Geometric Pattern Matching Reduces to k-SUM

Boris Aronov
Department of Computer Science and Engineering, Tandon School of Engineering,
New York University, Brooklyn, NY ,USA
https://engineering.nyu.edu/faculty/boris-aronov
boris.aronov@nyu.edu

Jean Cardinal
Université libre de Bruxelles (ULB), Belgium
http://homepages.ulb.ac.be/~jcardin/
jcardin@ulb.ac.be

Abstract

We prove that some exact geometric pattern matching problems reduce in linear time to k-SUM when the pattern has a fixed size k. This holds in the real RAM model for searching for a similar copy of a set of $k \geq 3$ points within a set of n points in the plane, and for searching for an affine image of a set of $k \geq d+2$ points within a set of n points in d-space.

As corollaries, we obtain improved real RAM algorithms and decision trees for the two problems. In particular, they can be solved by algebraic decision trees of near-linear height.

2012 ACM Subject Classification Theory of computation → Design and analysis of algorithms; Theory of computation → Computational geometry

Keywords and phrases Geometric pattern matching, k-SUM problem, Linear decision trees

Digital Object Identifier 10.4230/LIPIcs.ISAAC.2020.32

Funding *Boris Aronov*: Partially supported by NSF grant CCF-15-40656 and by grant 2014/170 from the US-Israel Binational Science Foundation. Work by B.A. on this paper has been partially carried out while visiting ULB in November-December 2019, with support from ULB and F.R.S.-FNRS (Fonds National de la Recherche Scientifique).
Jean Cardinal: Supported by the F.R.S.-FNRS (Fonds National de la Recherche Scientifique) under CDR Grant J.0146.18.

1 Introduction

The k-SUM problem is a fixed-parameter version of the NP-complete SUBSET SUM problem. It consists of deciding, given a set of n numbers, whether any subset of size k sum to zero. The problem for $k = 3$, known as 3-SUM, is now a well-established bottleneck problem in fine-grained complexity theory (see for instance [1, 28] and references therein). While there are many reductions showing 3-SUM- or k-SUM-hardness of computational problems in geometry, only few reductions *to* 3-SUM and k-SUM are known. We give examples of computational geometry problems that reduce to 3-SUM or k-SUM.

Our results are motivated by the nontrivial improved upper bounds on the complexity of 3-SUM and k-SUM proven in the recent years. While it has long been conjectured that no subquadratic algorithm for 3-SUM existed, it is now known to be solvable in time $O((n^2/\log n)(\log\log n)^2)$ in the real RAM model, and in time $O((n^2/\log^2 n)(\log\log n)^{O(1)})$ if we allow bitwise operations on fixed-length words [25, 20, 22, 14]. The existence of an $O(n^{2-\delta})$ algorithm for some $\delta > 0$ remains an open problem. Using folklore *meet-in-the-middle* algorithms, k-SUM can be solved in time $O(n^{\lceil k/2 \rceil})$ if k is odd, and in time $O(n^{k/2}\log n)$ if k is even. Recently, Kane, Lovett, and Moran [26] showed that it can be solved in time $O(n\log^2 n)$ in the linear decision tree model, improving on previous polynomial bounds [13, 19].

Table 1 Known upper bounds on the time complexity of exact geometric pattern matching in various settings (taken from [11] and [23], Chapter 54). We indicate the dependency on the pattern size k.

Transformations	Dimension	Complexity
congruence	2	$O(kn^{4/3}\log n)$ [11]
congruence	3	$O(kn^{5/3}\log n 2^{O(\alpha(n)^2)})$ [5]
translation	d	$O(kn\log n)$ (easy)
homothety	d	$O(kn^{1+1/d}\log n)$ [18, 11]
similarity	d	$O(kn^{d}\log n)$ [11]
affine	d	$O(kn^{d+1}\log n)$ [11]

Geometric pattern matching

We consider two problems involving searching for a given set P of k points, called the *pattern*, within a larger set S of points, up to some geometric transformation. Here we focus on *exact* algorithms, in which the pattern must match the subset of points exactly. We consider the following two problems.

