Fast Approximation and Randomized Algorithms for Diameter

Sharareh Alipour* Bahman Kalantari† Hamid Homapour*

Abstract

We consider approximation of diameter of a set S of n points in dimension m. Eğecioğlu and Kalantari [8] have shown that given any $p \in S$, by computing its farthest in S, say q, and in turn the farthest point of q, say q', we have $\mathrm{diam}(S) \leq \sqrt{3}\ d(q, q')$. Furthermore, iteratively replacing p with an appropriately selected point on the line segment pq, in at most $t \leq n$ additional iterations, the constant bound factor is improved to $c_* = \sqrt{5 - 2\sqrt{3}} \approx 1.24$. Here we prove when $m = 2$, $t = 1$. This suggests in practice a few iterations may produce good solutions in any dimension. Here we also propose a randomized version and present large scale computational results with these algorithm for arbitrary m. The algorithms outperform many existing algorithms. On sets of data as large as $1,000,000$ points, the proposed algorithms compute solutions to within an absolute error of 10^{-4}.

1 Introduction

Given a finite set of points S in $\mathbb{R}^m$, the diameter of S, denoted by $\mathrm{diam}(S)$, is the maximum distance between two points in S. Let $d(\cdot, \cdot)$ denote the Euclidean distance. For a given point $p \in \mathbb{R}^m$, let $f(p)$ denote the farthest point of p in S. Let $r_p = d(p, f(p))$. We write $f^2(p)$ for $f(f(p))$. For $m = 2$ the problem can be solved in $O(n \log n)$ time. Computing the diameter of a point set is a fundamental problem and has a long history. Clarkson and Shor gave a randomized $O(n \log n)$ algorithm [5]. Recent attempts to solve the 3-dimensional diameter problem led to $O(n \log^3 n)$ [7, 2] and $O(n \log^2 n)$ deterministic algorithms [7, 3]. Finally Ramos found an optimal $O(n \log n)$ deterministic algorithm [6]. All these algorithms use complex data structures and algorithmic techniques such as 3-dimensional convex hulls, intersection of balls, furthest-point Voronoi diagrams, point location search structures or parametric search. There are many other papers that focus on this problem, see [9, 4, 11, 10]. The first nontrivial approximation algorithm for this problem for arbitrary m was given in [8], approximating the diameter to within a factor of $\sqrt{3}$. The operation cost of this algorithm is $O(mn)$. Additionally, [8] describes an iterative algorithm that in $t \leq n$ iterations, each of cost $O(mn)$, produces an approximation of $\mathrm{diam}(S)$ to within a factor of $c_* \approx 1.24$.

In Section 2, we describe the algorithm in [8] for arbitrary m and state a new result for $m = 2$ in approximation of the diameter to within a factor of c_*. In Section 3, we describe iterative algorithms for arbitrary dimension m. In Section 4, we present experimental results of the proposed algorithms in various dimensions and make comparison with several existing algorithms. We conclude in Section 5.

2 Fast Approximation

Consider the following algorithm. Pick arbitrary $p \in S$. Compute $f(p)$. Clearly, $r_p \leq \mathrm{diam}(S) \leq 2r_p$. Next compute $f^2(p)$. In [8] it is shown $r_{f(p)} \leq \mathrm{diam}(S) \leq \sqrt{3} r_{f(p)}$. To improve this bound, in [8] the following iterative procedure is described: Let $p' = \alpha p + (1 - \alpha) f(p)$, $\alpha = r_{f(p)}/r_p$. Let q be the midpoint of p' and $f(p)$. Compute $f(q)$ and $f^2(q)$. If $d(f(q), f^2(q)) \leq d(f(p), f^2(p))$, then $\mathrm{diam}(S) \leq c_* d(f(p), f^2(p))$. Otherwise, replaces S with $S \setminus \{p, f(p)\}$, and repeat the process, replacing p with q, $f(p)$ with $f(q)$. Eventually, in $t \leq n$ iterations, each of cost $O(mn)$, we obtain an approximation of $\mathrm{diam}(S)$ to within a factor of c_*. However, in [8] no constant bound on t is given. We prove.

Theorem 1. ([1]) *When* $m = 2$, $\mathrm{diam}(S) \leq c_* \max\{d(f(p), f^2(p)), d(f(q), f^2(q))\}$. □

*Department of Computer engineering, Sharif University of Technology, Tehran, Iran. {shalipour, homapour}@ce.sharif.edu

†Department of Computer Science, Rutgers, State University of New Jersey. kalantari@cs.rutgers.edu

3 Iterative Algorithms

We propose two iterative algorithms for approximating the diameter in any dimension. The first is essentially [8] and the second a randomized version. They each have an input t as the number of iterations. However, we used small t since they produce high accuracy solution. In the randomized algorithm we begin with arbitrary $p \in S$ and compute $f(p)$. Next let q to be the midpoint of p and $f(p)$ and compute $f(q)$ and $f^2(q)$. We iterate this algorithm. In the next step, we can either begin from $f(p)$ or $f^2(q)$. To do so, we randomly choose one with equal probability. This becomes our new point. Then we compute the farthest point from the chosen point and compare the estimate of diameter of previous step with the new one. In practice we used $t = 2, 3$ and 5.

