On Closest Pair in Euclidean Metric: Monochromatic is as Hard as Bichromatic

Karthik C. S.[1]
Weizmann Institute of Science, Rehovot, Israel
karthik.srikanta@weizmann.ac.il
https://orcid.org/0000-0001-9105-364X

Pasin Manurangsi[2]
University of California, Berkeley, USA
pasin@berkeley.edu

Abstract

Given a set of n points in $\mathbb{R}^d$, the (monochromatic) *Closest Pair* problem asks to find a pair of distinct points in the set that are closest in the ℓ_p-metric. Closest Pair is a fundamental problem in Computational Geometry and understanding its fine-grained complexity in the Euclidean metric when $d = \omega(\log n)$ was raised as an open question in recent works (Abboud-Rubinstein-Williams [FOCS'17], Williams [SODA'18], David-Karthik-Laekhanukit [SoCG'18]).

In this paper, we show that for every $p \in \mathbb{R}_{\geq 1} \cup \{0\}$, under the Strong Exponential Time Hypothesis (SETH), for every $\varepsilon > 0$, the following holds:

- No algorithm running in time $O(n^{2-\varepsilon})$ can solve the Closest Pair problem in $d = (\log n)^{\Omega_\varepsilon(1)}$ dimensions in the ℓ_p-metric.
- There exists $\delta = \delta(\varepsilon) > 0$ and $c = c(\varepsilon) \geq 1$ such that no algorithm running in time $O(n^{1.5-\varepsilon})$ can approximate Closest Pair problem to a factor of $(1+\delta)$ in $d \geq c \log n$ dimensions in the ℓ_p-metric.

In particular, our first result is shown by establishing the computational equivalence of the *bichromatic* Closest Pair problem and the (monochromatic) Closest Pair problem (up to n^ε factor in the running time) for $d = (\log n)^{\Omega_\varepsilon(1)}$ dimensions.

Additionally, under SETH, we rule out nearly-polynomial factor approximation algorithms running in subquadratic time for the (monochromatic) *Maximum Inner Product* problem where we are given a set of n points in $n^{o(1)}$-dimensional Euclidean space and are required to find a pair of distinct points in the set that maximize the inner product.

At the heart of all our proofs is the construction of a dense bipartite graph with low *contact dimension*, i.e., we construct a balanced bipartite graph on n vertices with $n^{2-\varepsilon}$ edges whose vertices can be realized as points in a $(\log n)^{\Omega_\varepsilon(1)}$-dimensional Euclidean space such that every pair of vertices which have an edge in the graph are at distance exactly 1 and every other pair of vertices are at distance greater than 1. This graph construction is inspired by the construction of locally dense codes introduced by Dumer-Miccancio-Sudan [IEEE Trans. Inf. Theory'03].

2012 ACM Subject Classification Theory of computation → Computational geometry, Theory of computation → Problems, reductions and completeness

Keywords and phrases Closest Pair, Bichromatic Closest Pair, Contact Dimension, Fine-Grained Complexity

Digital Object Identifier 10.4230/LIPIcs.ITCS.2019.17

[1] Supported by Irit Dinur's ERC-CoG grant 772839 and BSF grant 2014371.
[2] Supported by NSF under Grants No. CCF 1655215 and CCF 1815434.

Related Version A full version of the paper is available at `https://arxiv.org/abs/1812.00901`.

Acknowledgements We are grateful to Madhu Sudan for extremely helpful and informative discussion about AG codes; in particular, Madhu pointed us to [54]. We thank Bundit Laekhanukit and Or Meir for general discussions, and the Simons Institute for their wonderful work-space. Finally, we would like to thank Lijie Chen for sharing [17], and Orr Paradise for useful comments on an earlier draft of this manuscript.

1 Introduction

The Closest Pair of Points problem or *Closest Pair* problem (CP) is a fundamental problem in computational geometry: given n points in a d-dimensional metric space, find a pair of distinct points with the smallest distance between them. The Closest Pair problem for points in the Euclidean plane [51, 11] stands at the origins of the systematic study of the computational complexity of geometric problems [45, 40, 36, 20]. Since then, this problem has found abundant applications in geographic information systems [28], clustering [58, 7], and numerous matching problems (such as stable marriage [56]).

The trivial algorithm for CP examines every pair of points in the point-set and runs in time $O(n^2 d)$. Over the decades, there have been a series of developments on CP in low dimensional space for the Euclidean metric [10, 29, 35, 51, 11], leading to a deterministic $O(2^{O(d)} n \log n)$-time algorithm [11] and a randomized $O(2^{O(d)} n)$-time algorithm [46, 35]. For low (i.e., constant) dimensions, these algorithms are tight as a matching lower bound of $\Omega(n \log n)$ was shown by Ben-Or [9] and Yao [57] in the *algebraic decision tree* model, thus settling the complexity of CP in low dimensions. On other hand, for very high dimensions (i.e., $d = \Omega(n)$) there are subcubic algorithms [27, 31] in the ℓ_1, ℓ_2, and ℓ_∞-metrics using fast matrix multiplication algorithms [25]. However, CP in medium dimensions, i.e., $d = \text{polylog}(n)$, and in various ℓ_p-metrics, have been a focus of study in machine learning and analysis of Big Data [37], and it is surprising that, even with the tools and techniques that have been developed over many decades, when $d = \omega(\log n)$, there is no known subquadratic-time (i.e., $O(2^{o(d)} n^{2-\varepsilon})$-time) algorithm, for CP in any standard distance measure [30, 4, 31] . The absence of such algorithms was explicitly observed as early as the late nineties by Cohen and Lewis [19] but there was not any explanation until recently.

David, Karthik, and Laekhanukit [21] showed that for all $p > 2$, assuming the *Strong Exponential Time Hypothesis* (SETH), for every $\varepsilon > 0$, no algorithm running in $n^{2-\varepsilon}$ time can solve CP in the ℓ_p-metric, even when $d = \omega(\log n)$. Their conditional lower bound was based on the conditional lower bound (again assuming SETH) of Alman and Williams [6] for the *Bichromatic Closest Pair* problem[3] (BCP) where we are given two sets of n points in a d-dimensional metric space, and the goal is to find a pair of points, one from each set, with the smallest distance between them. Alman and Williams showed that for all $p \in \mathbb{R}_{\geq 1} \cup \{0\}$, assuming SETH, for every $\varepsilon > 0$, no algorithm running in $n^{2-\varepsilon}$ time can solve BCP in the $\omega(\log n)$-dimensional ℓ_p-metric space. Given that [6] show their lower bound on BCP for all ℓ_p-metrics, the lower bound on CP of [21] feels unsatisfactory, since the ℓ_2-metric is arguably the most interesting metric to study CP on. On the other hand, the answer to

[3] We remark that BCP is of independent interest as it's equivalent to finding the *Minimum Spanning Tree* in ℓ_p-metric [3, 38]. Moreover, understanding the fine-grained complexity of BCP has lead to better understanding of the query time needed for *Approximate Nearest Neighbor* search problem (see Razenshteyn's thesis [47] for a survey about the problem) with polynomial preprocessing time [50].

the complexity of CP in the Euclidean metric might be on the positive side, i.e., there might exist an algorithm that performs well in the ℓ_2-metric because there are more tools available, e.g., Johnson-Lindenstrauss' dimension reduction [33]. Thus we have the following question:

▶ **Open Question 1** (Abboud-Rubinstein-Williams[4] [2], Williams [55], David -Karthik-Laekhanukit [21]). *Is there an algorithm running in time $n^{2-\varepsilon}$ for some $\varepsilon > 0$ which can solve* CP *in the Euclidean metric when the points are in $\omega(\log n)$ dimensions?*

Even if the answer to the above question is negative, this does not rule out strong approximation algorithms for CP in the Euclidean metric, which might suffice for all applications. Indeed, we do know of subquadratic approximation algorithms for CP. For example, LSH based techniques can solve $(1+\delta)$-CP (i.e., $(1+\delta)$ factor approximate CP) in $n^{2-\Theta(\delta)}$ time [32], but cannot do much better [42, 43]. In a recent breakthrough, Valiant [52] obtained an approximation algorithm for $(1+\delta)$-CP with runtime of $n^{2-\Theta(\sqrt{\delta})}$. The state of the art is an $n^{2-\widetilde{\Theta}(\delta^{1/3})}$-time algorithm by Alman, Chan, and Williams [5]. Can the dependence on δ be improved indefinitely? For the case of $(1+\delta)$-BCP, assuming SETH, Rubinstein [50] answered the question in the negative. Does $(1+\delta)$-CP also admit the same negative answer?