▶ **Problem 1** (SIMILARITY MATCHING). *For a fixed integer $k \geq 3$, given a set P of k points in the plane and a set S of n points in the plane, determine whether S contains the image of P under a similarity transformation.*

▶ **Problem 2** (AFFINE MATCHING). *For fixed integers $d \geq 2$ and $k \geq d+2$, given a set P of k points in $\mathbb{R}^d$ containing $d+1$ affinely independent points, and a set S of n points in $\mathbb{R}^d$, determine whether S contains the image of P under an affine transformation.*

A large body of the computational geometry and pattern recognition literature is dedicated to the problems of finding *approximate* matches up to some geometric transformation, where the quality of the approximation is typically measured by the Hausdorff distance [15, 24, 21, 6]. For exact pattern matching problems under different families of transformations, known upper bounds on time complexity have been compiled in a survey by Peter Braß [11]. We reproduce them in Table 1.

The complexity of these algorithms are directly related to bounds on the maximum number of occurrences of a pattern or a distance in a set of n points. In fact, such bounds directly yield a lower bound on the computational problem of *listing* all occurrences of the pattern. A prototypal example is Erdős' unit distance problem; see Braß and Pach [12] for more examples. It is known, in particular, that there can be $\Theta(n^2)$ similar copies of a pattern in an n-point set [18, 3, 4]. Structural results on the extremal point sets are also known [2]. For affine transformations in $\mathbb{R}^d$, there exist pairs P, S such that S contains $\Theta(n^{d+1})$ copies of P: for instance the d-dimensional lattice $\{1, 2, \ldots, n^{1/d}\}^d$ contains $\Theta(n^{d+1})$ affine images of a cube.

Our results

We suppose we can perform exact computations over the reals. Therefore, all the algorithms that we consider are either uniform algorithms in the real RAM model, or nonuniform algorithms in the algebraic decision tree model.

Our main result is the following.

▶ **Theorem 1.** SIMILARITY MATCHING *and* AFFINE MATCHING *reduce in randomized linear time to* k-SUM.

We refer the reader to the exact definitions of the k-SUM problem and the notion of randomized linear-time reduction given later. Theorem 1 has a number of consequences. Let us consider the special case of the SIMILARITY MATCHING problem in which $k = 3$.

▶ **Problem 3** (TRIANGLE). *Given a triangle Δ and a set S of n points in the plane, determine whether S contains three points whose convex hull is similar to Δ.*

Combining the reduction provided by Theorem 1 with the real RAM algorithm for 3-SUM from Chan [14], we obtain the following.

▶ **Corollary 2.** *There exists an $O((n^2/\log n)(\log\log n)^2)$ randomized real RAM algorithm for* TRIANGLE*. In particular, there exists a subquadratic algorithm to detect equilateral triangles in a point set.*

This contrasts with our current knowledge on the related 3-SUM-hard problem of finding three collinear points, also known as GENERAL POSITION TESTING. Despite recent attempts [10, 14], it is still an open problem to find a subquadratic algorithm for GENERAL POSITION TESTING.

Our next corollary is obtained directly from known algorithms for k-SUM. It improves on the best known $O(n^{d+1}\log n)$ algorithm whenever $k < 2(d+1)$.

▶ **Corollary 3.** *There exists an $O(n^{\lceil k/2\rceil})$ (for k odd), or an $O(n^{k/2}\log n)$ (for k even) randomized real RAM algorithm for* AFFINE MATCHING.

Finally, we consider the nonuniform *decision tree* complexity, also known as *query complexity*, of the two problems. By applying a recent result of Kane, Lovett, and Moran [26], we can bound the number of algebraic tests that are required to detect copies of P in an input set S.

▶ **Corollary 4.** *There exist randomized algebraic decision trees of height $O(n\log^2 n)$ for* SIMILARITY MATCHING *and* AFFINE MATCHING.

In fact, if the pattern P is a fixed parameter, that is, when P is not part of the input, but known at the algorithm design time, then the decision tree in the statement above only involves *linear* tests.

▶ **Corollary 5.** *There exist randomized linear decision trees of height $O(n\log^2 n)$ for the fixed-parameter versions of* SIMILARITY MATCHING *and* AFFINE MATCHING*, in which P is a fixed parameter of the problems.*

In a recent paper, Aronov, Ezra, and Sharir [8] study the following problem: Given three sets A, B, C of n points in the plane, decide whether there exists $(a, b, c) \in A \times B \times C$ that simultaneously satisfies two real polynomial equations. They provide a subquadratic upper bound on the algebraic decision tree complexity of this problem. In a preliminary version of their paper [9] (version 2, Corollary 4.4), they considered the TRIANGLE problem as a special case of this problem. This version also contains a proof that TRIANGLE is 3-SUM-hard. As our result shows, it turns out that this special case is in fact much easier than the general problem, as the two polynomial equations can be made linear. Hence TRIANGLE is actually linear-time equivalent to 3-SUM, and its decision tree complexity is near-linear. We refer to [8, 9] for a thorough discussion of the relation between these and other related problems.

