HAL open science

All non-trivial variants of 3-LDT are equivalent

Bartlomiej Dudek, Pawel Gawrychowski, Tatiana Starikovskaya

To cite this version:

Bartlomiej Dudek, Pawel Gawrychowski, Tatiana Starikovskaya. All non-trivial variants of 3-LDT are equivalent. Proceedings of the 52nd annual ACM SIGACT Symposium on Theory of Computing (STOC) 2020, 2020, Chicago, United States. hal-03942980

HAL Id: hal-03942980

https://hal.science/hal-03942980v1

Submitted on 17 Jan 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

arXiv:2001.01289v1 [cs.DS] 5 Jan 2020

All non-trivial variants of 3-LDT are equivalent

Bartłomiej Dudek[1], Paweł Gawrychowski[1], and Tatiana Starikovskaya[2]

[1]Institute of Computer Science, University of Wrocław, Poland
{bartlomiej.dudek,gawry}@cs.uni.wroc.pl

[2]DIENS, École normale supérieure, PSL Research University, France
tat.starikovskaya@gmail.com

Abstract

The popular 3-SUM conjecture states that there is no strongly subquadratic time algorithm for checking if a given set of integers contains three distinct elements that sum up to zero. A closely related problem is to check if a given set of integers contains distinct x_1, x_2, x_3 such that $x_1 + x_2 = 2x_3$. This can be reduced to 3-SUM in almost-linear time, but surprisingly a reverse reduction establishing 3-SUM hardness was not known.

We provide such a reduction, thus resolving an open question of Erickson [23]. In fact, we consider a more general problem called 3-LDT parameterized by integer parameters $\alpha_1, \alpha_2, \alpha_3$ and t. In this problem, we need to check if a given set of integers contains distinct elements x_1, x_2, x_3 such that $\alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_3 = t$. For some combinations of the parameters, every instance of this problem is a NO-instance or there exists a simple almost-linear time algorithm. We call such variants trivial. We prove that all non-trivial variants of 3-LDT are equivalent under subquadratic reductions. Our main technical contribution is an efficient deterministic procedure based on the famous Behrend's construction that partitions a given set of integers into few subsets that avoid a chosen linear equation.

1 Introduction

The well-known 3-SUM problem is to decide, given a set X of n integers, whether any three distinct elements of X sum up to zero. This can be easily solved in quadratic time by first sorting X and then scanning the sorted sequence with two pointers. For many years no faster algorithm was known, and it was conjectured that no significantly faster algorithm exists. This assumption led to strong lower bounds for multiple problems in computational geometry [28] and, more recently, became a central problem in the field of fine-grained complexity [43]. Furthermore, it has been proven that in some restricted models of computation 3-SUM requires $\Omega(n^2)$ time [5, 24].

However, in 2014 Grønlund and Pettie [32] proved that the decision tree complexity of 3-SUM is only $\mathcal{O}(n^{1.5}\sqrt{\log n})$, which ruled out any almost quadratic-time lower bounds in the decision tree model. This was later improved by Gold and Sharir to $\mathcal{O}(n^{1.5})$ [29] and finally to $\mathcal{O}(n \log^2 n)$ by Kane et al. [34]. The upper bounds for the decision tree model were later used to design a series of algorithms for a harder version of 3-SUM where the set X can contain real numbers. Grønlund and Pettie [32] derived an $\mathcal{O}(n^2(\log\log n)^2/\log n)$ time randomized algorithm and an $\mathcal{O}(n^2(\log\log n/\log n)^{2/3})$ time deterministic algorithm for the real RAM model. The deterministic bound was soon improved to $\mathcal{O}(n^2 \log\log n/\log n)$ by Gold and Sharir [29] and (independently) Freund [27] and then to $\mathcal{O}(n^2(\log\log n)^{\mathcal{O}(1)}/\log^2 n)$ by Chan [16]. These results immediately imply similar bounds for the integer version of 3-SUM. In the word RAM model with machine words of size w, Baran et al. [11] provided an algorithm with $\mathcal{O}(n^2/\max\{\frac{w}{\log^2 w}, \frac{\log^2 n}{(\log\log n)^2}\})$ expected time.

Even though asymptotically faster than $\mathcal{O}(n^2)$, these algorithms are not strongly subquadratic, meaning working in $\mathcal{O}(n^{2-\varepsilon})$ time, for some $\varepsilon > 0$. This motivates the popular modern version of the conjecture, which is that the 3-SUM problem cannot be solved in $\mathcal{O}(n^{2-\varepsilon})$ time (even in expectation), for any $\varepsilon > 0$, on the word RAM model with words of size $\mathcal{O}(\log n)$ [38]. By now we have multiple examples of other problems that can be shown to be hard assuming this conjecture, especially in geometry [4, 8–10, 13, 15, 18, 19, 25, 26, 28, 37, 42], but also in dynamic data structures [2, 35, 38], string algorithms [3, 7], finding exact weight subgraphs [1, 45] and finally in partial matrix multiplication and reporting variants of convolution [30].

In particular, it is well-known that the 3-SUM problem defined above is subquadratically equivalent to its 3-partite variant, where we are given three sets A_1, A_2, A_3 containing n integers each, and must decide whether there is $x_1 \in A_1$, $x_2 \in A_2$, and $x_3 \in A_3$ such that $x_1 + x_2 + x_3 = 0$. To reduce 3-partite 3-SUM to 1-partite, we can add a multiple of some sufficiently big number M to all elements in every set and take the union, for example:

$$X = \{3M + x : x \in A_1\} \cup \{M + x : x \in A_2\} \cup \{-4M + x : x \in A_3\}$$

M is chosen so that the only possibility for the three elements of X to sum up to 0 is that they correspond to three elements belonging to distinct sets A_1, A_2, and A_3. To show the reduction from 1-partite 3-SUM to 3-partite, a natural approach would be to take $A_1 = A_2 = A_3 = X$. However, this approach does not quite work as in the 1-partite variant we desire x_1, x_2, x_3 to be *distinct*. In the folklore reduction, this technicality is overcome using the so-called color-coding technique by Alon et al. [6].

A natural generalization of 3-SUM is 3-variate linear degeneracy testing, or 3-LDT [5]. In this problem, we are given a set X of n integers, integer parameters $\alpha_1, \alpha_2, \alpha_3$ and t, and must decide whether there are 3 distinct numbers $x_1, x_2, x_3 \in X$ such that $\sum_{i=1}^{3} \alpha_i x_i = t$. Similar to 3-SUM, the 3-LDT problem can be considered in the 3-partite variant as well.

A particularly natural variant of the 1-partite 3-LDT problem is as follows: given a set X of n numbers, are there three distinct $x_1, x_2, x_3 \in X$ such that $x_1 + x_2 - 2x_3 = 0$? Following

Erickson [23], we call this problem AVERAGE. It is easy to see that AVERAGE reduces to $\mathcal{O}(\log n)$ instances of 3-partite 3-SUM where the j-th instance consists of the sets $A_j, X \setminus A_j, -2X$ where $A_j = \{x_i \in X : \text{ the } j\text{-th bit of } i \text{ is } 1\}$ (and $X = \{x_1, \ldots, x_n\}$). However, a reverse reduction seems more elusive and in fact according to Erickson [23] it is not known whether AVERAGE is 3-SUM-hard[1]. This suggests the following question.

Question 1.1. *Can we design a reduction from 3-SUM to AVERAGE? Or is AVERAGE easier than 3-SUM?*

A more ambitious question would be to provide a complete characterisation of all variants of 3-LDT parameterized by $\alpha_1, \alpha_2, \alpha_3, t$. We know that in the restricted 3-linear decision tree model solving every variant where all coefficients α_i are nonzero requires quadratic time [5, 24], but by now we know that this model is not necessarily the most appropriate for such problems.