▶ **Open Question 2.** *Is there an algorithm running in time $n^{2-\varepsilon}$ for some $\varepsilon > 0$ which can solve $(1+\delta)$-*CP *in the Euclidean metric when the points are in $\omega(\log n)$ dimensions for every $\delta > 0$?*

Another important geometric problem is the *Maximum Inner Product* problem (MIP): given n points in the d-dimensional Euclidean space, find a pair of distinct points with the largest inner product. This problem along with its bichromatic variant (*Bichromatic Maximum Inner Product* problem, denoted BMIP) is extensively studied in literature (see [2] and references therein). Abboud, Rubinstein, and Williams [2] showed that assuming SETH, for every $\varepsilon > 0$, no $2^{(\log n)^{1-o(1)}}$-approximation algorithm running in $n^{2-\varepsilon}$ time can solve BMIP when $d = n^{o(1)}$. It is a natural question to ask if their inapproximability result can be extended to MIP:

▶ **Open Question 3.** *Is there an algorithm running in time $n^{2-\varepsilon}$ for some $\varepsilon > 0$ which can solve γ-*MIP *in $n^{o(1)}$ dimensions for even $\gamma = 2^{(\log n)^{1-o(1)}}$?*

1.1 Our Results

In this paper we address all three previously mentioned open questions. First, we almost completely resolve Open Question 1. In particular, we show the following.

▶ **Theorem 4** (Subquadratic Hardness of CP). *Let $p \in \mathbb{R}_{\geq 1} \cup \{0\}$. Assuming* SETH*, for every $\varepsilon > 0$, no algorithm running in $n^{2-\varepsilon}$ time can solve* CP *in the ℓ_p-metric, even when $d = (\log n)^{\Omega_\varepsilon(1)}$.*

In particular we would like to emphasize that the dimension for which we show the lower bound on CP depends on ε. We would also like to remark that our lower bound holds even when the input point-set of CP is a subset of $\{0,1\}^d$. Finally, we note that the centerpiece of the proof of the above theorem (and also the proofs of the other results that will be subsequently mentioned) is the construction of a dense bipartite graph with low *contact dimension*, i.e., we construct a balanced bipartite graph on n vertices with $n^{2-\varepsilon}$ edges whose

[4] Please see the erratum in [1].

vertices can be realized as points in a $(\log n)^{\Omega_\varepsilon(1)}$-dimensional ℓ_p-metric space such that every pair of vertices which have an edge in the graph are at distance exactly 1 and every other pair of vertices are at distance greater than 1. This graph construction is inspired by the construction of locally dense codes introduced by Dumer, Miccancio, and Sudan [23] and uses special density properties of Reed Solomon codes. A detailed proof overview is given in Section 2.1.

Next, we improve our result in Theorem 4 in some aspects by showing $1 + o(1)$ factor inapproximability of CP even in $O_\varepsilon(\log n)$ dimensions, but can only rule out algorithms running in $n^{1.5-\varepsilon}$ time (as opposed to Theorem 4 which rules out exact algorithms for CP running in $n^{2-\varepsilon}$ time). More precisely, we show the following.

▶ **Theorem 5** (Subquadratic Hardness of gap-CP). *Let $p \in \mathbb{R}_{\geq 1} \cup \{0\}$. Assuming* SETH*, for every $\varepsilon > 0$, there exists $\delta(\varepsilon) > 0$ and $c(\varepsilon) > 1$ such that no algorithm running in $n^{1.5-\varepsilon}$ time that can solve $(1+\delta)$-CP in the ℓ_p-metric, even when $d = c \log n$.*

We remark that the $n^{1.5-\varepsilon}$ lower bound on approximate CP is an artifact of our proof strategy and that a different approach or an improvement in the state-of-the-art bound on the number of minimum weight codewords in algebraic geometric codes (which are used in our proof), will lead to the complete resolution of Open Question 2.

It should also be noted that the approximate version of CP and the dimension are closely related. Namely, using standard dimensionality reduction techniques [33][5] for $(1+\delta)$-CP, one can always assume that $d = O_\delta(\log n)$. In other words, hardness of $(1 + \delta)$-CP immediately yields logarithmic dimensionality bound as a byproduct.

Finally, we completely answer Open Question 3 by showing the following inapproximability result for MIP, matching the hardness for BMIP from [2].

▶ **Theorem 6** (Subquadratic Hardness of gap-MIP). *Assuming* SETH*, for every $\varepsilon > 0$, no algorithm running in $n^{2-\varepsilon}$ time can solve γ-MIP for any $\gamma \leq 2^{(\log n)^{1-o(1)}}$, even when $d = n^{o(1)}$.*

Recently, there have been a lot of results connecting BCP or $(1 + o(1))$-BCP to other problems (see [50, 15, 16, 17]). Now such connections can be extended to CP as well. For example, the following conditional lower bound follows from [50] for gap-CP in the edit distance metric.

▶ **Theorem 7** (Subquadratic Hardness of gap-CP in edit distance metric). *Assuming* SETH*, for every $\varepsilon > 0$, there exists $\delta(\varepsilon) > 0$ and $c(\varepsilon) > 1$ such that no algorithm running in $n^{1.5-\varepsilon}$ time can solve $(1+\delta)$-CP in the edit distance metric, even when $d = c \log n \log\log n$.*

2 Proof Overview

In this section, we provide an overview of our proofs and the formal proofs may be found in the full version of the paper. For ease of presentation, we will sometimes be informal here; all notions and proofs are formalized in subsequent sections. Our overview is organized as follows. First, in Subsection 2.1, we outline our proof of running time lower bounds for exact CP (Theorem 4). Then, in Subsection 2.2, we abstract part of our reduction using error-correcting codes, and relate them back to the works on locally dense codes [23, 18, 41] that inspire our constructions. Finally, in Subsection 2.3, we briefly discuss how to modify the base construction (i.e. code properties) to give conditional lower bounds for approximate CP and MIP (Theorems 5 and 6).

[5] In fact, since our results applies to $\{0, 1\}$-vectors, simply subsampling coordinates would also work.

2.1 Conditional Lower Bound on Exact Closest Pair

In this subsection, we provide a proof overview of a slightly weaker version of Theorem 4, i.e., we show that assuming SETH, for every $p \in \mathbb{R}_{\geq 1} \cup \{0\}$, no subquadratic time algorithm can solve CP in the ℓ_p-metric when $d = (\log n)^{\omega(1)}$. We prove such a result by reducing BCP in dimension d to CP in dimension $d + (\log n)^{\omega(1)}$, and the subquadratic hardness for CP follows from the subquadratic hardness of BCP established by [6]. Note that the results in this paper remain interesting even if SETH is false, as our reduction shows that BCP and CP are computationally equivalent[6] (up to $n^{o(1)}$ factor in the running time) when $d = (\log n)^{\omega(1)}$. The conditional lower bound on CP is merely a consequence of this computational equivalence. Finally, we note that a similar equivalence also holds between MIP and BMIP.

Understanding an obstacle of [21]. Our proof builds on the ideas of [21] who showed that assuming SETH, for every $p > 2$, no subquadratic time algorithm can solve CP in the ℓ_p-metric when $d = \omega(\log n)$. They did so by connecting the complexity of CP and BCP via the *contact dimension* of the balanced complete bipartite graph (biclique), denoted by $K_{n,n}$. We elaborate on this below.

