

Approximation and Inapproximability Results for Maximum Clique of Disc Graphs in High Dimensions

Peyman Afshani^a and Hamed Hatami^b

Department of Computer Science

^aUniversity of Waterloo

^bUniversity of Toronto

Abstract

We prove algorithmic and hardness results for the problem of finding the largest set of a fixed diameter in the Euclidean space. In particular we prove that if A is the largest subset of diameter r of n points in the Euclidean space, then for every $\epsilon > 0$ there exists a polynomial time algorithm that outputs a set B of size at least $|A|$ and of diameter at most $r(\sqrt{2} + \epsilon)$. On the hardness side roughly speaking we show that unless $P = NP$ for every $\epsilon > 0$ it is not possible to guarantee the diameter $r(\sqrt{4/3} - \epsilon)$ for B .

1 Introduction

The problem that we consider in this paper can be formulated as a clustering problem. These type of problems have been studied for quite long time and have many theoretical and practical applications in computer science [7]. A branch of clustering problems includes problems in which given a set of points the goal is to find a “cluster” (or clusters) with minimum size or maximum number of points. Typical examples of clusters include spheres, boxes, or any other shape of fixed complexity. Of course the difficulty of the problem greatly depends on the definition of cluster. The clusters that we consider here are all the shapes of constant diameter. Specifically, we consider the following problem:

Problem 1: Let P be a set of n points in \mathbb{R}^d and $r > 0$ be a real number. Find a subset $S \subseteq P$ of maximum size which satisfies $\text{diam}(S) \leq r$. Here, $\text{diam}(S)$ is the diameter of set S .

This problem is in fact the maximum clique problem in disc graphs: build a graph G with the vertex set $V(G) = P$ and connect two points x, y if and only if $|x - y| \leq r$. There is a one to one correspondence between cliques in G and sets of diameter at most r in P . Thus, here the diameter is fixed and our aim is to maximize the number of points. On the other hand, we can fix the number of points and ask for the minimum diameter:

Problem 2: Let P be a set of n points in \mathbb{R}^d and $k > 0$ be an integer. Find a subset $S \subseteq P$ of size k with minimum diameter.

We show that both these problems are NP-hard when the dimension is sufficiently large, $d = \Omega(\log n)$. In fact, we prove a stronger result which shows that even certain approximations of this problem is impossible unless $P=NP$. This approximation is defined in the following way:

Definition. A (t, s) -approximation algorithm for Problem 1 is an algorithm that returns a set S of size at least $t \times \text{Opt}$ so that $\text{diam}(S) \leq sr$, where Opt is the size of the optimal answer to Problem 1.

Definition. A (t, s) -approximation algorithm for Problem 2 is an algorithm that returns a set S of size at least tk so that $\text{diam}(S) \leq s \times \text{Opt}$, where Opt is the optimal answer to Problem 2.

These two problems are obtained by relaxing the size and diameter constraints of the output set. A simple observation shows that these two problems are equivalent.

Lemma 1 *There exists a polynomial time (t, s) -approximation algorithm for Problem 1 if and only if there exists a (t, s) -approximation algorithm for Problem 2.*

Proof. Let \mathcal{A} be a (t, s) -approximation algorithm for Problem 1. Consider an instance (P, k) of Problem 2. For every pair of points $x, y \in P$, run \mathcal{A} with parameter $r_{x,y} := |x - y|$. We output the minimum $r_{x,y}$ for which the answer of \mathcal{A} is of size at least kt . Let S_o be the optimal solution to the (P, k) instance of Problem 2. At some point \mathcal{A} is called with parameter $r := \text{diam}(S_o)$. Now the output of \mathcal{A} is a set of size at least $|S_o| \times t \geq kt$ and with diameter at most $rs = s \times \text{diam}(S_o)$.

To prove the other direction let \mathcal{B} be a (t, s) -approximation algorithm for Problem 2. Consider an instance (P, r) for Problem 1. A (t, s) -approximation algorithm for Problem 1 can be obtained in a similar way by running \mathcal{B} for every $k = 1, \dots, n$. ■

Since these two problems are equivalent we refer to both of them as the Diameter Approximation Problem.

