an algorithm `MinEBConvexSolver` matching the running time of [17] but with much simpler analysis.

Remark. For both MaxIB and MinEB, one can also use interior-point types of algorithms to obtain a convergence rate of $\log(1/\varepsilon)$. However, this fast convergence rate comes at the cost of having expensive iterations: each iteration typically requires solving a linear equation system in the input size, making it impractical for very-large-scale inputs. Therefore, in this paper, we choose to focus on iterative methods whose iterations run in nearly-linear time.

1.1 Our Techniques

Our `MaxIBSPSolver` and `MinEBSPSolver` rely on (min-max) saddle-point optimization to solve MaxIB and MinEB respectively. More specifically, we reduce MaxIB and MinEB to solving the regularized saddle-point program:

$$\max_{x\in\mathbb{R}^d}\min_{y\in\Delta_m}\frac{1}{d}y^T Ax+\frac{1}{d}y^T b+\lambda H(y)-\frac{\gamma}{2}\|x\|_2^2 \ ,$$

where $H(\cdot)$ is the entropy function over m-dimensional probabilities vectors, and $\lambda, \gamma > 0$ are fixed regularization parameters. We call this ℓ_1-ℓ_2 saddle-point optimization because, borrowing language from optimization, this objective is strongly convex with respect to the ℓ_1 norm on the y side and strongly concave with respect to the ℓ_2 norm on the x side.

To solve this saddle-point problem efficiently, we iteratively update x and y. In particular, in each iteration we update x by a *random coordinate*, and update y fully using multiplicative weight updates. Therefore, this method can be viewed as an *accelerated, coordinate-based*, first-order method for saddle-point optimization. To the best of our knowledge, the only previously known accelerated, coordinate-based method on saddle-point optimization was SPDC [22], one of the state-of-the-art algorithms used for empirical risk minimizations in machine learning. We call our algorithm `L1L2SPSolver`.

A Surprising Hadamard Rotation. Unfortunately, solely applying `L1L2SPSolver` does not solve MinEB or MaxIB fast enough. In particular, the running time of `L1L2SPSolver` relies on the largest absolute values of A's entries. If the entries of A are very non-uniform – say, with a few very large entries and mostly small ones – the performance could be somewhat unsatisfactory. (In particular, we no longer have a $\sqrt{d}$ factor speed-up.)

To overcome this difficulty, we apply a randomized Hadamard transformation on A to uniformize its entries, so that all entries of A are relatively small. This transformation is inspired by the fast Johnson-Lindenstrauss transform [2] proposed for numerical linear algebra and compressive sensing purposes, and is another crucial ingredient behind our running time improvements.

Surprisingly, this Hadamard rotation comes solely from our optimization view but not the geometry. Indeed, it is not immediately clear why MaxIB or MinEB, as geometric problems, should be easier to solve if we rotate the space by a unitary (Hadamard) matrix.

Our Contributions. We summarize the main contributions of this paper as follows:

- We provide significantly faster algorithms on MaxIB and MinEB.
- This is the first time coordinate-based saddle-point optimization algorithm is applied to MaxIB, MinEB, or perhaps to any computational geometry problem.
- Since the ℓ_1-ℓ_2 saddle-point problem seems very natural, our `L1L2SPSolver` method can potentially lead to other applications in the future.
- The speed-up we obtained from the Hadamard rotation is an algebraic technique but applied to geometric problems. It sheds lights on solving perhaps more geometric problems faster using optimization insights.

2 Main Body of This Paper

We defer all the mathematical details of this paper to `http://arxiv.org/abs/1412.1001` [3].

Acknowledgements. We thank Lorenzo Orecchia for insightful comments and discussions. He took an active part in stages of this research, and yet declined to be a co-author. We thank Sepideh Mahabadi and Jinhui Xu for helpful conversations. We also thank the anonymous reviewers' valuable comments on the earlier versions of this paper.

References

1 Pankaj K. Agarwal, Sariel Har-Peled, and Kasturi R. Varadarajan. Geometric approximation via coresets. *Combinatorial and computational geometry*, 52:1–30, 2005.

2 Nir Ailon and Bernard Chazelle. Faster dimension reduction. *Communications of the ACM*, 53(May):97–104, 2010. doi:10.1145/1646353.1646379.