To motivate the idea behind [21], let us first consider the trivial reduction from BCP to CP: given an instance A, B of BCP, we simply output $A \cup B$ as an instance of CP. This reduction fails because there is no guarantee on the distances of a pair of points both in A (or both in B). That is, there could be two points $\mathbf{a}, \mathbf{a}' \in A$ such that $\|\mathbf{a} - \mathbf{a}'\|_p$ is much smaller than the optimum of BCP on A, B. If we simply solve CP on $A \cup B$, we might find such $\mathbf{a}, \mathbf{a}'$ as the optimal pair but this does not give the answer to the original BCP problem. In order to circumvent this issue, one needs a gadget that "stretch" pairs of points both in A or both in B further apart while keeping the pairs of points across A and B close (and preserving the optimum of BCP on A, B). It turns out that this notion corresponds exactly to the contact dimension of the biclique, which we define below.

▶ **Definition 8** (Contact Dimension [44]). For any graph $G = (V, E)$, a mapping $\tau : V \to \mathbb{R}^d$ is said to *realize* G *(in the ℓ_p-metric)* if for some $\beta > 0$, the following holds for every distinct vertices u, v:

$$\|\tau(u) - \tau(v)\|_p = \beta \text{ if } \{u, v\} \in E, \text{ and,} \tag{1}$$

$$\|\tau(u) - \tau(v)\|_p > \beta \text{ otherwise.} \tag{2}$$

The *contact dimension* (in the ℓ_p-metric) of G, denoted by $\mathsf{cd}_p(G)$, is the minimum $d \in \mathbb{N}$ such that there exists $\tau : V \to \mathbb{R}^d$ realizing G in the ℓ_p-metric.

In this paper, we will be mainly interested in the contact dimension of bipartite graphs. Specifically, [21] only consider the contact dimension of the biclique $K_{n,n}$. Notice that a realization of biclique ensures that vertices on the same side are far from each other while vertices on different sides are close to each other preserving the optimum of BCP; these are exactly the desired properties of a gadget outlined above. Using this, [21] give a reduction from BCP to CP which shows that the two are computationally equivalent whenever $d = \Omega(\mathsf{cd}_p(K_{n,n}))$, as follows.

[6] We can reduce an instance of CP to an instance of BCP by randomly partitioning the input set of CP instance into two, and the optimal closest pair of points will be in different sets with probability $1/2$ (and this reduction can be made deterministic).

Let $A, B \subseteq \mathbb{R}^d$ each of cardinality n be an instance of BCP and let $\tau : A \dot\cup B \to \mathbb{R}^{\mathsf{cd}_p(K_{n,n})}$ be a map realizing the biclique $(A \dot\cup B, A \times B)$ in the ℓ_p-metric; we may assume w.l.o.g. that $\beta = 1$. Let δ be the distance between any point in A and any point in B (i.e., δ is an upper bound on the optimum of BCP). Let $\rho > 0$ be such that $\|\tau(\mathbf{a}) - \tau(\mathbf{b})\|_p > 1 + \rho$ for all $\mathbf{a} \in A, \mathbf{b} \in B$ (and this is guaranteed to exist by (2)). Moreover, let $k > \delta/\rho$ be any sufficiently large number. Consider the point-sets $\widetilde{A}, \widetilde{B} \subseteq \mathbb{R}^{d+\mathsf{cd}_p(K_{n,n})}$ of cardinality n each defined as

$$\widetilde{A} = \{\mathbf{a} \circ (k \cdot \tau(\mathbf{a})) \mid \mathbf{a} \in A\},\ \widetilde{B} = \{\mathbf{b} \circ (k \cdot \tau(\mathbf{b})) \mid \mathbf{b} \in B\},$$

where $\circ$ denotes the concatenation between two vectors and $k \cdot \mathbf{x}$ denotes the usual scalar-vector multiplication (i.e. scaling $\mathbf{x}$ up by a factor of k). For brevity, we write $\widetilde{\mathbf{a}}$ and $\widetilde{\mathbf{b}}$ to denote $\mathbf{a} \circ (k \cdot \tau(\mathbf{a}))$ and $\mathbf{b} \circ (k \cdot \tau(\mathbf{b}))$ respectively.

We now argue that, if we can find the closest pair of points in $\widetilde{A} \cup \widetilde{B}$, then we also immediately solve BCP for (A, B). More precisely, we claim that $(\mathbf{a}^*, \mathbf{b}^*) \in A \times B$ is a bichromatic closest pair of (A, B) if and only if $(\widetilde{\mathbf{a}^*}, \widetilde{\mathbf{b}^*})$ is a closest pair of $\widetilde{A} \cup \widetilde{B}$.

To see that this is the case, observe that, for cross pairs $(\widetilde{\mathbf{a}}, \widetilde{\mathbf{b}}) \in \widetilde{A} \times \widetilde{B}$, (1) implies that the distance $\|\widetilde{\mathbf{a}} - \widetilde{\mathbf{b}}\|_p$ is exactly $(k^p + \|\mathbf{a} - \mathbf{b}\|_p^p)^{1/p}$; hence, among these pairs, $(\widetilde{\mathbf{a}^*}, \widetilde{\mathbf{b}^*})$ is a closest pair iff $(\mathbf{a}^*, \mathbf{b}^*)$ is a bichromatic closest pair in A, B. Notice also that, since the bichromatic closest pair in A, B is of distance at most δ, the closest pair in $\widetilde{A} \cup \widetilde{B}$ is of distance at most $(k^p + \delta^p)^{1/p} \leq k + \delta$.

On the other hand, for pairs both from $\widetilde{A}$ or both from $\widetilde{B}$, the distance must be at least $k(1 + \rho)$, which is more than $k + \delta$ from our choice of k. As a result, these pairs cannot be a closest pair in $\widetilde{A} \cup \widetilde{B}$, and this concludes the sketch of the proof.

There are a couple of details that we have glossed over here: one is that the gap ρ cannot be too small (e.g., ρ cannot be as small as $1/2^n$) and the other is that we should be able to construct τ efficiently. Nevertheless, these are typically not an issue.

[21] show that $\mathsf{cd}_p(K_{n,n}) = \Theta(\log n)$ when $p > 2$ and that the realization can be constructed efficiently and with sufficiently large ρ. This implies the subquadratic hardness of CP (by reduction from BCP) in the ℓ_p-metric for all $p > 2$ and $d = \omega(\log n)$. However, it was known that $\mathsf{cd}_2(K_{n,n}) = \Theta(n)$ [24]. Thus, they could *not* extend their conditional lower bound to CP in the Euclidean metric[7] even when $d = o(n)$. In fact, this is a serious obstacle as it rules out many natural approaches to reduce BCP to CP in a black-box manner. Elaborating, the lower bound on $\mathsf{cd}_2(K_{n,n})$ rules out local gadget reductions which would replace each point with a composition of that point and a gadget with a small increase in the number of dimensions, as such gadgets can be used to construct a realization of $K_{n,n}$ in the Euclidean metric in a low dimensional space, contradicting the lower bound on $\mathsf{cd}_2(K_{n,n})$.

Overcoming the Obstacle: Beyond Biclique. We overcome the above obstacle by considering dense bipartite graphs, instead of the biclique. More precisely, we show that there exists a balanced bipartite graph $G^* = (A^* \dot\cup B^*, E^*)$ on $2n$ vertices such that $|E^*| \geq n^{2-o(1)}$ and $\mathsf{cd}_p(G^*)$ is small (i.e. $\mathsf{cd}_p(G^*) \leq (\log n)^{\omega(1)}$). We give a construction of such a graph below but before we do so, let us briefly argue why this suffices to show that BCP and CP are computationally equivalent (up to $n^{o(1)}$ multiplicative overhead in the running time) for dimension $d = \Omega(\mathsf{cd}_p(G^*))$.