Both Problems 1 and 2 are solvable in the 2-dimensional plane in polynomial time [2, 7, 8]. For Problem 2 the fastest known algorithm achieves the running time $O(n \log n + k^{2.5} n \log k)$ [8]. It is shown in [1] that in the 3-dimensional space there is a $(\frac{\pi}{\cos^{-1} 1/3}, 1)$ -approximation algorithm. Finally, when the dimension d is a fixed constant, one can design a polynomial time approximation scheme achieving a $(1, 1 + \epsilon)$ -approximation, for every $\epsilon > 0$ [1]. It is also easy to see that there exists a trivial $(1, 2)$ -approximation algorithm for this problem in any dimension: a ball with radius r about a point $x \in P$ containing the maximum number of points is a $(1, 2)$ -approximation for Problem 1. Thus, it is interesting to study at which point the problem turns from polynomially solvable to NP-hard. We have the following result in this direction:

Theorem 2 *For every $\epsilon > 0$ there exists $d = \Omega(\log n)$ and $c_0 > 0$, so that there is no polynomial time $(1 + c_0, \sqrt{4/3} - \epsilon)$ -approximation algorithm for Diameter Problem unless $P=NP$.*

We also improve upon the trivial $(1, 2)$ -approximation algorithm and obtain the following theorem:

Theorem 3 *For every $\epsilon > 0$ there is a polynomial time $(1, \sqrt{2} + \epsilon)$ -approximation algorithm for Diameter Problem in any dimension.*

The proof of Theorem 2 uses spectral properties of graphs to move from combinatorics of graphs to geometry of Euclidean space. Then we conclude from the fact that finding maximum independent set of a 3-regular graph is NP-hard, that there is no $(1+c_0, \sqrt{4/3}-\epsilon)$ -approximation algorithm for the Diameter Problem unless $P=NP$. Interestingly, having Theorem 3 in hand, it is possible to move in the other direction. We can apply Theorem 3 to the geometric representation of the graph and solve maximum independent set for certain graphs.

2 From Graphs to Euclidean Space

In this section we prove Theorem 2. We use spectral techniques to show that a certain metric on the graph embeds isometrically into the Euclidean space. This type of reduction from geometry to problems in graph theory via metric embedding has been successfully applied in [10, 13]. However, our problem is different and does not fit in their framework. Our proof of Theorem 2 relies on the following result:

Theorem 4 [3] *The problem of finding the size of the maximum independent set in a 3-regular graph is APX-hard.*

2.1 Proof of Theorem 2

We show that unless $P=NP$, there is no $(1+c_0, \sqrt{4/3}-\epsilon)$ -approximation algorithm for Diameter Problem for a constant $1 + c_0$. To prove this we reduce it to the maximum independent set problem in 3-regular graphs.

Consider a 3-regular graph G . Denote by G^c the complement of G , and notice that cliques in G^c correspond to independent sets in G . We begin by finding a lower bound for the minimum eigenvalue of the adjacency matrix A_{G^c} of G^c . Denote by $\lambda_1 \leq \dots \leq \lambda_n$ the eigenvalues of A_{G^c} . Since G^c is a $(n-4)$ -regular graph, the eigenvalues of its Laplacian are $n-4-\lambda_n \leq \dots \leq n-4-\lambda_1$. It is well-known [9] that the maximum eigenvalue of the Laplacian is bounded by n . This shows that the minimum eigenvalue of A_G is at least -4 . So,

$$Q := A_{G^c} + 4I$$

is a symmetric positive semi-definite matrix. It is possible to find the Cholesky decomposition of $Q = U^t U$ in polynomial time. Construct a mapping $f : V(G^c) \rightarrow \mathbb{R}^n$ by mapping the vertex i to the i -th row of U . Note that

$$|f(i)|^2 = f(i) \cdot f(i) = Q_{i,i} = 4,$$

and

$$|f(i) - f(j)|^2 = |f(i)|^2 + |f(j)|^2 - 2f(i) \cdot f(j) = 8 - 2Q_{i,j}.$$

Thus

$$|f(i) - f(j)| = \begin{cases} \sqrt{6} & ij \in E(G^c) \\ \sqrt{8} & ij \notin E(G^c) \end{cases}$$

In other words, each vertex of $V(G^c)$ is mapped to a vector of size 2 and the distance between two vertices is $\sqrt{6}$, if there is an edge between them, and $\sqrt{8}$ if not.