Question 1.2. *Which variants of 3-LDT are easier than others? Or are they all equivalent?*

Formal definitions of 3-LDT, 3-SUM, and AVERAGE. We are interested in connecting the complexity of AVERAGE and more generally speaking any variant of 3-LDT to the 3-SUM conjecture, and so from now on assume that the input consists of integers. 3-SUM is widely believed to be hard even for polynomial universes, because one can always hash down the universe to $\{-n^3, \ldots, n^3\}$ while maintaining the expected running time [38] (although a deterministic reduction is not known). In fact, the so-called strong 3-SUM conjecture stipulates that there is no subquadratic time algorithm even if the universe is $\{-n^2, \ldots, n^2\}$. A similar randomized reduction can be applied to any variant of the 3-LDT problem, so for concreteness we will assume that the universe is $\{-n^3, \ldots, n^3\}$ and work with the following formulation:

1-partite 3-LDT$(1, \bar{\alpha}, t)$ **Parameters:** Integer coefficients $\alpha_1, \alpha_2, \alpha_3$ and t. **Input:** Set $X \subseteq \{-n^3, \ldots, n^3\}$ of size n. **Output:** Are there distinct $x_1, x_2, x_3 \in X$ such that $\sum_{i=1}^{3} \alpha_i x_i = t$?

3-partite 3-LDT$(3, \bar{\alpha}, t)$ **Parameters:** Integer coefficients $\alpha_1, \alpha_2, \alpha_3$ and t. **Input:** Sets $A_1, A_2, A_3 \subseteq \{-n^3, \ldots, n^3\}$ of size n. **Output:** Are there $x_1 \in A_1, x_2 \in A_2, x_3 \in A_3$ such that $\sum_{i=1}^{3} \alpha_i x_i = t$?

The 3-SUM problem is defined as 3-LDT$(p, \bar{\alpha}, 0)$, where $p = 1$ or $p = 3$ depending on the partity, and $\bar{\alpha} = (1, 1, 1)$. The AVERAGE problem, introduced by Erickson [22], is defined as 3-LDT$(1, \bar{\alpha}, 0)$, where $\bar{\alpha} = (1, 1, -2)$.

Our contribution and techniques. We start by introducing a definition that plays a crucial role in the characterization of the 3-LDT variants.

Definition 1.3. *We call a variant 3-LDT$(p; \bar{\alpha}; t)$ of the 3-LDT problem* trivial*, if either*

1. *Any of the coefficients α_i is zero, or*
2. *$t \neq 0$ and $\gcd(\alpha_1, \alpha_2, \alpha_3) \nmid t$*

[1] Also see https://cs.stackexchange.com/questions/10681/is-detecting-doubly-arithmetic-progressions-3sum-hard/10725#10725

and otherwise non-trivial.

Notice that if any of the coefficients is 0, then we need to find at most two numbers satisfying a linear relation, which can be done in total $\mathcal{O}(n \log n)$ time by first sorting and then scanning the sorted sequence with two pointers. Also, if both $t \neq 0$ and $\gcd(\alpha_1, \alpha_2, \alpha_3) \nmid t$ then every instance of such a variant is a NO-instance, and we can return the answer in constant time. Our main contribution is a series of deterministic subquadratic reductions establishing the following theorem.

Theorem 1.4. *All non-trivial variants (1- and 3-partite) of 3-LDT are subquadratic-equivalent.*

In particular, this implies the following.

Corollary 1.5. *AVERAGE is subquadratic-equivalent to 3-SUM.*

Thus, we completely resolve both Question 1.1 and Question 1.2.

To design the most interesting of our reductions, from 3-LDT$(3; \bar{\alpha}; 0)$ to 3-LDT$(1; \bar{\alpha}; 0)$, we need the following notion. We call a set $S \subseteq [n]$ *progression-free* if it contains no non-trivial arithmetic progression, that is, three distinct elements a, b, c such that $a + b - 2c = 0$. Erdős and Turan [21] introduced the question of exhibiting a dense subset with such a property, and presented a construction with $\Omega(n^{\log_3 2})$ elements. This was improved by Salem and Spencer [40] to $n^{1-\mathcal{O}(1/\log\log n)}$, and then again by Behrend [14] to $\Omega(n/2^{2\sqrt{2}\cdot\sqrt{\log n}} \cdot \log^{1/4} n)$. More recently, Elkin [20] showed how to construct a set consisting of $\Omega(n \log^{1/4} n/2^{2\sqrt{2}\cdot\sqrt{\log n}})$ elements. One could naturally ask for a dense subset that avoids a certain linear equation $\alpha_1 x_1 + \alpha_2 x_2 = (\alpha_1 + \alpha_2) x_3$, where α_1, α_2 are positive integers. Indeed, it turns out that Behrend's argument works with minor modifications also for such equations [39, Theorem 2.3]. We use Behrend's set to partition an arbitrary set into a small number of progression-free sets. The main idea is that we can deterministically choose a shift so that the intersection of Behrend's set and a given set is a large progression-free set, which is always possible due to the density of Behrend's set.

Related work. Surprisingly few reductions to 3-SUM are known. It is known that 3-SUM is equivalent to convolution 3-SUM [38], which is widely used in the proofs of 3-SUM-hardness. In addition, Jafargholi and Viola [33] showed that solving 3-SUM in $\tilde{\mathcal{O}}(n^{1+\varepsilon})$ time for some $\varepsilon < 1/15$ would lead to a surprising algorithm for triangle listing.

Other variants of 3-SUM have been also considered. Among them are clustered 3-SUM and 3-SUM for monotone sets in 2D that are surprisingly solvable in truly subquadratic time [17]; algebraic 3-SUM, a generalization which replaces the sum function by a constant-degree polynomial [12]; and 3-SUM$^+$ in which, given three sets A, B, S one needs to return $(A + B) \cap S$ [11, 32]. An interesting generalisation of the 3-SUM conjecture states that there is no algorithm preprocessing two lists of n elements A, B in $n^{2-\Omega(1)}$ time and answering queries "Does c belong to $A + B$?" in $n^{1-\Omega(1)}$ time. Very recently, this conjecture was falsified in two independent papers [31, 36].

2 Preliminaries

Notation. Below we use the following notation: $\bar{x}$ denotes the triple (x_1, x_2, x_3), when omitting superscript in the sum $\sum_i$ we mean all possible values of i; $[k] = \{1, 2, \ldots, k\}$; $f[A] = \{f(a) : a \in A\}$ is the image of f over A; adding a number to a set denotes adding the number to all elements from the set $A + x = \{a + x : a \in A\}$. All numbers in the considered problems and reductions are integer. All reductions, unless said otherwise, are deterministic.

Subquadratic reductions. We provide the formal definition of a subquadratic reduction:

Definition 2.1 (cf. [44])**.** *Let A and B be computational problems with a common size measure m on inputs. We say that there is a* subquadratic reduction *from A to B if there is an algorithm $\mathcal{A}$ with oracle access to B, such that for every $\varepsilon > 0$ there is $\delta > 0$ satisfying three properties:*

1. *For every instance x of A, $\mathcal{A}(x)$ solves the problem A on x.*
2. *$\mathcal{A}$ runs in $\mathcal{O}(m^{2-\delta})$ time on instances of size m.*
3. *For every instance x of A of size m, let m_i be the size of the ith oracle call to B in $\mathcal{A}(x)$. Then $\sum_i m_i^{2-\varepsilon} \leq m^{2-\delta}$.*

We use the notation $A \leq_2 B$ to denote the existence of a subquadratic reduction from A to B. If $A \leq_2 B$ and $B \leq_2 A$, we say that A and B are subquadratic-equivalent.