Plan

In the next section, we define a number of variants of the k-SUM problem and prove they are all equivalent in the computation model we consider. In Section 3, we prove our main result for SIMILARITY MATCHING. Section 4 considers the AFFINE MATCHING problem. The last section is dedicated to the proof of Corollaries 4 and 5.

2 Linear degeneracy testing

We first give a definition of the k-SUM problem. Here, $k \geq 3$ is a fixed integer, and X is a ring.

▶ **Problem 4** (k-SUM(X)). *Given k sets $A_1, \ldots, A_k$ of n elements of X, determine whether there exists a k-tuple $(a_1, \ldots, a_k) \in \bigtimes_{i=1}^{k} A_i$ such that $\sum_{i=1}^{k} a_i = 0$.*

Our next problem is often referred to as *linear degeneracy testing* [7, 17]. We consider the cases where $X = \mathbb{R}$ or $\mathbb{C}$ with the usual addition and multiplication operations, or where $X = \mathbb{R}^d$ or $\mathbb{C}^d$ for some integer $d \geq 2$, with the vector addition and Hadamard (entrywise) product defined by $(uv)_i = u_i v_i$. In the latter cases, the all-zero vector is denoted by 0, and the all-one vector by 1.

▶ **Problem 5** (k-LDT(X)). *For a linear function $f\colon X^k \to X$ given by $f(a_1, \ldots, a_k) = \beta_0 + \sum_{i=1}^{k} \beta_i a_i$ with $\beta_i \in X$ for $0 \leq i \leq k$, given k sets $A_1, \ldots, A_k$ of n elements of X, determine whether there exists a k-tuple $(a_1, \ldots, a_k) \in \bigtimes_{i=1}^{k} A_i$ such that $f(a_1, \ldots, a_k) = 0$.*

We make two observations. First, these are *fixed-parameter problems*: the integer k is part of the definition of the problem, not of the input. The same can be assumed for the function f. Such parameters will be referred to as *fixed* in what follows. Another observation is that using the Hadamard product in the definition of the function f allows us to combine conditions on the sought k-tuples: In the ring X, searching for k-tuples that simultaneously satisfy d linear equations can be cast as k-LDT(X^d).

It is clear that k-SUM is the special case of k-LDT in which $\beta_0 = 0$ and $\beta_i = 1$ for $1 \leq i \leq k$. On the other hand, k-LDT is not harder than k-SUM.

▶ **Lemma 6.** *For any integer $d > 0$, k-LDT(X) reduces in linear time to k-SUM(X).*

Proof. Consider the sets A_i from the k-LDT instance, and let $A'_i := \{\beta_i a \mid a \in A_i\}$ for all $1 \leq i < k$, and $A'_k := \{\beta_k a + \beta_0 \mid a \in A_k\}$. Then the instance of k-SUM composed of the sets A'_i has a solution if and only if the instance of k-LDT has a solution. ◀

In what follows, we say that a problem A *reduces to problem B in randomized $g(n)$ time* if there exists an algorithm in the real RAM model with access to random real numbers in $[0, 1]$ that maps any instance of size n of A to an equivalent instance of B in time $O(g(n))$ with probability 1.

Over the reals, the vector and scalar versions of k-SUM are also essentially equivalent, up to such a randomized reduction.

▶ **Lemma 7.** *For any fixed integer $d > 0$, k-SUM($\mathbb{R}^d$) reduces in randomized linear time to k-SUM($\mathbb{R}$) .*