[7] Note that plugging in the bound on $\mathsf{cd}_2(K_{n,n})$ in the result of [21] yields that assuming SETH, no subquadratic in n running time algorithm can solve CP when $d = \Omega(n)$. This is not a meaningful lower bound as just the input size of CP when $d = \Omega(n)$ is $\Omega(n^2)$.

Let us consider the same reduction which produces $\widetilde{A}, \widetilde{B}$ as before, but instead of using a realization of the biclique, we use a realization τ of G^*. This reduction is of course incorrect: if $(\mathbf{a}^*, \mathbf{b}^*)$ is not an edge in G^*, then $\|\tau(\mathbf{a}^*) - \tau(\mathbf{b}^*)\|_p$ could be large and, thus the corresponding pair of points $(\widetilde{\mathbf{a}^*}, \widetilde{\mathbf{b}^*}) \in \widetilde{A} \times \widetilde{B}$, may not be the closest pair. Nevertheless, we are not totally hopeless: if $(\mathbf{a}^*, \mathbf{b}^*)$ is an edge, then we are in good shape and the reduction is correct.

With the above observation in mind, consider picking a random permutation π of $A \cup B$ such that $\pi(A) = A$ and $\pi(B) = B$ and then initiate the above reduction with the map $(\tau \circ \pi)$ instead of τ. Note that $\tau \circ \pi$ is simply a realization of an appropriate permutation G' of G^* (i.e., G' is isomorphic to G^*). Due to this, the probability that we are "lucky" and $(\mathbf{a}^*, \mathbf{b}^*)$ is an edge in G' is $p := |E|/n^2$; when this is the case, solving CP on the resulting instance would give the correct answer for the original BCP instance. If we repeat this $\log n/p = n^{o(1)}$ times, we would find the optimum of the original BCP instance with high probability.

To recap, even when G^* is not a biclique, we can still use it to give a reduction from BCP to CP, except that the reduction produces multiple (i.e. $\widetilde{O}(n^2/|E^*|)$) instances of CP. We remark here that the reduction can be derandomized: we can deterministically (and efficiently) pick the permutations so that the permuted graphs covers $K_{n,n}$. As a minor digression, we would like to draw a parallel here with a recent work of Abboud, Rubinstein, and Williams [2]. The obstacle raised in [21] is about the impossibility of certain kinds of many-one gadget reductions. We overcame it by designing a reduction from BCP to CP which not only increased the number of dimensions but also the number of points (by creating multiple instances of CP). This technique is also utilized in [2] where they showed the impossibility of Deterministic Distributed PCPs (Theorem I.2 in [2]) but then overcame that obstacle by using an advice (which is then enumerated over resulting in multiple instances) to build Non-deterministic Distributed PCPs.

Constructing a dense bipartite graph with low contact dimension. We now proceed to construct the desired graph $G^* = (A^* \cup B^*, E^*)$. Note that any construction of a dense bipartite graph with contact dimension $n^{o(1)}$ is non-trivial. This is because it is known that a random graph has contact dimension $\Omega(n)$ in the Euclidean metric with high probability [49, 13], and therefore our graph construction must be significantly better than a random graph.

Our realization τ^* of G^* will map into a subset of $\{0,1\}^{(\log n)^{\omega(1)}}$. As a result, we can fix $p = 0$, since a realization of a graph with entries in $\{0,1\}$ in the Hamming-metric also realizes the same graph in every ℓ_p-metric for any $p \neq \infty$.

Fix $g = \omega(1)$. We associate $[n]$ with $\mathbb{F}_q^h$ where $q = \Theta\left((\log n)^g\right)$ is a prime and $h = \Theta\left(\frac{\log n}{g \cdot \log\log n}\right)$. Let $\mathcal{P}$ be the set of all univariate polynomials (in x) over $\mathbb{F}_q$ of degree at most $h-1$. We have that $|\mathcal{P}| = q^h = n$ and associate $\mathcal{P}$ with A^*. Let $\mathcal{Q}$ be the set of all univariate monic polynomials (in x) over $\mathbb{F}_q$ of degree h, i.e.,

$$\mathcal{Q} = \{x^h + p(x) \mid p(x) \in \mathcal{P}\}.$$

We associate the polynomials in $\mathcal{Q}$ with the vertices in B^* (note that $|\mathcal{Q}| = n$). In fact, we view the vertices in A^* and B^* as being uniquely labeled by polynomials in $\mathcal{P}$ and $\mathcal{Q}$ respectively. For notational clarity, we write p_a (resp. p_b) to denote the polynomial in $\mathcal{P}$ (resp. $\mathcal{Q}$) that is associated to $a \in A^*$ (resp. $b \in B^*$).

For every $a \in A^*$ and $b \in B^*$, we include (a, b) as an edge in E^* if and only if the polynomial $p_b - p_a$ (which is of degree h) has h distinct roots. This completes the construction of G^*. We have to now show the following two claims about G^*: (i) $|E^*| = n^{2-O(1/g)} = n^{2-o(1)}$ and (ii) there is $\tau : A^* \dot{\cup} B^* \to \{0,1\}^{(\log n)^{O(g)}} = \{0,1\}^{(\log n)^{\omega(1)}}$ that realizes G^*.

To show (i), let $\mathcal{R}$ be the set of all monic polynomials of degree h with h distinct roots. We have that $|\mathcal{R}| = \binom{q}{h}$. Fix a vertex $a \in A^*$. Its degree in G^* is exactly $|\mathcal{R}| = \binom{q}{h}$. This is because, for every polynomial $r \in \mathcal{R}$, $r + a$ belongs to $\mathcal{Q}$, and therefore $(a, r + a) \in E^*$. This implies the following bound on $|E^*|$:

$$|E^*| = q^h \cdot \binom{q}{h} \geq q^h \cdot \frac{q^h}{h^h} > \frac{n^2}{(\log n)^{\Theta((\log n)/(g \cdot \log\log n))}} = n^{2-O(1/g)}.$$

Next, to show (ii), we construct a realization $\tau^* : A^* \dot\cup B^* \to \mathbb{F}_q^q$ of G^*. We note that, it is simple to translate the entries to $\{0, 1\}$ instead of $\mathbb{F}_q$, by replacing $i \in \mathbb{F}_q$ with the i-th standard basis $\mathbf{e}_i \in \{0, 1\}^q$. This would result in a realization $\tau^* : A^* \dot\cup B^* \to \{0, 1\}^{q^2}$ of G^*; notice that the dimension of τ^* is $q^2 = \Theta((\log n)^{2g})$ as claimed.

We define τ^* as follows.

- For every $a \in A^*$, $\tau^*(a)$ is simply the vector of evaluation of p_a on every element in $\mathbb{F}_q$. More precisely, for every $j \in [q]$, the j-th coordinate of $\tau^*(a)$ is $p_a(j - 1)$.
- Similarly, for every $b \in B^*$ and $j \in [q]$, the j-th coordinate of $\tau^*(b)$ is $p_b(j - 1)$.

We now show that τ^* is indeed a realization of G^*; specifically, we show that τ^* satisfies (1) and (2) with $\beta = q - h$.

Consider any edge $(a, b) \in E^*$. Notice that $\|\tau^*(a) - \tau^*(b)\|_0$ is the number of $x \in \mathbb{F}_q$ such that $p_b(x) - p_a(x) \neq 0$. By definition of E^*, $p_b - p_a$ is a polynomial with h distinct roots over $\mathbb{F}_q$. Thus, $\|\tau^*(a) - \tau^*(b)\|_0 = q - h = \beta$ as desired.

Next, consider a non-edge $(a, b) \in (A^* \times B^*) \setminus E^*$. Then, we know that $p_b - p_a$ has at most $h - 1$ distinct roots over $\mathbb{F}_q$. Therefore, the polynomial $p_b - p_a$ is non-zero on at least $q - h + 1$ coordinates. This implies that $\|\tau^*(a) - \tau^*(b)\|_0 \geq q - h + 1 > \beta$.