Size of the universe. Some of our reductions increase the size of the universe by a constant factor. This is possibly an issue, as in any instance of 3-LDT on n numbers we want to keep the universe $\{-n^3, \ldots, n^3\}$. To mitigate this, in Section 4 we design a simple reduction that decreases the size of the universe by a constant factor.

We start with the following preliminary lemma.

Lemma 2.2. *All non-trivial 3-partite variants of 3-LDT are subquadratic-equivalent.*

Proof. We need to show a reduction between any two non-trivial 3-partite variants of 3-LDT. To this end, we establish three reductions: 3-LDT$(3;\bar{\alpha};0) \leq_2$ 3-LDT$(3;\bar{\alpha};t)$, 3-LDT$(3;\bar{\alpha};t) \leq_2$ 3-LDT$(3;\bar{\alpha};0)$ and 3-LDT$(3;\bar{\alpha};0) \leq_2$ 3-LDT$(3;\bar{\beta};0)$ for all possible $\bar{\alpha}, \bar{\beta}$ and $t \neq 0$. Reductions between other variants can be obtained by combining at most three of the above.

1. 3-LDT$(3;\bar{\alpha};0) \leq_2$ 3-LDT$(3;\bar{\alpha};t)$. We have $\gcd(\alpha_1, \alpha_2, \alpha_3)|t$ because 3-LDT$(3;\bar{\alpha};t)$ is a non-trivial variant of 3-LDT, so by the Chinese remainder theorem there exist an integer triple $\bar{y}$ such that $\sum_i \alpha_i y_i = t$. Given the three sets A_1, A_2, A_3 we construct an instance of 3-LDT$(3;\bar{\alpha};t)$ consisting of three sets A'_1, A'_2, A'_3 where $A'_i = \{x + y_i : x \in A_i\}$. Then there is $\bar{x} \in A_1 \times A_2 \times A_3$ satisfying $\sum_i \alpha_i x_i = 0$ iff there is $\bar{x}' \in A'_1 \times A'_2 \times A'_3$ satisfying $\sum_i \alpha_i x'_i = t$.
2. 3-LDT$(3;\bar{\alpha};t) \leq_2$ 3-LDT$(3;\bar{\alpha};0)$ as above but subtracting the y_i terms.
3. 3-LDT$(3;\bar{\alpha};0) \leq_2$ 3-LDT$(3;\bar{\beta};0)$. Define $q = \mathrm{lcm}(\beta_1, \beta_2, \beta_3)$ so that $\frac{\alpha_i q}{\beta_i}$ is an integer. In the reduction we set $A'_i = \{x\frac{\alpha_i q}{\beta_i} : x \in A_i\}$. □

By the above lemma, to prove Theorem 1.4 we only need to establish an equivalence between 1- and 3-partite variants with the same coefficients $\bar{\alpha}$ and t. To show 3-LDT$(1;\bar{\alpha};t) \leq_2$ 3-LDT$(3;\bar{\alpha};t)$, we can apply the folklore reduction from 1-partite 3-SUM to 3-partite 3-SUM based on the color-coding technique of Alon et al. [6]. As a corollary, we obtain the following result (we provide the proof in the appendix for completeness).

Theorem 2.3. *For all coefficients $\bar{\alpha}$ and t, we have 3-LDT$(1;\bar{\alpha};t) \leq_2$ 3-LDT$(3;\bar{\alpha};t)$.*

3 From 3-partite to 1-partite

In this section, we show how to reduce an arbitrary non-trivial 3-partite variant of 3-LDT to a 1-partite one with the same coefficients $\bar{\alpha}$ and t.

Let C be a sufficiently big number to be fixed later. We would like to construct the set X by setting $X = \bigcup_i \{Cx + \gamma_i : x \in A_i\}$, where $\bar{\gamma}$ are coefficients chosen so as to ensure that all triples $\bar{x}$ consisting of distinct elements from X satisfying $\sum_i \alpha_i x_i = t$ also satisfy that x_i corresponds to an element of A_i, for $i \in [3]$.

Combination is a function $f : [3] \to [3]$ encoding that x_i corresponds to an element of $A_{f(i)}$, for every $i \in [3]$. For example, suppose that x_1 comes from set A_2 and x_2 and x_3 from A_3, then $f(1) = 2$ and $f(2) = f(3) = 3$. The desired combination of elements from X satisfying $\sum_i \alpha_i x_i = t$ a triple $\bar{x}$ is the identity ($f(i) = i$), so in particular we want to forbid using more than one number from the same set ($|\{f(i) : i \in [3]\}| < 3$). However, some coefficients from $\bar{\alpha}$ might be equal, so we need also to allow combinations in which we permute the elements with the same values of α_i. This is formalized in the following definition.

Definition 3.1. *For any coefficients $\bar{\alpha}$, we call a combination f* allowed *if $\forall i \in [3]\{f(x) : x \in [3], \alpha_x = \alpha_i\} = \{x : x \in [3], \alpha_x = \alpha_i\}$, and otherwise we call it* forbidden*. In addition, if for some fixed $j \in [3]$, for all $i \in [3]$ we have $f(i) = j$, then we call the combination* constant*.*

Now we show that it is always possible to find a triple $\bar{\gamma}$ which excludes solutions from most of the forbidden combinations. Clearly, we need to ensure that in the allowed combination the C-part cancels out, so we require that $\sum_i \alpha_i \gamma_i = 0$.

Lemma 3.2. *For any triple $\bar{\alpha}$ of nonzero coefficients there exists a triple $\bar{\gamma}$ of nonzero coefficients such that $\sum_i \alpha_i \gamma_i = 0$ and for every non-constant forbidden combination f we have $\sum_i \alpha_i \gamma_{f(i)} \neq 0$.*

Proof. Consider the 3-dimensional space $\mathbb{R}^3$. The set of all triples $\bar{\gamma}$ such that $\sum_i \alpha_i \gamma_i = 0$ spans a plane Γ_{id} there. There are less than $3^3 = \mathcal{O}(1)$ non-constant forbidden combinations f and each of them gives an equation $\sum_i \alpha_i \gamma_{f(i)} = 0$ that must be avoided, which also corresponds to a forbidden plane Γ_f. By the definition of a forbidden combination f, we have $\Gamma_f \neq \Gamma_{id}$. Next, as we need all the coefficients γ_i to be nonzero, we add forbidden planes $\Gamma_i = \{\bar{\gamma} : \gamma_i = 0\}$, for $i \in [3]$. Clearly, $\Gamma_i \neq \Gamma_{id}$ because the coefficients $\bar{\alpha}$ are nonzero. Then let $\mathcal{F} = \{\Gamma_f : f \text{ is non-constant and forbidden}\} \cup \{\Gamma_i : i \in [3]\}$ be the set of all forbidden planes. Now we need to show that $\Gamma_{id} \setminus \bigcup_{F \in \mathcal{F}} F \neq \emptyset$.