4 Experimental Results

The most comparable approaches to ours are the algorithms proposed in [11] and [10]. We have used the package implemented by Malandain and Boissonnat's in [11]. They have implemented their algorithms and we have also implemented our algorithms and added them to their package. In their experiments they generated 2 types of data set: Volume based distributions, in a cube, in a ball, and in sets of constant width (only in 2D); and Surface based distributions, on a sphere, and on ellipsoids. They also used real inputs. We have also used the same package to generate data sets and the same real inputs. Malandain and Boissonnat's have implemented the following algorithms (i) Malandain and Boissonnat's exact algorithm; (ii) Malandain and Boissonnat's approximation algorithm; (iii) Har-Peled's algorithm: implemented by Malandain and Boissonnat; (iv) Hybrid1 algorithm: proposed by Malandain and Boissonnat which is combination of their algorithm and Har-Peled's algorithm; (v) Hybrid2 algorithm: another modification of the two algorithms presented by Malandain and Boissonnat's algorithm and Har-Peled's algorithm. We have generated the data sets and computed the diameter for each set using each of the above algorithms and our proposed algorithms. The experimental results are shown in detail in [1]. The first 5 algorithms are implemented by Malandain and Boissonnat's and the next one is the implementation of the first algorithm with $t = 2$, and the randomized algorithm with $t = 2, 3$ and 5. In all the data sets, the difference between the approximated value and exact value of diameter is less than 10^{-4} where $\text{diam}(S) > 1$ even with $t = 2$ iterations for both algorithms. The running time of the randomized algorithm, with $t = 2$ iteration is better than all others. The proposed algorithms are more efficient in higher dimensions and require no extra memory. Also, by virtue of their efficiency they can be implemented for big data sets. Another advantage of these algorithms is that in higher dimensions, their running time is significantly better than the other algorithms.

5 Conclusion

We studied the diameter problem, a significant problem in computational geometry. We presented a fast 1.24-approximation algorithm in dimension 2. We believe the same bound applies to any dimension. Its verification is the subject of future work. We proposed two iterative algorithms, one is randomized. We also implemented these algorithms and compared the running times with related works. Based on experimental results the algorithms are very efficient.

References

[1] Allipour, Sharareh, B. Kalantari, and H. Homapour. "Fast approximation and randomized algorithms for diameter." http://arxiv.org/pdf/1410.2195v1.pdf, 2014.

[2] Amato, Nancy M., Michael T. Goodrich, and Edgar A. Ramos. "Parallel algorithms for higher-dimensional convex hulls." Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on. IEEE, 1994.

[3] Bespamyatnikh, Sergei. "An efficient algorithm for the three-dimensional diameter problem." Discrete and Computational Geometry 25.2 (2001): 235-255.

[4] Chan, Timothy M. "Faster core-set constructions and data stream algorithms in fixed dimensions." Proceedings of the twentieth annual symposium on Computational geometry. ACM, 2004.

[5] Clarkson, Kenneth L., and Peter W. Shor. "Applications of random sampling in computational geometry, II." Discrete and Computational Geometry 4.1 (1989): 387-421.

[6] Ramos, Edgar A. "Deterministic algorithms for 3-D diameter and some 2-D lower envelopes." Proceedings of the sixteenth annual symposium on Computational geometry. ACM, 2000.

[7] Ramos, Edgar A. "Intersection of unit-balls and diameter of a point set in R¡ sup¿ 3¡/sup¿." Computational Geometry 8.2 (1997): 57-65.

[8] Egecioglu, Ömer, and Bahman Kalantari. "Approximating the diameter of a set of points in the Euclidean space." Information Processing Letters 32.4 (1989): 205-211.

[9] Fournier, Herv, and Antoine Vigneron. "A tight lower bound for computing the diameter of a 3D convex polytope." Algorithmica 49.3 (2007): 245-257.

[10] Har-Peled, Sariel. "A practical approach for computing the diameter of a point set." Proceedings of the seventeenth annual symposium on Computational geometry. ACM, 2001.

[11] Malandain, Grgoire, and Jean-Daniel Boissonnat. "Computing the diameter of a point set." International Journal of Computational Geometry and Applications 12.06 (2002): 489-509.