Proof. Given an instance $\{A_1, \ldots, A_k\}$ of k-SUM($\mathbb{R}^d$), pick a uniform random unit vector $v \in \mathbb{R}^d$ (see for instance Chapter V in Devroye's classical textbook [16] for the generation of random vectors on the unit hypersphere) and consider the sets $A'_i := \{a \cdot v \mid a \in A_i\} \subset \mathbb{R}$,

where $a \cdot v$ is the usual dot product. They form an instance of k-SUM($\mathbb{R}$) such that any solution to the original instance of k-SUM($\mathbb{R}^d$) is also a solution. In the other direction, suppose there is a k-tuple $(a'_1, \ldots, a'_k) \in \bigtimes_{i=1}^k A'_i$ such that $\sum_{i=1}^k a'_i = 0$, where $a'_i = a_i \cdot v$. Hence we have $\sum_{i=1}^k a_i \cdot v = 0$, which is either because $v \perp \sum_{i=1}^k a_i$ and $\sum_{i=1}^k a_i \neq 0$, or because $\sum_{i=1}^k a_i = 0$. Since $v \perp \sum_{i=1}^k a_i$ and $\sum_{i=1}^k a_i \neq 0$ occurs with probability 0, the k-tuple $(a_1, \ldots, a_k)$ is a solution of the instance $\{A_1, \ldots, A_k\}$ of k-SUM($\mathbb{R}^d$) with probability 1. ◀

We also make the following simple observation:

▶ **Observation 8.** *k-SUM($\mathbb{C}^d$) is equivalent to k-SUM($\mathbb{R}^{2d}$).*

3 Searching for a similar copy

Recall that in the TRIANGLE problem, we want to determine whether an input set S of n points in the plane contains three points whose convex hull is similar to a given triangle Δ. The short proof of the following result uses the interpretation of points in the plane as complex numbers, an idea that was exploited in a combinatorial context before [18, 27].

▶ **Lemma 9.** ANGLE *reduces in linear time to* 3-SUM($\mathbb{C}$).

Proof. Let $u = re^{i\theta}$ be such that the three numbers $0, 1, u$ are the vertices of a triangle similar to Δ in the complex plane. Recall that multiplying by $re^{i\theta}$ has a geometric interpretation in the complex plane as scaling by a factor r and rotating by an angle θ. Hence three other complex numbers $a, b, c \in \mathbb{C}$ form a triangle similar to Δ in the complex plane with the same orientation if and only if $c - a = u(b - a)$, or equivalently if $(u - 1)a - ub + c = 0$. Hence TRIANGLE reduces to 3-LDT($\mathbb{C}$) with $\beta = (0, u - 1, -u, 1)$. From Lemma 6, it reduces in linear time to 3-SUM($\mathbb{C}$). ◀

Combining with Observation 8 and Lemma 7, we obtain:

▶ **Theorem 10.** TRIANGLE *reduces in randomized linear time to* 3-SUM($\mathbb{R}$).

Recall that TRIANGLE is also known to be 3-SUM-hard [9], hence it is actually linear-time equivalent to 3-SUM. Our result generalizes naturally to larger patterns.

▶ **Lemma 11.** SIMILARITY MATCHING *reduces in linear time to k-SUM($\mathbb{C}^{k-2}$).*

Proof. Let $u_1, \ldots, u_{k-2} \in \mathbb{C}$ be such that the set $Q = \{0, 1, u_1, \ldots, u_{k-2}\}$ is similar to P in the complex plane. Then k numbers $a_1, \ldots, a_k \in \mathbb{C}$ form a similar copy of Q in the complex plane, with a_1 mapped to 0, a_2 to 1, and so on, if and only if $a_i - a_1 = u_{i-2}(a_2 - a_1)$ for all $3 \leq i \leq k$. These are $k - 2$ linear equations on the k complex numbers $a_1, \ldots, a_k$, hence SIMILARITY MATCHING reduces in linear time to k-LDT($\mathbb{C}^{k-2}$). From Lemma 6, it reduces in linear time to k-SUM($\mathbb{C}^{k-2}$). ◀

Again, combining with Observation 8 and Lemma 7, we obtain the first statement of Theorem 1.

▶ **Theorem 12.** SIMILARITY MATCHING *reduces in randomized linear time to k-SUM($\mathbb{R}$).*

4 Searching for an affine image

We now prove the analogous result for the affine case. As a warm-up, we first consider the following simpler special case of AFFINE MATCHING in which the pattern is a square. Four points form the affine image of vertices of a square if and only if they are the vertices of a (possibly degenerate) parallelogram. Hence the problem can be cast as follows.

▶ **Problem 6** (PARALLELOGRAM). *Given a set S of n points in the plane, determine whether S contains four points whose convex hull is a parallelogram.*

▶ **Theorem 13.** PARALLELOGRAM *reduces in randomized linear time to* 4-SUM($\mathbb{R}$).