We apply the Johnson-Lindenstrauss dimension reduction lemma [11] to the point set $f(G^c)$ causing distortion at most δ for a sufficiently small constant $\delta = \delta(\epsilon)$ to drop the dimension to $c \log n$ for $c = \delta^{-2}$. Let $g(G^c)$ be the corresponding set of points in $\mathbb{R}^{c \log n}$. Now applying a $(1 + c_0, \sqrt{4/3} - \epsilon)$ -approximation algorithm to Problem 1 with $g(G^c)$ and $r = \sqrt{6}$, finds a clique of size at most $(1 + c_0)$ times the size of the maximum clique in G^c . Theorem 4 shows that there is a constant $c_0 > 0$ such that this is impossible unless $P = NP$.

2.2 Proof of Theorem 3

In this section we prove Theorem 3 by applying simple geometric techniques. We follow the general ideas and techniques of [2, 1], borrowing and generalizing the main tool from [2]. The idea is to extend and generalize the trivial $(1, 2)$ -approximation. One way to interpret the $(1, 2)$ -approximation is to say that any set of diameter r can be placed inside a sphere of diameter $2r$. To obtain a $(1, \sqrt{2} + \epsilon)$ -approximation we first show that any set of diameter r can actually be placed inside a ball of diameter $(\sqrt{2} + \epsilon)r$ then we produce a polynomial time algorithm to compute such a sphere.

Let A be the optimal answer to Problem 1. We start by proving that A is inside a ball of diameter $(\sqrt{2} + \epsilon)r$. Let $B(P, t)$ denote a ball of radius t centered at point P . At the beginning, P_1 is an arbitrary point of A and thus we have $A \subset B(P_1, r)$. At the i -th step we assume $A \subset B(V_i, r_i)$ for a $V_i \in \mathbb{R}^d$ and some value $r_i \leq r$ to be determined later.

Let P_{i+1} be the point with maximum distance to V_i . This implies,

$$A \subset B(V_i, r_i) \cap B(P_{i+1}, r) \cap B(V_i, |V_i - P_{i+1}|)$$

and since $|V_i - P_{i+1}| \leq r_i$, we have:

$$A \subset B(P_{i+1}, r) \cap B(V_i, |V_i - P_{i+1}|).$$

If $x = |V_i - P_{i+1}| \leq r\sqrt{2}/2$ then the set of points is inside a ball of diameter $\sqrt{2}r$. So we assume $x > r\sqrt{2}/2$.

Consider a point L on the intersection of boundaries of the two balls $B(P_{i+1}, r)$ and $B(V_i, x)$ (Figure 1). Consider the plane passing through L , P_{i+1} and V_i and draw the line LV_{i+1} perpendicular to the segment $P_{i+1}V_i$. A simple calculation proves that:

$$|LV_{i+1}| = r\sqrt{1 - \frac{r^2}{4x^2}} \leq r\sqrt{1 - \frac{r^2}{4r_i^2}}$$

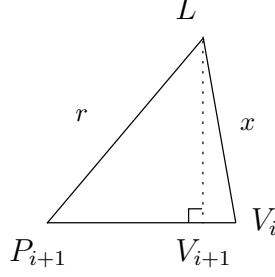


Figure 1: The way we determine V_{i+1} .

Define $r_{i+1} = r\sqrt{1 - \frac{r^2}{4r_i^2}}$. It can be also verified that if $x > r\sqrt{2}/2$ then $|P_{i+1}V_{i+1}| < |V_{i+1}L|$ and the ball $B(V_{i+1}, |LV_{i+1}|)$ will contain the intersection $B(P_{i+1}, r) \cap B(V_i, x)$. This implies $A \subset B(V_{i+1}, |LV_{i+1}|) \subset B(V_{i+1}, r_{i+1})$. It is easy to check that the sequence r_1, r_2, \dots converges to $r\sqrt{2}/2$. Thus given any ϵ it is possible to fix a constant k such that $r_k < r\sqrt{2}/2 + \epsilon$.