Finally, for any distinct $a, a' \in A^*$, we have $\|\tau^*(a) - \tau^*(a')\|_0 \geq q - h + 1$ because $p_a - p_{a'}$ is a non-zero polynomial of degree at most $h - 1$ and thus can be zero over $\mathbb{F}_q$ in at most $h - 1$ locations. Similarly, $\|\tau^*(b) - \tau^*(b')\|_0 \geq q - h + 1$ for any distinct $b, b' \in B^*$.

This completes the proof sketch for both the claims about G^* and yields Theorem 4 for $d = (\log n)^{\omega(1)}$. Finally we remark that in the actual proof of Theorem 4, we will set the parameters in the above construction more carefully and achieve the bound $\mathsf{cd}_p(G^*) = (\log n)^{O_\varepsilon(1)}$.

2.2 Abstracting the Construction via Error-Correcting Codes

Before we move on to discuss the proofs of Theorems 6 and 5, let us give an abstraction of the construction in the previous subsection. This will allow us to easily generalize the construction for the aforemention theorems, and also to explain where our motivation behind the construction comes from in the first place.

Dense Bipartite Graph with Low Contact Dimension from Codes. In order to construct a balanced bipartite graph G^* on $2n$ vertices with $n^{2-o(1)}$ edges such that $\mathsf{cd}_p(G^*) \leq d^*$, it suffices to have a code C^* with the following properties:

- $C^* \subseteq \mathbb{F}_q^\ell$ of cardinality n is a linear code with block length ℓ over alphabet $\mathbb{F}_q$, and minimum distance Δ.
- There exists a *center* $s^* \in \mathbb{F}_q^\ell$ and $r^* < \Delta$ such that $|C^*|^{1-o(1)}$ codewords are at Hamming distance exactly r^* from s^* and no codeword is at distance less than r^* from s^*.
- $q \cdot \ell = d^*$.

We also require that C^* and s^* can be constructed in poly(n) time but we shall ignore this requirement for the ease of exposition.

We describe below how to construct G^* from C^*, but first note that the construction of G^* we saw in the previous subsubsection was just showing that Reed Solomon codes [48] of block length $q = \Theta((\log n)^g)$ and message length $h = \Theta\left(\frac{\log n}{g \cdot \log\log n}\right)$ over alphabet $\mathbb{F}_q$ with minimum distance $q - h + 1$ has the above properties. The center s^* in that construction was the evaluation of the polynomial x^h over $\mathbb{F}_q$, and r^* was $q - h$.

In general, to construct G^* from C^*, we first define a subset $S^* \subseteq \mathbb{F}_q^\ell$ of cardinality n as follows:

$$S^* = \{\mathbf{s}^* + \mathbf{c} \mid \mathbf{c} \in C^*\}.$$

We associate the vertices in A^* with the codewords of C^* and vertices in B^* with the strings in S^*. For any $(\mathbf{a}, \mathbf{b}) \in A^* \times B^*$, let $(\mathbf{a}, \mathbf{b}) \in E^*$ if and only if $\|\mathbf{b} - \mathbf{a}\|_0 = r^*$. This completes the construction of G^*. We have to now show the following claims about G^*: (i) $|E^*| = n^{2-o(1)}$ and (ii) there is $\tau : A^* \dot\cup B^* \to \{0,1\}^{q \cdot \ell}$ that realizes G^*.

Item (i) follows rather easily from the properties of C^* and s^*. Let T^* be the subset of C^* of all codewords which are at distance exactly equal to r^* from s^*. From the definition of s^*, we have $|T^*| = |C^*|^{1-o(1)}$. Fix $\mathbf{a} \in A^*$. Its degree in G^* is $|T^*| = |C^*|^{1-o(1)}$. This is because for every codeword $\mathbf{t} \in T^*$ we have that $\mathbf{t} - \mathbf{a}$ is a codeword in C^* (from the linearity of C^*) and thus $\mathbf{s}^* - \mathbf{t} + \mathbf{a}$ is in S^*, and therefore $(\mathbf{a}, \mathbf{s}^* - \mathbf{t} + \mathbf{a}) \in E^*$.

For item (ii), consider the identity mapping $\tau^* : A^* \dot\cup B^* \to \mathbb{F}_q^\ell$ that maps each string to itself. It is simple to check that τ^* realizes G^* in the Hamming metric (with $\beta = r^*$).

Recall from the previous subsection that given $\tau^* : A^* \dot\cup B^* \to \mathbb{F}_q^\ell$ that realizes G^* in the Hamming metric, it is easy to construct $\tau : A^* \dot\cup B^* \to \{0,1\}^{q \cdot \ell}$ that realizes G^* in the Hamming metric with a q multiplicative factor blow-up in the dimension. This completes the proof of both the claims about G^* and gives a general way to prove Theorem 4 given the construction of C^* and s^*.

Finding Center from Another Code. One thing that might not be clear so far is: where does the center s^* come from? Here we provide a systematic way to produce such an s^*, by looking at another code that contains C^*. More precisely, let $C^* \subseteq \widetilde{C}^* \subseteq \mathbb{F}_q^\ell$ be two linear codes with the same block length and alphabet. Suppose that the distance of C^* is Δ, the distance of $\widetilde{C}^*$ is r^* and that $r^* < \Delta$. It is easy to see that, by taking s^* to be any element of $\widetilde{C}^* \setminus C^*$, it holds that every codeword in C^* is at distance at least r^* from s^*, simply because both s^* and the codewords of C^* are codewords of $\widetilde{C}^*$.

Hence, we are only left to argue that there are many codewords of C^* that is of distance exactly r^* from s^*. While this is not true in general, we can show by an averaging argument that this is true (for some $s^* \in \widetilde{C}^*$) if a large fraction (e.g. $|C^*|^{-o(1)}$ fraction) of codewords of $\widetilde{C}^*$ has Hamming weight exactly r^*.

Indeed, viewing in this light, our previous choice of center for Reed-Solomon code (i.e. evaluation of x^h) is not coincidental: we simply take $\widetilde{C}^*$ to be another Reed-Solomon code with message length $h + 1$ (whereas the base code C^* is of message length h).

Comparison to Locally Dense Codes. We end this subsection by remarking that the codes that we seek are very similar to locally dense codes [23, 18, 41], which is indeed our inspiration. A *locally dense code* is a linear code of block length ℓ and large minimum distance

Δ, admitting a ball centered at s of radius[8] $r < \Delta$ and containing a large (i.e. $\exp(\text{poly}(\ell))$) number of codewords[9]. Such codes are non-trivial to construct and in particular all known constructions of locally dense codes are using codes that beat the Gilbert-Varshamov (GV) bound [26, 53]; in other words we need to do better than random codes to construct them. This is because (as noted in [23]), for a random code $C \subseteq \mathbb{F}_q^\ell$ (or any code that does not beat the GV bound), a random point in $\mathbb{F}_q^\ell$ acting as the center contains in expectation less than one codeword in a ball of radius Δ. Of course, this is simply an intuition and not a formal proof that a locally dense code needs to beat the GV bound, since there may be more sophisticated ways to pick a center.

Although the codes we require are similar to locally dense codes, there are differences between the two. Below we list four such differences: the first two makes it *harder* for us to construct our codes whereas the latter two makes it *easier* for us.