Proof. Four points $a_1, a_2, a_3, a_4 \in S$ in this order form a parallelogram with a_1a_2 parallel to a_4a_3 and a_2a_3 parallel to a_1a_4 if and only if $a_2 - a_1 = a_3 - a_4$, or equivalently if $a_1 - a_2 + a_3 - a_4 = 0$. Hence PARALLELOGRAM reduces to 4-LDT($\mathbb{R}^2$) with $\beta = ((0,0),(1,1),(-1,-1),(1,1),(-1,-1))$. From Lemmas 6 and 7, it also reduces in randomized linear time to 4-SUM($\mathbb{R}$). ◀

The general case follows from the following observation. Consider a matrix $Q \in \mathbb{R}^{n\times n}$, and let Q_k denote the matrix obtained from Q by replacing its kth column by the column vector x^T, where $x_1, x_2, \ldots, x_n$ are variables. Then $\det Q_k$ is a linear combination of $x_1, x_2, \ldots, x_n$, with coefficients defined by Q.

▶ **Lemma 14.** AFFINE MATCHING *reduces in linear time to* k-SUM($\mathbb{R}^\ell$) *with* $\ell = d(k - (d+1))$.

Proof. We use the notation $[k] := \{1, 2, \ldots, k\}$. Let $p_i = (p_{i,1}, \ldots, p_{i,d})$ be a row vector representing the ith point of P. From the problem definition, P must contain $d+1$ affinely independent points. Since we suppose k and d fixed, these points can be determined in constant time. We therefore assume without loss of generality that they are the first $d+1$ points $p_1, \ldots, p_{d+1}$. Let $A = \{a_1, \ldots, a_k\} \in \binom{S}{k}$ be a candidate match. In order for the set A to be the image of P under an affine transformation, there must be a solution to the system of k linear equations of the form $p_iF + t = a_i$ for all $i \in [k]$, with $d^2 + d$ real unknowns $F \in \mathbb{R}^{d\times d}$ and $t \in \mathbb{R}^d$. The system can be decomposed into d systems, one for each coordinate $j \in [d]$. Each consists of k equations with $d+1$ unknowns, of the form $p_iF_j + t_j = a_{ij}$ for $i \in [k]$, where F_j is the jth column of F. We consider one such system, for a fixed $j \in [d]$, and restrict it to the first $d+1$ equations only:

$$Q \cdot \begin{pmatrix} F_j \\ t_j \end{pmatrix} = \begin{pmatrix} a_{1,j} \\ \vdots \\ a_{d+1,j} \end{pmatrix}, \text{ where } Q = \begin{pmatrix} p_1 & 1 \\ \vdots & \vdots \\ p_{d+1} & 1 \end{pmatrix}.$$

Since the first $d+1$ points of P are affinely independent, Q is invertible and the system defines a unique solution for the coefficients F_j and t_j of the affine transformation. From Cramer's rule, the value of the kth unknown is the ratio $\det Q_k / \det Q$, where Q_k is the matrix obtained by replacing the kth column of Q by $(a_{1,j}, \ldots, a_{d+1,j})^T$. From the above observation and the fact that Q does not depend on S, the expressions $\det Q_k / \det Q$ are linear combinations of the values $a_{1,j}, \ldots, a_{d+1,j}$, with coefficients determined by P. Hence the explicit solution for the coefficients F_j and t_j are linear combinations of the $a_{1,j}, \ldots, a_{d+1,j}$.

A necessary and sufficient condition for the set A to be a match is that the remaining $k - d - 1$ points of A are also images of the corresponding points in P. Hence we require that for all $i > d+1$ the ith equation $p_iF_j + t_j = a_{ij}$ is also satisfied by this solution. The

unknowns F_j and t_j can be replaced by linear combinations of $a_{1,j}, \ldots, a_{d+1,j}$. Hence we obtain a set of $k-(d+1)$ linear equations on the variables $a_{1,j}, \ldots, a_{k,j}$, with coefficients depending on P.

Since these $k-(d+1)$ equations must hold for all coordinates $j \in [d]$ simultaneously, we obtain that AFFINE MATCHING reduces to k-LDT($\mathbb{R}^\ell$) with $\ell = d(k-(d+1))$. From Lemma 6 it also reduces to k-SUM($\mathbb{R}^\ell$). Since d and k are fixed, the reduction takes linear time. ◀

Combining with the randomization step in Lemma 7, we get the second part of Theorem 1.