Clearly all planes $f \in \mathcal{F}$ contain the origin $o = (0, 0, 0)$. Consider an arbitrary line $\ell \in \Gamma_{id}$ that does not pass through the origin o and contains infinitely many points with all coordinates rational. For example, we can take the line passing through $(1, 0, -\alpha_1/\alpha_3)$ and $(0, 1, -\alpha_2/\alpha_3)$. Observe that for any $F \in \mathcal{F}$, $\Gamma_{id} \cap F \neq \ell$, because otherwise there would be three non-collinear points (two from ℓ and o) belonging to two distinct planes Γ_{id} and F. Hence $\ell \cap F$ is either empty or a point. Recall that there is a constant number of planes in $\mathcal{F}$. Then $\Gamma_{id} \setminus \bigcup_{F \in \mathcal{F}} F \supseteq \ell \setminus \bigcup_{F \in \mathcal{F}} F$ contains some point with rational coordinates, because there are infinitely many such points on ℓ. This gives us a point in $\mathbb{Q}^3$ that belongs to Γ_{id} and does not belong to any $F \in \mathcal{F}$. By scaling its coordinates, we obtain $\bar{\gamma}$. □

The case of $t \neq 0$. First we show the reduction for the case of $t \neq 0$.

Lemma 3.3. *Assume $t \neq 0$. Any non-trivial 3-partite variant of 3-LDT with coefficients $\bar{\alpha}$ and t can be subquadratically reduced to a 1-partite one with the same coefficients $\bar{\alpha}$ and t.*

Proof. From Lemma 2.2, we know that 3-$\mathsf{LDT}(3;\bar{\alpha};t) \leq_2$ 3-$\mathsf{LDT}(3;\bar{\alpha};0)$, so it suffices to show that 3-$\mathsf{LDT}(3;\bar{\alpha};0) \leq_2$ 3-$\mathsf{LDT}(1;\bar{\alpha};t)$. By the Chinese remainder theorem, we can choose an integer triple $\bar{y}$ such that $\sum_i \alpha_i y_i = t$, and apply Lemma 3.2 on $\bar{\alpha}$ to obtain $\bar{\gamma}$. We construct the set X as follows:

$$X = \bigcup_i \{C^2 x + C\gamma_i + y_i : x \in A_i\},$$

where C is a sufficiently big constant such that the absolute value of any linear combination of γ's or y's with coefficients α_i is smaller than C (formally, we can take $C = 1 + (\max_i \max\{|\gamma_i|, |y_i|\}) \cdot \sum_i |\alpha_i|$). If there is a triple $\bar{x}$ such that $x_i \in A_i$ and $\sum_i \alpha_i x_i = 0$, then by the choice of $\bar{\gamma}$ and $\bar{y}$ we have $\sum_i \alpha_i z_i = t$, where $z_i = C^2 x_i + C\gamma_i + y_i$.

Now consider a triple $\bar{z}$ such that $z_i \in X$ and $\sum \alpha_i z_i = t$. Let $z_i = C^2 x_{f(i)} + C\gamma_{f(i)} + y_{f(i)}$, where $x_{f(i)} \in A_{f(i)}$. By the definition of C and the fact that $\sum_i \alpha_i z_i = t$, it holds that $\sum \alpha_i x_{f(i)} = 0$, $\sum \alpha_i \gamma_{f(i)} = 0$ and $\sum \alpha_i y_{f(i)} = t$. We will show that f is an allowed combination which guarantees that $x_{f(1)}, x_{f(2)}, x_{f(3)}$ is a valid solution of 3-$\mathsf{LDT}(3;\bar{\alpha};0)$.

By Lemma 3.2, $\sum_i \alpha_i \gamma_{f(i)} = 0$ implies that the combination f is either constant or allowed. If f is constant, i.e. for some fixed $j \in [3]$, $f(i) = j$ for all $i \in [3]$, from $\sum_i \alpha_i \gamma_{f(i)} = 0$ we have $\sum_i \alpha_i \gamma_j = 0$ and therefore $\sum_i \alpha_i = 0$ as $\gamma_j \neq 0$. It implies $\sum_i \alpha_i y_{f(i)} = \sum_i \alpha_i y_j = 0 \neq t$, hence f cannot be constant and is allowed. □

The case of $t = 0$. For $t = 0$ we would like to proceed as in the proof of Lemma 3.3. We apply Lemma 3.2 on $\bar{\alpha}$ to obtain $\bar{\gamma}$ and construct the set $X = \bigcup_i \{Cx + \gamma_i : x \in A_i\}$. This is enough to exclude all non-constant forbidden combinations and, if $\sum_i \alpha_i \neq 0$, also the constant combinations. However, if $\sum_i \alpha_i = 0$ then we cannot exclude the constant combinations. In other words, no matter what the chosen γ's are we are not able to exclude the solutions that use three distinct elements corresponding to the elements of the same set A_j. This suggests that we should partition each of the sets A_j into a few sets that contain no triple $\bar{x}$ of distinct elements such that $\sum_i \alpha_i x_i = 0$.

Definition 3.4. *For any $\gamma, \delta > 0$, a set X is* (γ,δ)-free *if no three distinct elements $a, b, c \in X$ satisfy $\gamma a + \delta b = (\gamma + \delta)c$.*

Lemma 3.5. *For any $N, \gamma, \delta > 0$, there exists a collection of (γ,δ)-free sets $S_1, S_2, \ldots, S_c \subseteq [N]$ where $c = 2^{\mathcal{O}(\sqrt{\log N})}$ and $\bigcup_i S_i = [N]$.*

Proof. By the result of Behrend [14], there exists a $(1,1)$-free set $Q \subseteq [N]$ of size N/w for $w = \mathcal{O}(2^{2\sqrt{2}\cdot\sqrt{\log N}} \cdot \log^{1/4} N)$. In other words, this set contains no three elements forming an arithmetic progression. As mentioned by Ruzsa [39, Theorem 2.3], this construction can be modified for any parameters $\gamma, \delta > 0$ in such a way that Q is (γ,δ)-free at the expense of setting $w = 2^{\mathcal{O}(\sqrt{\log N})}$.

Next, we choose c numbers Δ_i from $[-(N-1), N-1]$ uniformly at random and create sets $S_i := Q + \Delta_i = \{x + \Delta_i : x \in Q\}$. For every $y \in [N]$, the probability that $y \in S_i$ is $\frac{|Q|}{2N-1} \geq 1/2w$, so the probability that $y \notin \bigcup_i S_i$ is at most $(1 - 1/2w)^c < 1/N$ for sufficiently large $c = \mathcal{O}(w \log N) = 2^{\mathcal{O}(\sqrt{\log N})}$. By the union bound, there exists a collection of sets with the sought properties. □

We now explain how to partition the sets A_i into a few sets that contain no triple $\bar{x}$ consisting of distinct elements such that $\sum_i \alpha_i x_i = 0$. Recall that it suffices to focus on the case $\sum_i \alpha_i = 0$. In this case, the sets should not contain a triple $\bar{x}$ consisting of distinct elements such that $\alpha_{j_1} x_{j_1} + \alpha_{j_2} x_{j_2} = (\alpha_{j_1} + \alpha_{j_2}) x_3$, where $\alpha_{j_1}, \alpha_{j_2} > 0$ and j_1, j_2, j_3 is a permutation of $[3]$.

