

Conditional Lower Bounds for Dynamic Geometric Measure Problems

Justin Dallant  

Université libre de Bruxelles, Belgium

John Iacono  

Université libre de Bruxelles, Belgium

Abstract

We give new polynomial lower bounds for a number of dynamic measure problems in computational geometry. These lower bounds hold in the Word-RAM model, conditioned on the hardness of either 3SUM, APSP, or the Online Matrix-Vector Multiplication problem [Henzinger et al., STOC 2015]. In particular we get lower bounds in the incremental and fully-dynamic settings for counting maximal or extremal points in \mathbb{R}^3 , different variants of Klee’s Measure Problem, problems related to finding the largest empty disk in a set of points, and querying the size of the i ’th convex layer in a planar set of points. We also answer a question of Chan et al. [SODA 2022] by giving a conditional lower bound for dynamic approximate square set cover. While many conditional lower bounds for dynamic data structures have been proven since the seminal work of Pătraşcu [STOC 2010], few of them relate to computational geometry problems. This is the first paper focusing on this topic. Most problems we consider can be solved in $O(n \log n)$ time in the static case and their dynamic versions have only been approached from the perspective of improving known upper bounds. One exception to this is Klee’s measure problem in \mathbb{R}^2 , for which Chan [CGTA 2010] gave an unconditional $\Omega(\sqrt{n})$ lower bound on the worst-case update time. By a similar approach, we show that such a lower bound also holds for an important special case of Klee’s measure problem in \mathbb{R}^3 known as the Hypervolume Indicator problem, even for amortized runtime in the incremental setting.

2012 ACM Subject Classification Theory of computation \rightarrow Computational geometry; Theory of computation \rightarrow Problems, reductions and completeness

Keywords and phrases Computational geometry, Fine-grained complexity, Dynamic data structures

Digital Object Identifier 10.4230/LIPIcs.ESA.2022.39

Related Version *Full Version*: <https://arxiv.org/abs/2112.10095> [28]

Funding *Justin Dallant*: Supported by the French Community of Belgium via the funding of a FRIA grant.

John Iacono: Supported by the Fonds de la Recherche Scientifique-FNRS under Grant no MISU F 6001 1.

Acknowledgements We thank Jean Cardinal for helpful discussions about the topic of this paper.

1 Introduction

In 1995, Gajentaan and Overmars [30] introduced the notion of 3SUM hardness, showing that a number of problems in computational geometry can not be solved in subquadratic time, assuming the so-called 3SUM problem can not be solved in subquadratic time.¹ The general approach of proving polynomial lower bounds based on a few conjectures about key problems has since grown into its own subfield of complexity theory known as *fine-grained*

¹ In 2014, Grønlund and Pettie [31] showed that the 3SUM problem can be solved in (slightly) subquadratic time. The modern formulation thus replaces “subquadratic” with “truly subquadratic”, i.e. $O(n^{2-\varepsilon})$ for some constant $\varepsilon > 0$.



complexity. The most popular of these conjectures concern the aforementioned 3SUM problem, All-Pairs-Shortest-Paths (APSP), Boolean Matrix Multiplication (BMM), Triangle finding in a graph, Boolean Satisfiability (SAT) and the Orthogonal Vectors problem (2OV) (see for example the introductory surveys by Bringmann [13] and V. V. Williams [55]). Another problem which crops up as a bottleneck in computational geometry is Hopcroft’s problem (see the recent paper by Chan and Zheng [24]).

Pătrașcu [50] launched the study of such polynomial lower bounds for dynamic problems, where instead of simply computing a function on a single input, we want to be able to update that input and get the corresponding output of the function without having to recompute it from scratch. In particular, he introduced the Multiphase problem and showed a polynomial lower bound on its complexity, conditioned on the hardness of the 3SUM problem. Using the Multiphase problem as a stepping stone, he showed conditional hardness results for a variety of dynamic problems. Improvements and other conditional lower bounds for dynamic problems (data structure problems) have since appeared in the literature [1–3, 5–7, 9, 11, 12, 25, 27, 34, 35, 38, 40, 41, 49, 53]. Of particular interest for the purpose of this work is a paper by Kopelowitz et al. [41] where the approach of Pătrașcu is improved by showing a tighter reduction from 3SUM to the so-called Set Disjointness problem (an intermediate problem between 3SUM and the Multiphase problem), as well as a paper by V. V. Williams and Xu [56], which obtains a similar reduction from the so-called Exact Triangle problem. Also particularly relevant here is the work of Henzinger et al. [34], who show that many of the known bounds on dynamic problems can be derived (and even strengthened) by basing proofs on a hardness conjecture about the Online Boolean Matrix-Vector Multiplication (OMv) problem which they introduce.