▶ **Theorem 15.** AFFINE MATCHING *reduces in randomized linear time to* k-SUM($\mathbb{R}$).

5 Algebraic decision tree complexity

An *algebraic decision tree* is a type of nonuniform algorithm for problems on inputs composed of n real numbers. For each input size n, it consists of a binary tree whose internal nodes are labeled with inequalities of the form "$q(x) \leq 0$" on the input $x \in \mathbb{R}^n$, where q is a bounded-degree n-variate polynomial in $x_1, x_2, \ldots, x_n$. Inequalities are interpreted as *queries* on the input, and the two subtrees correspond to the possible outcomes of the query on the input. Leaves of the tree are labeled with the answer to the problem. The minimum height $h(n)$ of an algebraic decision tree solving instances of size n the problem is the *decision tree* complexity, or *query* complexity of the problem. When the queries only involve linear functions, such trees are called *linear decision trees*. In that case, a query is said to be t-*sparse* when it involves at most t numbers of the input.

We have the following recent result on the linear decision tree complexity of the k-SUM problem.

▶ **Theorem 16** (Kane, Lovett, Moran [26])**.** *The* k-SUM *problem on* n *elements can be solved by a linear decision tree of height* $O(n \log^2 n)$ *in which all the queries are* $2k$*-sparse and have only* $\{-1, 0, 1\}$ *coefficients.*

We now show that this result directly applies to the SIMILARITY MATCHING and AFFINE MATCHING problems, thereby proving Corollary 4.

We first consider the SIMILARITY MATCHING problem, an instance y of which consists of two coordinates per point of P and S, hence of $2(k+n)$ real numbers. Suppose we apply the randomized reduction proposed in Theorem 12 to obtain an instance of k-SUM($\mathbb{R}$). Now consider the linear decision tree from Theorem 16. Each linear query on the transformed input maps to a query on the original input numbers y. Because the reduction only involves multiplications and additions on these numbers, such queries are algebraic queries on the original input y. Therefore, the linear decision tree for k-SUM maps to an algebraic decision tree of the same height for SIMILARITY MATCHING. The same reasoning applies to AFFINE MATCHING. In that case, it suffices to observe that multiplying both sides of every query by the quantity $\det Q$ for the matrix Q used in the proof of Lemma 14 yields algebraic queries again. Note that since k and d are constant and the linear queries in Theorem 16 are sparse, the queries have bounded degree and bounded size. This proves Corollary 4.

Also note that if we suppose the pattern P is a fixed parameter of the problem, then the two problems are solved by *linear* decision trees of height $O(n \log^2 n)$. It can indeed be checked that the algebraic queries do not involve multiplications between coordinates of the points of S, hence are linear whenever P is fixed. This proves Corollary 5. It applies in particular to the PARALLELOGRAM problem, or for finding an equilateral triangle in a point set.

References

1 Amir Abboud, Virginia Vassilevska Williams, and Huacheng Yu. Matching triangles and basing hardness on an extremely popular conjecture. *SIAM J. Comput.*, 47(3):1098–1122, 2018. doi:10.1137/15M1050987.

2 Bernardo M. Ábrego, György Elekes, and Silvia Fernández-Merchant. Structural results for planar sets with many similar subsets. *Combinatorica*, 24(4):541–554, 2004. doi:10.1007/s00493-004-0033-8.

3 Bernardo M. Ábrego and Silvia Fernández-Merchant. On the maximum number of equilateral triangles, I. *Discrete & Computational Geometry*, 23(1):129–135, 2000. doi:10.1007/PL00009486.

4 Bernardo M. Ábrego, Silvia Fernández-Merchant, Daniel J. Katz, and Levon Kolesnikov. On the number of similar instances of a pattern in a finite set. *Electr. J. Comb.*, 23(4):P4.39, 2016. URL: http://www.combinatorics.org/ojs/index.php/eljc/article/view/v23i4p39.

5 Pankaj Agarwal and Micha Sharir. The number of congruent simplices in a point set. *Discrete & Computational Geometry*, 28(2):123–150, 2002. doi:10.1007/s00454-002-0727-x.

6 Dror Aiger and Klara Kedem. Geometric pattern matching for point sets in the plane under similarity transformations. *Inf. Process. Lett.*, 109(16):935–940, 2009. doi:10.1016/j.ipl.2009.04.021.