- We seek a center s^* so that no codewords in C^* lies at distance less than r^*, as opposed to locally dense codes which allows codewords to be close to s^*. This is indeed where our idea of using another code $\widetilde{C}^* \supseteq C^*$ comes in, as picking s^* from $\widetilde{C}^* \setminus C^*$ ensures us that no codeword of C^* is too close to s^*.
- Another difference is that we need the number of codewords at distance r^* from s^* to be very large, i.e., $|C^*|^{1-o(1)}$, whereas locally dense codes allow for much smaller number of codewords. Indeed, the deterministic constructions from [18, 41] only yield the bound of $2^{O(\sqrt{\log |C^*|})}$. Hence, these do not directly work for us.
- Locally dense codes requires r to be at most $(1-\varepsilon)\Delta$ for some constant $\varepsilon > 0$, whereas we are fine with any $r^* < \Delta$. In fact, our Reed-Solomon code based construction above only yields $r^* = \Delta - 1$ which would not suffice for locally dense codes. Nevertheless, as we will see later for inapproximability of CP, we will also need the ratio r^*/Δ to be a constant bounded away from 1 as well and, since we need a code with these extraordinary properties, they are very hard to find. Indeed, in this case we only manage to prove a weaker lower bound on gap-CP.
- Finally, we remark that locally dense codes are required to be efficiently constructed in $\text{poly}(\log |C^*|)$ time, which is part of why it is hard to find. Specifically, while [23] shows that an averaging argument works for a random center, derandomizing this is a big issue and a few subsequent works are dedicated solely to this issue [18, 41]. (We also note that it remains open whether a center can be deterministically found for a variant of locally dense codes used in hardness of parameterized version of the minimum distance problem. See [12] for more details.) On the other hand, brute force search (over all codewords in $\widetilde{C}^*$) suffices to find a center for us, as we are allowed construction time of $\text{poly}(|C^*|)$.

2.3 Inapproximability of Closest Pair and Maximum Inner Product

In this subsection, we sketch our inapproximability results for MIP and CP. Both these results use the same reduction that we had from BCP to CP, except that we now need stronger properties from the gadget, i.e., the previously used notions of contact dimension does not suffice anymore. Below we sketch the required strengthening of the gadget properties and explain how to achieve them.

---

[8] Clearly, for the ball to contain more than a single codeword, it must be $r \geq \Delta/2$. Here we are interested in balls with radius not much bigger than that, say $r < \gamma \cdot \Delta$ for some constant $1/2 < \gamma < 1$.

[9] Strictly speaking, a locally dense code also requires an auxiliary matrix T used to index these codewords. However, in previous works, finding T is typically not hard given the center s. Hence, we ignore T in our discussion here for the ease of exposition.

2.3.1 Approximate Maximum Inner Product

Observe that the gadget we construct for CP in Subsection 2.2 can also be written in terms of inner product as follows: there exists a dense balanced bipartite graph $G^* = (A^* \dot\cup B^*, E^*)$, a mapping $\tau : A^* \dot\cup B^* \to \{0,1\}^{q\cdot\ell}$ such that the following holds.

- **(i)** For all edges $(a,b) \in E^*$, $\langle \tau(a), \tau(b)\rangle = \ell - r^*$.
- **(ii)** For all edges $(a,b) \in (A^* \times B^*) \setminus E^*$, $\langle \tau(a), \tau(b)\rangle < \ell - r^*$.
- **(iii)** For all distinct a, b both from A^* or both from B^*, $\langle \tau(a), \tau(b)\rangle \le \ell - \Delta$.

Notice that we wrote the conditions above in a slightly different way than in previous subsections; previously in the contact dimension notation, (ii) and (iii) would be simply written together as: for all non-edge (a,b), $\langle \tau(a), \tau(b)\rangle < \ell - r^*$. This change is intentional, since, to get gap in our reductions, we only need a gap between the bounds in (i) and (iii) (but not in (ii)). In particular, to get hardness of approximating MIP, we require $\frac{\ell - r^*}{\ell - \Delta}$ to be at least $(1+\varepsilon)$ for some $\varepsilon > 0$.

From our Reed-Solomon construction above, $\ell - \Delta$ and $\ell - r^*$ are exactly the message length of C^* minus one and the message length of $\widetilde{C}^*$ minus one respectively. Previously, we selected these two to be h and $h+1$. Now to obtain the desired gap, we simply take the larger code $\widetilde{C}^*$ to be a Reed-Solomon code with larger (i.e. $(1+\varepsilon)h$) message length[10].

Finally, we note that even with the above gadget, the reduction only gives a small (i.e. $1 + o(1)$) factor hardness of approximating MIP. To boost the gap to near polynomial, we simply tensor the vectors with themselves.

2.3.2 Approximate Closest Pair

Once again, recall that we have the following gadget from Subsection 2.2: there exists a dense balanced bipartite graph $G^* = (A^* \dot\cup B^*, E^*)$, a mapping $\tau : A^* \dot\cup B^* \to \{0,1\}^{q\cdot\ell}$ such that the following holds.

- **(i)** For all edges $(a,b) \in E^*$, $\|\tau(a) - \tau(b)\|_0 = r^*$.
- **(ii)** For all edges $(a,b) \in (A^* \times B^*) \setminus E^*$, $\|\tau(a) - \tau(b)\|_0 > r^*$.
- **(iii)** For all distinct a, b both from A^* or both from B^*, $\|\tau(a) - \tau(b)\|_0 \ge \Delta$.

Once again, we need an $(1+\varepsilon)$ gap between the bounds in (iii) and (i), i.e., $\frac{\Delta}{r^*}$. Unfortunately, we cannot construct such codes using any of the Reed-Solomon code families. We turn to another type of codes that beat the Gilbert-Varshamov bound: Algebraic- Geometric (AG) codes. Similar to the Reed-Solomon code based construction, we take C^* as an AG code and $\widetilde{C}^*$ to be a "higher degree" AG code; getting the desired gap simply means that the distance of C^* must be at least $(1+\varepsilon)$ times the distance of $\widetilde{C}^*$.

Recall from Subsection 2.2 also that, to bound the density of G^*, we need a lower bound on the number of minimum weight codewords of $\widetilde{C}^*$. Such bounds for AG codes are non-trivial and we turn to the bounds from [8, 54]. Unfortunately, this only gives G^* with density $|C^*|^{-1/2-o(1)}$, instead of $|C^*|^{-o(1)}$ as before. This is indeed the reason that our running time lower bound for approximate CP is only $n^{1.5-\varepsilon}$.

We are not aware of any result on the (asymptotic) tightness of the bounds from [8, 54] that we use. However, improving upon such bounds would have other consequences, such as a better bound on the kissing numbers of lattices constructed in [54]. As a result, it seems likely that more understanding of AG codes (and perhaps even new constructions) are needed in order to improve these bounds.

[10] This approach can in fact give not just $(1+\varepsilon)$ but arbitrarily large constant gap between the two cases. In the actual reduction, we take this gap to be 3, which makes some computations simpler.

3 Discussion and Open Questions

It remains open to completely resolve Open Questions 1 and 2. It is still possible that our framework can be used to resolve these problems: we just need to construct gadgets with better parameters! In particular, to resolve Question 2, it suffices to obtain codes which have a much larger fraction of minimum weight codewords than the state-of-the-art algebraic geometric codes while having the desirable properties of algebraic geometric codes (formalized below). This motivates us to ask the following purely coding theoretic question:

▶ **Open Question 9.** *For every $0 < \delta < 1$, are there linear codes $\mathcal{C}_1 \subseteq \mathcal{C}_2 \subseteq \mathbb{F}_q^N$ both of block length N over alphabet $\mathbb{F}_q$ such that the following holds:*

- $\Delta(\mathcal{C}_1) \geq (1 + f(\delta)) \cdot \Delta(\mathcal{C}_2)$*, for some* $f : (0, 1) \to (0, 1)$.
- $|A_{\Delta(\mathcal{C}_2)}(\mathcal{C}_2)|/|\mathcal{C}_2| \geq |\mathcal{C}_1|^{-\delta}$.