To obtain an algorithm from the discussion above, we only need to consider all different possible choices for P_1, \dots, P_k . Discarding the invalid choices or choices that result in invalid state, each choice for P_1, \dots, P_k leads to a ball with radius at most $r\sqrt{2}/2 + \epsilon$. Now the algorithm outputs the one which contains the maximum number of points. Since k is a constant, the algorithm is polynomial.

Remark 5 We believe that the result of Theorem 3 is almost sharp and the constant $\sqrt{4/3} - \epsilon$ in Theorem 2 can be improved to $\sqrt{2} - \epsilon$.

Remark 6 We must note that finding the smallest ball that covers any set of diameter one is indeed an old problem and more precise results have been obtained using Helly type arguments [6, 12]. However, our problem is different since first, we are interested in the algorithmic aspect of the problem and second, the set of diameter one which is to be covered is not known to the algorithm.

Corollary 7 Fix $\epsilon > 0$ and let G be a graph so that there exists a mapping $f : V(G) \rightarrow \mathbb{R}^n$ satisfying $|f(u) - f(v)| > \sqrt{2} + \epsilon$ if $u \sim v$ and $|f(u) - f(v)| \leq 1$ otherwise. Then it is possible to find the size of the maximum independent set of G in polynomial time.

Proof. The mapping f can be found using a simple semi-definite programming. Then we apply the algorithm of Theorem 3 to $f(V(G))$ to find an independent set in G . ■

Corollary 8 Fix $\epsilon > 0$ and let G be a graph whose minimum eigenvalue is at least $-2 + \epsilon$. it is possible to find the size of the maximum independent set of G in polynomial time.

There is already a body of work dedicated to characterization of all graphs with the smallest eigenvalue of at least -2 , [5, 4]. Finally, these graphs have been characterized as “generalized line graphs” plus some finite set of exceptions. This characterization gives an alternative polynomial algorithm for finding the maximum independent set in such graphs.

References

- [1] P. Afshani and T. Chan. Approximation algorithms for maximum cliques in 3d unit-disk graphs. In *Proceedings of the 17th Canadian Conference on Computational Geometry (CCCG'05)*, pages 19–22, 2005.
- [2] A. Aggarwal, H. Imai, N. Katoh, and S. Suri. Finding k points with minimum diameter and related problems. *Journal of Algorithms*, 12(1):38–56, 1991.
- [3] P. Berman and M. Karpinski. On some tighter inapproximability results (extended abstract). In *ICAL '99: Proceedings of the 26th International Colloquium on Automata, Languages and Programming*, pages 200–209, London, UK, 1999. Springer-Verlag.
- [4] F. C. Bussemaker and A. Neumaier. Exceptional graphs with smallest eigenvalue -2 and related. *Mathematics of Computation*, 59(200):583–608, October 1992.
- [5] P. J. Cameron, J. M. Goethals, J. J. Seidel, and E. E. Shult. Line graphs, root systems, and elliptic geometry. *Journal of Algebra*, 43:305–327, November 1976.
- [6] L. Danzer, B. Grünbaum, and V. Klee. Helly’s theorem and its relatives. In *Convexity*, volume 7 of *Proc. Symp. Pure Math.*, pages 101–180. American Mathematical Society, Providence, 1963.
- [7] A. Datta, H.-P. Lenhof, C. Schwarz, and M. Smid. Static and dynamic algorithms for k -point clustering problems. *Journal of Algorithms*, 19:474–503, 1995.
- [8] D. Eppstein and J. Erickson. Iterated nearest neighbors and finding minimal polytopes. *Discrete and Computational Geometry*, 11:321–350, 1994.
- [9] C. Godsil and G. Royle. *Algebraic graph theory*, volume 207 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 2001.
- [10] V. Guruswami and P. Indyk. Embeddings and non-approximability of geometric problems. In *SODA '03: Proceedings of the fourteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 537–538, Philadelphia, PA, USA, 2003. Society for Industrial and Applied Mathematics.
- [11] W. Johnson and J. Lindenstrauss. Extensions of lipschitz maps into a hilbert space. *Contemporary Mathematics*, 26:189–206, 1984.
- [12] J. Matousek. *Lectures on Discrete Geometry*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2002.
- [13] L. Trevisan. When hamming meets euclid: the approximability of geometric tsp and mst (extended abstract). In *STOC '97: Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 21–29, New York, NY, USA, 1997. ACM Press.