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

8 Boris Aronov, Esther Ezra, and Micha Sharir. Testing polynomials for vanishing on cartesian products of planar point sets. In Sergio Cabello and Danny Z. Chen, editors, *36th International Symposium on Computational Geometry, SoCG 2020, June 23-26, 2020, Zürich, Switzerland*, volume 164 of *LIPIcs*, pages 8:1–8:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.SoCG.2020.8.

9 Boris Aronov, Esther Ezra, and Micha Sharir. Testing polynomials for vanishing on cartesian products of planar point sets, 2020. arXiv:2003.09533.

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

11 Peter Braß. Combinatorial geometry problems in pattern recognition. *Discrete & Computational Geometry*, 28(4):495–510, 2002. doi:10.1007/s00454-002-2884-3.

12 Peter Braß and János Pach. Problems and results on geometric patterns. In David Avis, Alain Hertz, and Odile Marcotte, editors, *Graph Theory and Combinatorial Optimization*, pages 17–36. Springer US, Boston, MA, 2005. doi:10.1007/0-387-25592-3_2.

13 Jean Cardinal, John Iacono, and Aurélien Ooms. Solving k-SUM using few linear queries. In Piotr Sankowski and Christos D. Zaroliagis, editors, *24th Annual European Symposium on Algorithms, ESA 2016, August 22-24, 2016, Aarhus, Denmark*, volume 57 of *LIPIcs*, pages 25:1–25:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016. doi:10.4230/LIPIcs.ESA.2016.25.

14 Timothy M. Chan. More logarithmic-factor speedups for 3SUM, (median, +)-convolution, and some geometric 3SUM-hard problems. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018*, pages 881–897, 2018. doi:10.1137/1.9781611975031.57.

15 L. Paul Chew, Michael T. Goodrich, Daniel P. Huttenlocher, Klara Kedem, Jon M. Kleinberg, and Dina Kravets. Geometric pattern matching under euclidean motion. *Comput. Geom.*, 7:113–124, 1997. doi:10.1016/0925-7721(95)00047-X.

16 Luc Devroye. *Non-Uniform Random Variate Generation*. Springer, 1986. URL: http://www.nrbook.com/devroye/.

17 Bartłomiej Dudek, Paweł Gawrychowski, and Tatiana Starikovskaya. All non-trivial variants of 3-LDT are equivalent. In *Proceedings of the 52nd Annual ACM Symposium on Theory of Computing STOC, June 22–26, 2020, Chicago (IL)*, 2020.

18 György Elekes and Paul Erdős. *Similar Configurations and Pseudogrids*, pages 85–104. Colloquia Mathematica Societatis János Bolyai, North Holland, Amsterdam, 1994.

19 Esther Ezra and Micha Sharir. A nearly quadratic bound for point-location in hyperplane arrangements, in the linear decision tree model. *Discrete & Computational Geometry*, 61(4):735–755, 2019. doi:10.1007/s00454-018-0043-8.

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

21 Martin Gavrilov, Piotr Indyk, Rajeev Motwani, and Suresh Venkatasubramanian. Combinatorial and experimental methods for approximate point pattern matching. *Algorithmica*, 38(1):59–90, 2004. doi:10.1007/s00453-003-1043-4.

22 Omer Gold and Micha Sharir. Improved bounds for 3SUM, k-SUM, and linear degeneracy. In *25th Annual European Symposium on Algorithms, ESA 2017, September 4-6, 2017, Vienna, Austria*, pages 42:1–42:13, 2017. doi:10.4230/LIPIcs.ESA.2017.42.

23 Jacob E. Goodman, Joseph O'Rourke, and Csaba D. Tóth, editors. *Handbook of Discrete and Computational Geometry, Third Edition*. CRC Press LLC, 2017.

24 Michael T. Goodrich, Joseph S. B. Mitchell, and Mark W. Orletsky. Approximate geometric pattern matching under rigid motions. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(4):371–379, 1999. doi:10.1109/34.761267.

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

26 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.

27 Miklós Laczkovich and Imre Z. Ruzsa. The number of homothetic subsets. In Ronald L. Graham and Jaroslav Nešetřil, editors, *The Mathematics of Paul Erdős II*, pages 294–302. Springer Berlin Heidelberg, 1997. doi:10.1007/978-3-642-60406-5_26.

28 Virginia Vassilevska Williams. On some fine-grained questions in algorithms and complexity. In *Proceedings of the ICM*, 2018.