Apart from the aforementioned questions, Rubinstein [50] pointed out an interesting obstacle, aptly dubbed the "triangle inequality barrier", to obtain fine-grained lower bounds against 3-approximation algorithms for BCP (see Open Question 3 in [50]). In the case of CP, this barrier turns out to be against 2-approximation algorithms as noted in [21]. We reiterate this below as an open problem to be resolved:

▶ **Open Question 10.** *Can we show that assuming* SETH*, for some constant $\varepsilon > 0$, no algorithm running in time $n^{1+\varepsilon}$ can solve 2-*CP *in* any *metric when the points are in $\omega(\log n)$ dimensions?*

Another interesting direction is to extend the hardness of MIP to the k-vector generalization of the problem, called k-MIP. In k-MIP, we are given a set of n points $P \subseteq \mathbb{R}^d$ and we would like to select k distinct points $\mathbf{a}_1, \ldots, \mathbf{a}_k \in P$ that maximizes

$$\langle \mathbf{a}_1, \ldots, \mathbf{a}_k \rangle := \sum_{j \in [d]} (\mathbf{a}_1)_j \cdots (\mathbf{a}_k)_j.$$

It is known that the k-chromatic variant of k-MIP is hard to approximate (see Appendix B of [34]) but this is not known to be true for k-MIP itself. Our approach seems quite compatible to tackling this problem as well; in particular, if we can construct a certain (natural) generalization of our gadget for MIP, then we would immediately arrive at the inapproximability of k-MIP even for $\{0, 1\}$-entries vectors. The issue in constructing this gadget is that we are now concerned about agreements of more than two vectors, which does not correspond to error-correcting codes anymore and some additional tools are needed to argue for this more general case.

It should be noted that the hardness of approximating k-MIP for $\{0, 1\}$-entry vectors is equivalent to the *one-sided k-biclique* problem [39], in which a bipartite graph is given and the goal is to select k vertices on the right that maximize the number of their common neighbors. The equivalence can be easily seen by viewing the coordinates as the left-hand-side vertices and the vectors as the right-hand-side vertices. The one-sided k-biclique is shown to be W[1]-hard to approximate by Lin [39] who also showed a lower bound of $n^{\Omega(\sqrt{k})}$ for the problem assuming ETH. If the generalization of our gadget for k-MIP works as intended, then this lower bound can be improved to $n^{\Omega(k)}$ under ETH and even $n^{k-o(1)}$ under SETH.

The one-sided k-biclique is closely related to the (two-sided) k-biclique problem, where we are given a bipartite graph and we wish to decide whether it contains $K_{k,k}$ as a subgraph. The k-biclique problem was consider a major open problem in parameterized complexity (see e.g., [22]) until it was shown by Lin to be W[1]-hard [39]. Nevertheless, the running time

lower bound known is still not tight: currently, the best lower bound known for this problem is $n^{\Omega(\sqrt{k})}$ both for the exact version (under ETH) [39] and its approximate variant (under Gap-ETH) [14]. It remains an interesting open question to close the gap between the above lower bounds and the trivial upper bound of $n^{O(k)}$. Progresses on the one-sided k-biclique problem could lead to improved lower bounds for k-biclique problem too, although several additional steps have to be taken care of.

References

1. Amir Abboud, Aviad Rubinstein, and Ryan Williams. Distributed PCP Theorems for Hardness of Approximation in P. *CoRR*, abs/1706.06407, 2017. arXiv:1706.06407.
2. Amir Abboud, Aviad Rubinstein, and Ryan Williams. Distributed PCP Theorems for Hardness of Approximation in P. In *FOCS*, pages 25–36, 2017.
3. Pankaj K. Agarwal, Herbert Edelsbrunner, Otfried Schwarzkopf, and Emo Welzl. Euclidean Minimum Spanning Trees and Bichromatic Closest Pairs. *Discrete & Computational Geometry*, 6:407–422, 1991. Preliminary version in SoCG'90. doi:10.1007/BF02574698.
4. Nir Ailon and Bernard Chazelle. The Fast Johnson–Lindenstrauss Transform and Approximate Nearest Neighbors. *SIAM J. Comput.*, 39(1):302–322, 2009. Preliminary version in STOC'06. doi:10.1137/060673096.
5. Josh Alman, Timothy M. Chan, and R. Ryan Williams. Polynomial Representations of Threshold Functions and Algorithmic Applications. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 467–476, 2016. doi:10.1109/FOCS.2016.57.
6. Josh Alman and Ryan Williams. Probabilistic Polynomials and Hamming Nearest Neighbors. In *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 136–150, 2015. doi:10.1109/FOCS.2015.18.
7. Ethem Alpaydin. *Introduction to Machine Learning*. The MIT Press, 2nd edition, 2010.
8. Alexei E. Ashikhmin, Alexander Barg, and Serge G. Vladut. Linear Codes with Exponentially Many Light Vectors. *J. Comb. Theory, Ser. A*, 96(2):396–399, 2001. doi:10.1006/jcta.2001.3206.
9. Michael Ben-Or. Lower Bounds for Algebraic Computation Trees (Preliminary Report). In *Proceedings of the 15th Annual ACM Symposium on Theory of Computing, 25-27 April, 1983, Boston, Massachusetts, USA*, pages 80–86, 1983. doi:10.1145/800061.808735.
10. Jon Louis Bentley. Multidimensional Divide-and-Conquer. *Commun. ACM*, 23(4):214–229, 1980. doi:10.1145/358841.358850.
11. Jon Louis Bentley and Michael Ian Shamos. Divide-and-Conquer in Multidimensional Space. In *Proceedings of the 8th Annual ACM Symposium on Theory of Computing, May 3-5, 1976, Hershey, Pennsylvania, USA*, pages 220–230, 1976. doi:10.1145/800113.803652.
12. Arnab Bhattacharyya, Suprovat Ghoshal, Karthik C. S., and Pasin Manurangsi. Parameterized Intractability of Even Set and Shortest Vector Problem from Gap-ETH. In *ICALP*, pages 17:1–17:15, 2018. doi:10.4230/LIPIcs.ICALP.2018.17.
13. Yonatan Bilu and Nathan Linial. Monotone maps, sphericity and bounded second eigenvalue. *J. Comb. Theory, Ser. B*, 95(2):283–299, 2005. doi:10.1016/j.jctb.2005.04.005.
14. Parinya Chalermsook, Marek Cygan, Guy Kortsarz, Bundit Laekhanukit, Pasin Manurangsi, Danupon Nanongkai, and Luca Trevisan. From Gap-ETH to FPT-Inapproximability: Clique, Dominating Set, and More. In *FOCS*, pages 743–754, 2017. doi:10.1109/FOCS.2017.74.

15 Lijie Chen. On The Hardness of Approximate and Exact (Bichromatic) Maximum Inner Product. In *33rd Computational Complexity Conference, CCC 2018, June 22-24, 2018, San Diego, CA, USA*, pages 14:1–14:45, 2018. doi:10.4230/LIPIcs.CCC.2018.14.

16 Lijie Chen. Toward Super-Polynomial Size Lower Bounds for Depth-Two Threshold Circuits. *CoRR*, abs/1805.10698, 2018. arXiv:1805.10698.

17 Lijie Chen and Ryan Williams. An Equivalence Class for Orthogonal Vectors. *To appear in SODA*, 2019.

18 Qi Cheng and Daqing Wan. A Deterministic Reduction for the Gap Minimum Distance Problem. *IEEE Trans. Information Theory*, 58(11):6935–6941, 2012. doi:10.1109/TIT.2012.2209198.

19 Edith Cohen and David D. Lewis. Approximating Matrix Multiplication for Pattern Recognition Tasks. *J. Algorithms*, 30(2):211–252, 1999. doi:10.1006/jagm.1998.0989.

20 Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms, Third Edition*. The MIT Press, 3rd edition, 2009.

21 Roee David, Karthik C. S., and Bundit Laekhanukit. On the Complexity of Closest Pair via Polar-Pair of Point-Sets. In *34th International Symposium on Computational Geometry, SoCG 2018, June 11-14, 2018, Budapest, Hungary*, pages 28:1–28:15, 2018. doi:10.4230/LIPIcs.SoCG.2018.28.