We put $N = 2n^3 + 1$, $\gamma = \alpha_{j_1}$ and $\delta = \alpha_{j_2}$ and consider the sets $S_1, S_2, \ldots, S_c \subset [N]$ of Lemma 3.5. We shift the sets by $-n^3 - 1$ so that they cover $[-n^3, n^3]$ while preserving that they are (γ, δ)-free. We partition every set A_i into c subsets $A_{i,j} := A_i \cap S_j$ such that every $A_{i,j}$ is (γ, δ)-free. We have $c = 2^{\mathcal{O}(\sqrt{\log n})}$. We reduce the 3-$\mathsf{LDT}(3; \bar{\alpha}; 0)$ instance of 3-LDT to c^3 instances of 3-$\mathsf{LDT}(1; \bar{\alpha}; 0)$ by considering all possible combinations of the subsets. To show that the reduction is subquadratic, we must analyze the sizes of the obtained instances. As $2^{\mathcal{O}(\sqrt{\log n})} < n^{\varepsilon'}$ for any $\varepsilon' > 0$, $2^{\mathcal{O}(\sqrt{\log n})} \cdot n^{2-\varepsilon} < n^{2-\delta}$ for all $0 < \delta < \varepsilon$. Hence, to prove the following theorem it remains to show how to efficiently (and deterministically) construct the sets $A_{i,j}$.

Theorem 3.6. *For all nonzero coefficients $\bar{\alpha}$, we have 3-$\mathsf{LDT}(3; \bar{\alpha}; 0) \leq_2$ 3-$\mathsf{LDT}(1; \bar{\alpha}; 0)$.*

3.1 Derandomization and efficient implementation of Lemma 3.5

Since $N = 2n^3 + 1$, we cannot compute the set Q and the sets S_i explicitly. Instead, we will construct a family of (γ, δ)-free sets $Q_0, Q_1, \ldots, Q_R \subseteq [N]$ with a guarantee that at least one of them is large, and we will show that for any sets A_i and Q_r we can efficiently construct the shift Δ which maximizes $|A_i \cap (Q_r + \Delta)|$. Behrend's existential proof actually implies the following statement:

Lemma 3.7 (cf. [14, 39])**.** *For any $\gamma, \delta > 0$ there exist (γ, δ)-free sets $Q_0, Q_1, \ldots, Q_R \subseteq [N]$ such that $R = 2^{\mathcal{O}(\sqrt{\log N})}$ and $|Q_r| \geq N/2^{\mathcal{O}(\sqrt{\log N})}$ for some r.*

Proof. (Sketch, see the appendix for details.) Let p be the smallest power of two larger than $\gamma + \delta$. Let $d = \lfloor \sqrt{\log_p N} \rfloor$ and $m = p^{d-1}$. The sets Q_r, $0 \leq r \leq d(m-1)^2$, are defined as follows:

$$Q_r = \{x = \sum_{i=0}^{d-1} x_i (pm)^i : \sum_{i=0}^{d-1} x_i^2 = r \text{ and } 0 \leq x_i < m \text{ for all } 0 \leq i \leq d-1\}$$

The sets Q_r are (γ, δ)-free and at least one of them is larger than $N/2^{\mathcal{O}(\sqrt{\log N})}$. □

Consider a set $A = A_i \subseteq [-N, N]$. To partition A into a small number of (γ, δ)-free sets, we will show how to efficiently find r and Δ such that $(Q_r + \Delta) \cap A$ is large. By iterating this procedure on the remaining part of A sufficiently many times, we will obtain the desired partition. We will start by showing two technical lemmas.

Note that given x and r there is a deterministic algorithm that can check if $x \in Q_r$ in constant space and $\mathcal{O}(\sqrt{\log N})$ time, as it is enough to represent a number in base pm and check the conditions of Lemma 3.7. We will also exploit the following generalization of this algorithm:

Lemma 3.8. *For any r, x, and k, there is a deterministic algorithm that calculates $|Q_r \cap [x, x+2^k)|$ in $2^{\mathcal{O}(\sqrt{\log N})}$ time and space.*

Proof. Recall from Lemma 3.7 that $pm = 2^{k'}$ for some integer k'. We must compute the number of $y \in [0, 2^k)$ such that $x + y \in Q_r$. Let $k = A \cdot k' + B$, where $B < k'$. Then the base-$2^{k'}$ representation of possible numbers y has either $A' = A$ (if $B = 0$) or $A' = A + 1$ (if $B > 0$) digits, where each digit is in $[0, 2^{k'})$.

Let $x = x_{d-1} x_{d-2} \ldots x_0$. To compute $|Q_r \cap [x, x + 2^k)| = |\{y \in [0, 2^k) : x + y \in Q_r\}|$, we will use a dynamic programming algorithm over the digits of y. Let $dp[i, c, s]$ be the number of y's from $[0, 2^{ik'})$ such that all i least significant digits of $x + y$ are smaller than m, their sum of squares

is $0 \le s \le d(m-1)^2$ and there is a carry $c \in \{0,1\}$ to the $(i+1)$-th digit of $x+y$. Clearly, $dp[0,0,0] = 1$ and $dp[0,\cdot,\cdot] = 0$ for the remaining entries. Then:

$$dp[i,c,s] = \sum_{y' \in [0,2^{k'}), c' \in \{0,1\}} dp[i-1, c', s - \sigma^2_{x_i,y',c',c}] \cdot [\sigma_{x_i,y',c',c} \in [0,m)]$$

where $\sigma_{x_i,y',c',c} := x_i + y' + c' - c2^{k'}$ is the digit that will appear at the i-th position of $x+y$ after adding y' and the carry c' to x_i, and carrying c to the $(i+1)$-th position. This way we process the A least significant digits of y. Next, if $B = k \mod k'$ is greater than 0, we need to check all possible remaining B bits of y. For this purpose we iterate over $y'' \in [0, 2^B)$ and all possible carries $c \in \{0,1\}$ to the $(A'+1)$-th digit of $x+y$. Finally, we need to verify that the first $d - A'$ digits of $x+y$ are smaller than m and choose the appropriate value of s to lookup from the dynamic program.

The dynamic program has $\mathcal{O}(d \cdot dm^2)$ states and requires $\mathcal{O}(2^{k'}) = \mathcal{O}(pm)$ computation time per state. As $d = \lfloor \sqrt{\log_p N} \rfloor$ and $m = p^{d-1}$, in total the algorithm uses $\mathcal{O}(d^2 m^3 p) = 2^{\mathcal{O}(\sqrt{\log N})}$ time and space. □

We now show how to use the above algorithm to compute the optimal shift for the given sets Q_r and A.

Lemma 3.9. *For any r, we can deterministically find $\Delta \in [-N+1, N-1]$ such that $|(Q_r+\Delta) \cap A| \ge |A||Q_r|/2N$ in $|A| \cdot 2^{\mathcal{O}(\sqrt{\log N})}$ time.*

Proof. First, we shift all elements of A by $N-1$ and then we need to find $\Delta \in [0, 2(N-1)]$. We will use the method of conditional expectations setting the bits of Δ from the most significant to the least. Let K be the smallest number such that $2^K > 2(N-1)$. We will narrow down the set of possible values Δ maintaining the following invariant:

$$\mathbb{E}[|(Q_r+\Delta) \cap (A-\tau)| \big| \Delta \in [0, 2^k)] \ge \frac{|A||Q_r|}{2^K}$$

where k goes from K to 0 and τ is a variable accumulating all the bits of Δ found so far. In the beginning we have $\tau = 0$ and $\mathbb{E}[|(Q_r+\Delta) \cap A| \big| \Delta \in [0, 2^K)] = \frac{|A||Q_r|}{2^K}$. For each k such that $K \ge k > 0$, we have

$$\begin{aligned}\mathbb{E}[|(Q_r+\Delta) \cap (A-\tau)| \big| \Delta \in [0, 2^k)] = \frac{1}{2}\Big(&\mathbb{E}[|(Q_r+\Delta) \cap (A-\tau)| \big| \Delta \in [0, 2^{k-1})] \\ &+ \mathbb{E}[|(Q_r+\Delta) \cap (A-\tau-2^{k-1})| \big| \Delta \in [0, 2^{k-1})]\Big)\end{aligned}$$