3 Zeyuan Allen-Zhu, Zhenyu Liao, and Yang Yuan. Optimization Algorithms for Faster Computational Geometry. *ArXiv e-prints*, abs/1412.1001, December 2014.

4 Zeyuan Allen-Zhu and Lorenzo Orecchia. Using optimization to break the epsilon barrier: A faster and simpler width-independent algorithm for solving positive linear programs in parallel. In *Proceedings of the 26th ACM-SIAM Symposium on Discrete Algorithms*, SODA'15, 2015.

5 Mihai Bădoiu and Kenneth L Clarkson. Optimal core-sets for balls. *Computational Geometry*, 40(1):14–22, 2008.

6 Mihai Bădoiu, Sariel Har-Peled, and Piotr Indyk. Approximate clustering via core-sets. In *Proceedings of the thiry-fourth annual ACM symposium on Theory of computing – STOC'02*, page 250, New York, New York, USA, 2002. ACM Press. doi:10.1145/509907.509947.

7 Kenneth L Clarkson. Coresets, sparse greedy approximation, and the frank-wolfe algorithm. *ACM Transactions on Algorithms (TALG)*, 6(4):63, 2010.

8 Kenneth L. Clarkson, Elad Hazan, and David P. Woodruff. Sublinear optimization for machine learning. *Journal of the ACM*, 59(5):23:1–23:49, October 2012. doi:10.1145/2371656.2371658.

9 Jack Elzinga and Thomas G. Moore. A central cutting plane algorithm for the convex programming problem. *Mathematical Programming*, 8(1):134–145, 1975.

10 Bernd Gärtner and Martin Jaggi. Coresets for polytope distance. In *Proceedings of the 25th annual symposium on computational geometry*, pages 33–42. ACM, 2009.

11 Sariel Har-Peled, Dan Roth, and Dav Zimak. Maximum margin coresets for active and noise tolerant learning. In *Proceedings of the 20th international joint conference on Artifical intelligence*, pages 836–841. Morgan Kaufmann Publishers Inc., 2007.

12 Piyush Kumar, Joseph S. B. Mitchell, and E. Alper Yildirim. Approximate minimum enclosing balls in high dimensions using core-sets. *Journal of Experimental Algorithmics*, 8:1–29, January 2003. doi:10.1145/996546.996548.

13 Chungmok Lee and Sungsoo Park. Chebyshev center based column generation. *Discrete Applied Mathematics*, 159(18):2251–2265, 2011.

14 Katta G. Murty. $o(m)$ bound on number of iterations in sphere methods for lp. *Algorithmic Operations Research*, 7(1):30–40, 2012.

15 Yurii Nesterov. A method of solving a convex programming problem with convergence rate $O(1/k^2)$. In *Doklady AN SSSR (translated as Soviet Mathematics Doklady)*, volume 269, pages 543–547, 1983.

16 Yurii Nesterov. Excessive Gap Technique in Nonsmooth Convex Minimization. *SIAM Journal on Optimization*, 16(1):235–249, January 2005. doi:10.1137/S1052623403422285.

17 Ankan Saha, S. V. N. Vishwanathan, and Xinhua Zhang. New Approximation Algorithms for Minimum Enclosing Convex Shapes. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms – SODA'11*, pages 1146–1160, September 2011. arXiv:0909.1062.

18 James Joseph Sylvester. A question in the geometry of situation. *Quarterly Journal of Pure and Applied Mathematics*, 1, 1857.

19 Emo Welzl. *Smallest enclosing disks (balls and ellipsoids)*. Springer, 1991.

20 Yulai Xie, Jack Snoeyink, and Jinhui Xu. Efficient algorithm for approximating maximum inscribed sphere in high dimensional polytope. In *Proceedings of the 22nd annual symposium on computational geometry – SCG'06*, 2006. `doi:10.1145/1137856.1137861`.

21 E. Alper Yildirim. Two algorithms for the minimum enclosing ball problem. *SIAM Journal on Optimization*, 19(3):1368–1391, 2008.

22 Yuchen Zhang and Lin Xiao. Stochastic Primal-Dual Coordinate Method for Regularized Empirical Risk Minimization. In *Proceedings of the 32nd International Conference on Machine Learning*, ICML 2015, 2015. `arXiv:1409.3257`.