22 Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013. doi:10.1007/978-1-4471-5559-1.

23 Ilya Dumer, Daniele Micciancio, and Madhu Sudan. Hardness of approximating the minimum distance of a linear code. *IEEE Trans. Information Theory*, 49(1):22–37, 2003. doi:10.1109/TIT.2002.806118.

24 Peter Frankl and Hiroshi Maehara. On the Contact Dimensions of Graphs. *Discrete & Computational Geometry*, 3:89–96, 1988. doi:10.1007/BF02187899.

25 François Le Gall. Powers of tensors and fast matrix multiplication. In *International Symposium on Symbolic and Algebraic Computation, ISSAC'14, Kobe, Japan, July 23-25, 2014*, pages 296–303, 2014. doi:10.1145/2608628.2608664.

26 E. N. Gilbert. A comparison of signalling alphabets. *Bell System Technical Journal*, 31:504–522, 1952.

27 Omer Gold and Micha Sharir. Dominance Products and Faster Algorithms for High-Dimensional Closest Pair under L_∞. *CoRR*, abs/1605.08107, 2016. arXiv:1605.08107.

28 Tomislav Hengl. Finding the right pixel size. *Computers & Geosciences*, 32(9):1283–1298, 2006. doi:10.1016/j.cageo.2005.11.008.

29 Klaus H. Hinrichs, Jürg Nievergelt, and Peter Schorn. Plane-Sweep Solves the Closest Pair Problem Elegantly. *Inf. Process. Lett.*, 26(5):255–261, 1988. doi:10.1016/0020-0190(88)90150-0.

30 Piotr Indyk. Dimensionality reduction techniques for proximity problems. In *Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms, January 9-11, 2000, San Francisco, CA, USA.*, pages 371–378, 2000. URL: http://dl.acm.org/citation.cfm?id=338219.338582.

31 Piotr Indyk, Moshe Lewenstein, Ohad Lipsky, and Ely Porat. Closest Pair Problems in Very High Dimensions. In *Automata, Languages and Programming: 31st International Colloquium, ICALP 2004, Turku, Finland, July 12-16, 2004. Proceedings*, pages 782–792, 2004. doi:10.1007/978-3-540-27836-8_66.

32 Piotr Indyk and Rajeev Motwani. Approximate Nearest Neighbors: Towards Removing the Curse of Dimensionality. In *Proceedings of the Thirtieth Annual ACM Symposium on the Theory of Computing, Dallas, Texas, USA, May 23-26, 1998*, pages 604–613, 1998. doi:10.1145/276698.276876.

33 William Johnson and Joram Lindenstrauss. Extensions of Lipschitz mappings into a Hilbert space. In *Conference in modern analysis and probability (New Haven, Conn., 1982)*, volume 26 of *Contemporary Mathematics*, pages 189–206. American Mathematical Society, 1984.

34 Karthik C. S., Bundit Laekhanukit, and Pasin Manurangsi. On the Parameterized Complexity of Approximating Dominating Set. In *STOC*, 2018. To appear.

35 Samir Khuller and Yossi Matias. A Simple Randomized Sieve Algorithm for the Closest-Pair Problem. *Inf. Comput.*, 118(1):34–37, 1995. doi:10.1006/inco.1995.1049.

36 Jon Kleinberg and Éva Tardos. *Algorithm Design*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2005.

37 Jon M. Kleinberg. Two Algorithms for Nearest-Neighbor Search in High Dimensions. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on the Theory of Computing, El Paso, Texas, USA, May 4-6, 1997*, pages 599–608, 1997. doi:10.1145/258533.258653.

38 Drago Krznaric, Christos Levcopoulos, and Bengt J. Nilsson. Minimum Spanning Trees in d Dimensions. *Nord. J. Comput.*, 6(4):446–461, 1999.

39 Bingkai Lin. The Parameterized Complexity of k-Biclique. In *SODA*, pages 605–615, 2015.

40 Udi Manber. *Introduction to Algorithms: A Creative Approach*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1989.

41 Daniele Micciancio. Locally Dense Codes. In *IEEE 29th Conference on Computational Complexity, CCC 2014, Vancouver, BC, Canada, June 11-13, 2014*, pages 90–97, 2014. doi:10.1109/CCC.2014.17.

42 Rajeev Motwani, Assaf Naor, and Rina Panigrahy. Lower Bounds on Locality Sensitive Hashing. *SIAM J. Discrete Math.*, 21(4):930–935, 2007. doi:10.1137/050646858.

43 Ryan O'Donnell, Yi Wu, and Yuan Zhou. Optimal Lower Bounds for Locality-Sensitive Hashing (Except When q is Tiny). *TOCT*, 6(1):5:1–5:13, 2014. doi:10.1145/2578221.

44 Janos Pach. Decomposition of multiple packing and covering. *Diskrete Geometrie*, 2 Kolloq. Math. Inst. Univ. Salzburg:169–178, 1980.

45 Franco P. Preparata and Michael I. Shamos. *Computational Geometry: An Introduction*. Springer-Verlag New York, Inc., New York, NY, USA, 1985.

46 Michael O. Rabin. Probabilistic Algorithms. In *Proceedings of a Symposium on New Directions and Recent Results in Algorithms and Complexity, Computer Science Department, Carnegie-Mellon University, April 7-9, 1976*, pages 21–39, 1976.

47 Ilya Razenshteyn. High-Dimensional Similarity Search and Sketching: Algorithms and Hardness. *PhD Thesis, MIT*, 2017.

48 Irving S. Reed and Gustave Solomon. Polynomial Codes over Certain Finite Fields. *Journal of the Society for Industrial and Applied Mathematics (SIAM)*, 8(2):300–304, 1960.

49 Jan Reiterman, Vojtech Rödl, and Edita Sinajová. Embeddings of Graphs in Euclidean Spaces. *Discrete & Computational Geometry*, 4:349–364, 1989. doi:10.1007/BF02187736.

50 Aviad Rubinstein. Hardness of approximate nearest neighbor search. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, pages 1260–1268, 2018. doi:10.1145/3188745.3188916.

51 Michael Ian Shamos and Dan Hoey. Closest-Point Problems. In *16th Annual Symposium on Foundations of Computer Science, Berkeley, California, USA, October 13-15, 1975*, pages 151–162, 1975. doi:10.1109/SFCS.1975.8.

52 Gregory Valiant. Finding Correlations in Subquadratic Time, with Applications to Learning Parities and the Closest Pair Problem. *J. ACM*, 62(2):13:1–13:45, 2015. doi:10.1145/2728167.

53 R. R. Varshamov. Estimate of the number of signals in error correcting codes. *Dokl. Akad. Nauk SSSR*, 117:739–741, 1957.

54 Serge Vlăduţ. Lattices with exponentially large kissing numbers. *arXiv preprint*, 2018. arXiv:1802.00886.

55 Ryan Williams. On the Difference Between Closest, Furthest, and Orthogonal Pairs: Nearly-Linear vs Barely-Subquadratic Complexity. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018*, pages 1207–1215, 2018. doi:10.1137/1.9781611975031.78.

56 Raymond Chi-Wing Wong, Yufei Tao, Ada Wai-Chee Fu, and Xiaokui Xiao. On Efficient Spatial Matching. In *Proceedings of the 33rd International Conference on Very Large Data Bases, University of Vienna, Austria, September 23-27, 2007*, pages 579–590, 2007. URL: http://www.vldb.org/conf/2007/papers/research/p579-wong.pdf.

57 Andrew Chi-Chih Yao. Lower Bounds for Algebraic Computation Trees with Integer Inputs. *SIAM J. Comput.*, 20(4):655–668, 1991. Preliminary version in FOCS'89. doi:10.1137/0220041.

58 Charles T. Zahn. Graph-Theoretical Methods for Detecting and Describing Gestalt Clusters. *IEEE Trans. Computers*, 20(1):68–86, 1971. doi:10.1109/T-C.1971.223083.