To compute the $(k-1)$-th bit of Δ, we compare the two right-hand terms, choose the bit corresponding to the larger one and accumulate it in τ. Then, the above expected value does not decrease, so the invariant is preserved. After K steps we obtain

$$\mathbb{E}[|(Q_r+\Delta) \cap (A-\tau)| \big| \Delta \in [0, 2^0)] = |(Q_r+\tau) \cap A| \ge \frac{|A||Q_r|}{2^K} \ge |A||Q_r|/2N$$

and τ is the value of Δ with the desired properties. Finally, we need to show how to efficiently

compute the conditional expectation:

$$\begin{aligned}\mathbb{E}[|(Q_r+\Delta)\cap(A-\tau)|\big|\Delta\in[0,2^k)] &= 2^{-k}\sum_{\Delta\in[0,2^k)}|(Q_r+\Delta)\cap(A-\tau)| \\ &= 2^{-k}\sum_{\Delta\in[0,2^k)}\sum_{a\in A}|Q_r\cap\{a-\tau-\Delta\}| \\ &= 2^{-k}\sum_{a\in A}|Q_r\cap[a-\tau-2^k+1,a-\tau]|\end{aligned}$$

which can be calculated with $|A|$ queries about $|Q_r\cap[x,x+2^k)|$ for different values of x. As we compute the conditional expectation $\mathcal{O}(\log N)$ times, the time complexity follows. □

To sum up, we obtain that any subset of $[-N,N]$ can be efficiently partitioned into few (γ,δ)-free sets.

Theorem 3.10. *For any $\gamma,\delta>0$ and set $A=A_i\subseteq[N]$, it is possible to construct $c=2^{\mathcal{O}(\sqrt{\log N})}$ sets $A_{i,1},A_{i,2},\ldots,A_{i,c}$ such that every $A_{i,j}$ is (γ,δ)-free and $\bigcup_j A_{i,j}=A$. The construction is deterministic and runs in $|A|\cdot 2^{\mathcal{O}(\sqrt{\log N})}$ time.*

Proof. We consider every set Q_r as a candidate for the largest set among $Q_0,\ldots,Q_R$. For each candidate Q_r, we iteratively choose the next value of Δ using Lemma 3.9 and then extract from A a (γ,δ)-free subset equal to the intersection of A and $Q_r+\Delta$. Having extracted $c=2^{\mathcal{O}(\sqrt{\log N})}$ subsets, we stop. If A is empty, we obtain a partition of A into at most c (γ,δ)-free sets. See Algorithm 1 for more details.

Let us show that the algorithm does return a partition of A into at most c (γ,δ)-free sets. By Lemma 3.7, there exists r such that $|Q_r|\geq N/2^{\mathcal{O}(\sqrt{\log n})}$. Lemma 3.9 guarantees that for this value of r, the size of A decreases by a factor of $(1-1/2^{\mathcal{O}(\sqrt{\log N})})$ in each iteration. Therefore, after c iterations, the size of A will become $|A|(1-1/2^{\mathcal{O}(\sqrt{\log N})})^c<1$. □

Algorithm 1 Partitioning a set into (γ,δ)-free sets.

```
function Partition(A, N, γ, δ)
    for r = 0, ..., R do
        𝒜 := [ ]
        A' := A
        for i = 1, ..., 2^{O(√log N)} do
            find Δ for A' and Q_r
            B := ∅
            for a ∈ A' do
                if a + Δ ∈ Q_r then
                    B := B ∪ {a}
            𝒜.append(B)
            A' := A' \ B
        if A' = ∅ then
            return 𝒜
```

4 Decreasing the size of the universe

In some of the above reductions we might increase the size of the universe by a constant factor. This is possibly an issue, because some conjectures assume a particular upper bound on its size, say $\{-n^3, \ldots, n^3\}$ or even $\{-n^2, \ldots, n^2\}$. In this section we show how to decrease the size of universe by a constant factor by reducing to a constant number of instances on a smaller universe of the same variant of 3-LDT. We first run the procedure from the following lemma to decrease the universe and only then apply the reduction.

Lemma 4.1. *Let $c > 1$ be any parameter. We can reduce an instance of any variant of 3-LDT$(3; \bar{\alpha}; t)$ over $[-U, U]$ to a constant number of instances of the same variant of 3-LDT over $[-U/c, U/c]$.*

Proof. We assume U to be large enough. Let V be the largest multiple of α_3 smaller than $\frac{U}{2c\sum_i |\alpha_i|}$. We divide numbers from sets A_i into $d = \Theta(c\sum_i |\alpha_i|)$ buckets of size V where the j-th bucket consists of elements from $B_j = [jV, (j+1)V)$ for $j \in [\lfloor -\frac{U}{V} \rfloor, \lfloor \frac{U}{V} \rfloor]$. We will choose the bucket B_{j_i} containing $x_i \in A_i$ and represent x_i as $x_i = x'_i + Vj_i$. For each of the d^3 triples $\bar{j} \in [\lfloor -\frac{U}{V} \rfloor, \lfloor \frac{U}{V} \rfloor]^3$ of the buckets, we create the following instance of 3-LDT over a smaller universe $[-U/c, U/c]$ by setting $A'_i = \{x' : (x' + Vj_i) \in (A_i \cap B_{j_i})\}$ for $i \in \{1, 2\}$ and

$$A'_3 = \left\{ x' + \frac{V}{\alpha_3} \sum_i \alpha_i j_i : (x' + Vj_3) \in (A_3 \cap B_{j_3}) \right\} \cap \left[-\frac{U}{c|\alpha_3|}, \frac{U}{c|\alpha_3|} \right].$$

From the definition, α_3 divides V so A'_3 consists of integers.

We claim that there is a solution of 3-LDT$(3; \bar{\alpha}; t)$ over $[-U, U]$ if and only if at least one of the created instances has a solution over $[-U/c, U/c]$. Assume first that $\bar{x}$ is a solution of 3-LDT$(3; \bar{\alpha}; t)$. Consider the instance corresponding to the triple $\bar{j} = (\lfloor x_1/V \rfloor, \lfloor x_2/V \rfloor, \lfloor x_3/V \rfloor)$. We define the triple $\bar{x}'$ as follows: $x'_1 = x_1 - Vj_1$, $x'_2 = x_2 - Vj_2$ and $x'_3 = (x_3 - Vj_3) + \frac{V}{\alpha_3} \sum_i \alpha_i j_i$. It is easy to see that $\sum_i \alpha_i x'_i = t$. It remains to show that $x'_3 \in [-\frac{U}{c|\alpha_3|}, \frac{U}{c|\alpha_3|}]$. This is indeed the case as

$$|\alpha_3 x'_3| = |t - \alpha_1 x'_1 - \alpha_2 x'_2| \leq t + U/2c \leq U/c.$$

To show the other direction, consider $\bar{x}'$ that is a solution of the instance over $[-U/c, U/c]$ corresponding to a triple $\bar{j}$. We then take $x_1 = x'_1 + Vj_1 \in A_1$, $x_2 = x'_2 + Vj_2 \in A_2$, and $x_3 = x'_3 + Vj_3 - \frac{V}{\alpha_3} \sum_i \alpha_i j_i \in A_3$ which obviously satisfies $\sum_i \alpha_i x_i = t$. □

References

[1] Amir Abboud and Kevin Lewi. Exact weight subgraphs and the k-SUM conjecture. In *40th ICALP*, pages 1–12, 2013. doi:10.1007/978-3-642-39206-1_1.

[2] Amir Abboud and Virginia Vassilevska Williams. Popular conjectures imply strong lower bounds for dynamic problems. In *55th FOCS*, pages 434–443. IEEE Computer Society, 2014. doi:10.1109/FOCS.2014.53.

[3] Amir Abboud, Virginia Vassilevska Williams, and Oren Weimann. Consequences of faster alignment of sequences. In *41st ICALP*, pages 39–51, 2014. doi:10.1007/978-3-662-43948-7_4.

[4] Manuel Abellanas, Ferran Hurtado, Christian Icking, Rolf Klein, Elmar Langetepe, Lihong Ma, Belén Palop, and Vera Sacristán. Smallest color-spanning objects. In *9th ESA*, pages 278–289, 2001. doi:10.1007/3-540-44676-1_23.

[5] Nir Ailon and Bernard Chazelle. Lower bounds for linear degeneracy testing. *J. ACM*, 52(2):157–171, 2005. doi:10.1145/1059513.1059515.

[6] Noga Alon, Raphael Yuster, and Uri Zwick. Color-coding. *J. ACM*, 42(4):844–856, 1995. doi:10.1145/210332.210337.

[7] Amihood Amir, Timothy M. Chan, Moshe Lewenstein, and Noa Lewenstein. On hardness of jumbled indexing. In *41st ICALP*, pages 114–125, 2014. doi:10.1007/978-3-662-43948-7_10.

[8] Daniel W. Archambault, William S. Evans, and David G. Kirkpatrick. Computing the set of all the distant horizons of a terrain. *Int. J. Comput. Geometry Appl.*, 15(6):547–564, 2005. doi:10.1142/S0218195905001841.

[9] Esther M. Arkin, Yi-Jen Chiang, Martin Held, Joseph S. B. Mitchell, Vera Sacristán, Steven Skiena, and Tae-Heng Yang. On minimum-area hulls. *Algorithmica*, 21(1):119–136, 1998. doi:10.1007/PL00009204.

[10] Boris Aronov and Sariel Har-Peled. On approximating the depth and related problems. *SIAM J. Comput.*, 38(3):899–921, 2008. doi:10.1137/060669474.

[11] Ilya Baran, Erik D. Demaine, and Mihai Pătraşcu. Subquadratic algorithms for 3SUM. *Algorithmica*, 50(4):584–596, 2008. doi:10.1007/s00453-007-9036-3.

[12] Luis Barba, Jean Cardinal, John Iacono, Stefan Langerman, Aurélien Ooms, and Noam Solomon. Subquadratic algorithms for algebraic 3SUM. *Discr. & Comput. Geometry*, 61(4):698–734, 2019. doi:10.1007/s00454-018-0040-y.

[13] Gill Barequet and Sariel Har-Peled. Polygon containment and translational min-Hausdorff-distance between segment sets are 3SUM-hard. *Int. J. Comput. Geometry Appl.*, 11(4):465–474, 2001. doi:10.1142/S0218195901000596.

[14] Felix Adalbert Behrend. On sets of integers which contain no three terms in arithmetical progression. In *Proc. Natl. Acad. Sci. (USA)*, volume 32(12), pages 331–332, 1946. doi:10.1073/pnas.32.12.331.

[15] Prosenjit Bose, Marc J. van Kreveld, and Godfried T. Toussaint. Filling polyhedral molds. *Computer-Aided Design*, 30(4):245–254, 1998. doi:10.1016/S0010-4485(97)00075-4.

[16] Timothy M. Chan. More logarithmic-factor speedups for 3SUM, (median, +)-convolution, and some geometric 3SUM-hard problems. In *29th SODA*, pages 881–897, 2018. doi:10.1137/1.9781611975031.57.

[17] Timothy M. Chan and Moshe Lewenstein. Clustered integer 3SUM via additive combinatorics. In *47th STOC*, pages 31–40, 2015. doi:10.1145/2746539.2746568.

[18] Otfried Cheong, Alon Efrat, and Sariel Har-Peled. On finding a guard that sees most and a shop that sells most. In *15th SODA*, pages 1098–1107, 2004.

[19] Mark de Berg, Marko de Groot, and Mark H. Overmars. Perfect binary space partitions. *Comput. Geom.*, 7:81–91, 1997. doi:10.1016/0925-7721(95)00045-3.

[20] Michael Elkin. An improved construction of progression-free sets. In *21th SODA*, pages 886–905, 2010. doi:10.1137/1.9781611973075.72.

[21] Paul Erdös and Paul Turán. On some sequences of integers. *Journal of the London Mathematical Society*, s1-11(4):261–264, 1936. doi:10.1112/jlms/s1-11.4.261.

[22] Jeff Erickson. Lower bounds for linear satisfiability problems. In *6th SODA*, pages 388–395, 1995.

[23] Jeff Erickson. Finding longest arithmetic progressions. Manuscript, 1999.

[24] Jeff Erickson. Lower bounds for linear satisfiability problems. *Chicago J. Theor. Comput. Sci.*, 1999.

[25] Jeff Erickson. New lower bounds for convex hull problems in odd dimensions. *SIAM J. Comput.*, 28(4):1198–1214, 1999. doi:10.1137/S0097539797315410.

[26] Jeff Erickson, Sariel Har-Peled, and David M. Mount. On the least median square problem. *Discr. & Comput. Geometry*, 36(4):593–607, 2006. doi:10.1007/s00454-006-1267-6.

[27] Ari Freund. Improved subquadratic 3SUM. *Algorithmica*, 77(2):440–458, 2017. doi:10.1007/s00453-015-0079-6.

[28] Anka Gajentaan and Mark H. Overmars. On a class of $O(n^2)$ problems in computational geometry. *Comput. Geom.*, 5:165–185, 1995. doi:10.1016/j.comgeo.2011.11.006.

[29] Omer Gold and Micha Sharir. Improved bounds for 3SUM, k-SUM, and linear degeneracy. In *25th ESA*, pages 42:1–42:13, 2017. doi:10.4230/LIPIcs.ESA.2017.42.

[30] Isaac Goldstein, Tsvi Kopelowitz, Moshe Lewenstein, and Ely Porat. How hard is it to find (honest) witnesses? In *24th ESA*, pages 45:1–45:16, 2016. doi:10.4230/LIPIcs.ESA.2016.45.

[31] Alexander Golovnev, Siyao Guo, Thibaut Horel, Sunoo Park, and Vinod Vaikuntanathan. 3SUM with preprocessing: Algorithms, lower bounds and cryptographic applications. *CoRR*, abs/1907.08355, 2019.

[32] Allan Grønlund and Seth Pettie. Threesomes, degenerates, and love triangles. *J. ACM*, 65(4):22:1–22:25, 2018. doi:10.1145/3185378.

[33] Zahra Jafargholi and Emanuele Viola. 3SUM, 3XOR, triangles. *Algorithmica*, 74(1):326–343, 2016. doi:10.1007/s00453-014-9946-9.

[34] Daniel M. Kane, Shachar Lovett, and Shay Moran. Near-optimal linear decision trees for k-SUM and related problems. *J. ACM*, 66(3):16:1–16:18, 2019. doi:10.1145/3285953.

[35] Tsvi Kopelowitz, Seth Pettie, and Ely Porat. Higher lower bounds from the 3SUM conjecture. In *27th SODA*, pages 1272–1287, 2016. doi:10.1137/1.9781611974331.ch89.

[36] Tsvi Kopelowitz and Ely Porat. The strong 3SUM-indexing conjecture is false. *CoRR*, abs/1907.11206, 2019.

[37] Martijn Lafeber. An annotated bibliography on 3SUM-hard problems. https://sarielhp.org/research/CG/collection/04/3sumhard.pdf, 2004.

[38] Mihai Pătraşcu. Towards polynomial lower bounds for dynamic problems. In *42nd STOC*, pages 603–610, 2010. doi:10.1145/1806689.1806772.

[39] Imre Z. Ruzsa. Solving a linear equation in a set of integers I. *Acta Arithmetica LXV.3*, 1993.

[40] Raphaël Salem and Donald C. Spencer. On sets of integers which contain no three terms in arithmetical progression. In *Proc. Natl. Acad. Sci. (USA)*, volume 28(12), pages 561–563, 1942. doi:doi:10.1073/pnas.28.12.561.

[41] Jeanette P. Schmidt and Alan Siegel. The spatial complexity of oblivious k-probe hash functions. *SIAM J. Comput.*, 19(5):775–786, 1990. doi:10.1137/0219054.

[42] Michael A. Soss, Jeff Erickson, and Mark H. Overmars. Preprocessing chains for fast dihedral rotations is hard or even impossible. *Comput. Geom.*, 26(3):235–246, 2003. doi:10.1016/S0925-7721(02)00156-6.

[43] Virginia Vassilevska Williams. Hardness of easy problems: Basing hardness on popular conjectures such as the strong exponential time hypothesis (invited talk). In *10th IPEC*, pages 17–29, 2015. doi:10.4230/LIPIcs.IPEC.2015.17.

[44] Virginia Vassilevska Williams and R. Ryan Williams. Subcubic equivalences between path, matrix, and triangle problems. *J. ACM*, 65(5):27:1–27:38, 2018. doi:10.1145/3186893.

[45] Virginia Vassilevska Williams and Ryan Williams. Finding, minimizing, and counting weighted subgraphs. *SIAM J. Comput.*, 42(3):831–854, 2013. doi:10.1137/09076619X.

A From 1-partite to 3-partite

Theorem 2.3. *For all coefficients $\bar{\alpha}$ and t, we have 3-LDT$(1;\bar{\alpha};t) \leq_2$ 3-LDT$(3;\bar{\alpha};t)$.*

Proof. In this reduction, given one set X we need to create a number of 3-partite instances of 3-LDT in such a way that there exist distinct $x_1, x_2, x_3 \in X$ satisfying the given equation iff at least one of the 3-partite instances is a YES-instance. The reduction will not change coefficients α_i and the parameter t. Note that simply creating a single 3-partite instance by making all three sets equal to X does not work, as we are not able to forbid taking the same element of X more than once.

We use the color-coding technique introduced by Alon et al. [6], in which we choose a number of colorings of the elements of X with k colors in such a way that, for every k-element subset of X, there is a coloring in which all elements from the subset have distinct colors. This can be achieved with high probability by simply choosing sufficiently many random colorings, but we will use the deterministic construction by Schmidt and Siegel [41].

Lemma A.1 (cf. [41])**.** *There exists a family F of $2^{\mathcal{O}(k)} \log^2 n$ functions $[n] \to [k]$ such that, for every k-element set $Y \subseteq [n]$, there exists a function $f \in F$ with $|f[Y]| = k$. Each function is described by a bitstring of length $\mathcal{O}(k) + 2\log\log n$ and, given constant-time read-only random access to the bitstring describing $f \in F$ and any $x \in [n]$, we can compute $f(x)$ in constant time.*

We work with $k = 3$, so the above lemma gives us a family F consisting of $\mathcal{O}(\log^2 n)$ functions. Given a set $X = \{x_1, \ldots, x_n\}$, for every function $f \in F$ and every permutation $\pi \in S_3$ we obtain a 3-partite instance of 3-LDT by setting $A_{\pi(i)} = \{x_c : f(c) = i\}$ for $i \in [3]$. In every 3-partite instance the sets A_i correspond to a partition of the original set X, and for any distinct $x_1, x_2, x_3 \in X$ there exists a 3-partite instance such that $x_1 \in A_1$, $x_2 \in A_2$, and $x_3 \in A_3$. Thus, we showed how to reduce a 1-partite instance of 3-LDT to $\mathcal{O}(\log^2 n)$ instances of 3-partite 3-LDT with the same coefficients $\bar{\alpha}$ and t. The reduction works in $\mathcal{O}(n \log^2 n)$ time. □

B Behrend's construction

Lemma 3.7 (cf. [14, 39]). *For any $\gamma, \delta > 0$ there exist (γ, δ)-free sets $Q_0, Q_1, \ldots, Q_R \subseteq [N]$ such that $R = 2^{\mathcal{O}(\sqrt{\log N})}$ and $|Q_r| \geq N/2^{\mathcal{O}(\sqrt{\log N})}$ for some r.*

Proof. Let p be the smallest power of two larger than $\gamma + \delta$. Let $d = \lfloor \sqrt{\log_p N} \rfloor$ and $m = p^{d-1}$. We consider the set P of all integer points in $\{0, \ldots, m-1\}^d$. Let P_r be the subset of points from P at distance $\sqrt{r}$ from the origin, that is, $P_r = \{p \in P : d^2(o, p) = r\}$, where $0 \leq r \leq d(m-1)^2$. Clearly, the sets P_r are disjoint and $\bigcup_r P_r = P$.

We map the points in P to numbers as follows: $\phi(x_0, \ldots, x_{d-1}) := \sum_i x_i (pm)^i$. The mapping is injective and reversible. We define the set Q_r, $0 \leq r \leq d(m-1)^2$, as $\phi[P_r]$. In other words, a number $x = \sum_i x_i (pm)^i \in Q_r$ iff:

- $\sum_i x_i^2 = r$, and
- for all $0 \leq i < d$ holds $0 \leq x_i < m$.

Note that the numbers in each set Q_r are bounded by $(pm)^d = p^{d^2} \leq N$.

We claim that the sets Q_r are (γ, δ)-free. Suppose the contrary, that there are three distinct elements $a, b, c \in Q_r$ such that $\gamma a + \delta b = (\gamma + \delta) c$. Consider their (pm)-base representations $\bar{a}, \bar{b}, \bar{c}$. Observe that as all coordinates are smaller than m, the equation $\gamma a + \delta b = (\gamma + \delta) c$ holds for every digit in their (pm)-base representations, so for every $0 \leq i < d$ we have $\gamma a_i + \delta b_i = (\gamma + \delta) c_i$. We combine this with the triangle inequality

$$(\gamma + \delta)\sqrt{r} = \|(\gamma + \delta)\bar{c}\|_2 = \|\gamma \bar{a} + \delta \bar{b}\|_2 \leq \|\gamma \bar{a}\|_2 + \|\delta \bar{b}\|_2 = (\gamma + \delta)\sqrt{r}$$

which becomes an equality if and only if $\bar{a}$ and $\bar{b}$ are collinear. As their norms are equal ($a, b \in Q_r$), we finally obtain $\bar{a} = \bar{b} = \bar{c}$ which contradicts the distinctness of a, b and c.

We now show that at least one of the sets Q_r is larger than $N/2^{\mathcal{O}(\sqrt{\log N})}$. The total size of Q_r is equal to the size of P, that is m^d. There are $d(m-1)^2 + 1$ sets Q_r, so at least one of them is larger than

$$\frac{m^d}{d(m-1)^2 + 1} \geq \frac{m^{d-2}}{d} = \frac{p^{(d-1)(d-2)+3d-2}}{dp^{3d-2}} = \frac{N}{2^{\mathcal{O}(\sqrt{\log N})}} \qquad \square$$