While computational geometry was one of first fields where conditional lower bounds for algorithms were applied, for example by showing that determining if a point set is in general position is 3SUM hard [30], the progress in conditional lower bounds for dynamic problems has not found widespread application to computational geometry; recent work has been largely confined to improved upper bounds. The only examples before the first version of this paper² relate to (approximate) nearest-neighbor search under different metrics (see the paper by Rubinfeld [51], the introductory article by Bringmann [14] as well as a preprint by Ko and Song [39]), a paper by Lau and Ritossa [44] with results for orthogonal range update on weighted point sets and an (unconditional) lower bound by Chan [17] for a dynamic version of Klee’s Measure Problem. After a previous version of the present paper appeared on arXiv, and independent of our work, Jin and Xu [37] studied generalized versions of the OMv and BMM problems and proved polynomial lower bounds for various dynamic problems based on their hardness, among which Dynamic 2D Orthogonal Range Color Counting, Counting Maximal Points, Dynamic Klee’s measure problem for unit hypercubes and Chan’s Halfspace Problem.

In this work, we exploit the results of Pătrașcu, Kopelowitz et al., V. V. Williams and Xu, and Henzinger et al. to give conditional polynomial lower bounds for a variety of dynamic problems in computational geometry, based on the hardness of 3SUM, APSP and Online Boolean Matrix-Vector Multiplication. Almost all the problems we study here share the common characteristic of being about computing a single global metric for a set of objects in space subject to updates. Moreover, in the static case (where there are no updates) most of these metrics can be computed in worst-case $O(n \log n)$ time using standard computational

² We exclude from this list examples where (conditional) bounds on the static case trivially imply polynomial bounds on the dynamic case.

geometry results. In particular, we show conditional hardness results for orthogonal range marking, maintaining the number of maximal or extremal points in a set of points in \mathbb{R}^3 , dynamic approximate square set cover, problems related to Klee’s Measure Problem, problems related to finding the largest empty disk in a set of points, testing whether a set of disks covers a given rectangle, and querying for the size of the i ’th convex layer of a set of points in the plane. We also give an unconditional lower bound for the incremental Hypervolume Indicator problem in \mathbb{R}^3 , where the goal is to maintain the volume of the union of a set of axis-aligned boxes which all have the origin as one of their vertices.

The most basic of these problems, and the one we present first, is Square Range Marking: given a set of n initially unmarked points in the plane, preprocess them to allow marking of the points in any given axis-aligned square and testing if there is any unmarked point. This encompasses the idea of augmenting a range query structure where augmentations can be applied to all data in a query range; a mark is the simplest such augmentation. While many variants of augmented orthogonal range queries have been studied (especially in the static case) [4, 21, 22, 32, 33, 36, 42, 45, 46, 52], this natural variant has been given little attention. This is perhaps no coincidence, as we show that the straightforward polynomial-time solution based on kd-trees is likely almost optimal, in contrast to standard 1-D range marking and other augmentation problems which are easily handled by suitable variants of BSTs [26, Ch. 14].

Lau and Ritossa [44] give similar lower bounds for data structures on weighted points, conditioned on the hardness of Online Boolean Matrix-Vector Multiplication, but explicitly leave open questions on points which have a color or a “category.” They show for example a lower bound for a data structure which allows to increment the weight of all points in an orthogonal range and to query the sum of weights for all points in a given range, as well as for variants of this problem.

1.1 Setting and computational model

We work in the standard Word RAM model, with words of $w = \Theta(\log n)$ bits unless otherwise stated, and for randomized algorithms we assume access to a perfect source of randomness. We will base our conditional lower bounds on the following well known hardness conjectures.

► **Conjecture 1** (3SUM conjecture). *The following problem (3SUM) requires $n^{2-o(1)}$ expected time to solve: given a set of n integers in $\{-n^3, \dots, n^3\}$, decide if three of them sum up to 0 .³*

► **Conjecture 2** (APSP conjecture). *The following problem (APSP) requires $n^{3-o(1)}$ expected time to solve: given an integer-weighted directed graph G on n vertices with no negative cycles, compute the distance between every pair of vertices in G .*

The 3SUM problem can easily be solved in $O(n^2)$ time, while APSP can be solved in cubic time by the Floyd–Warshall algorithm, for example. The best known methods improve these runtimes by subpolynomial factors [19, 54].

In addition to being the basis for these standard conjectures in fine-grained complexity, the 3SUM problem and the APSP problem are related in other ways (see [56]). In particular, they both fine-grained reduce to the Exact Triangle problem, meaning that if either the 3SUM conjecture or the APSP conjecture is true, then the following conjecture is true.

³ The assumption that the integers are in $\{-n^3, \dots, n^3\}$ is done without loss of generality. In the model we consider one can always reduce the problem to this setting while preserving the expected run-time, via known hashing methods [8].

► **Conjecture 3** (Exact Triangle conjecture). *The following problem (Exact Triangle) requires $n^{3-o(1)}$ expected time to solve: given an integer-weighted graph G and a target weight T , determine if there is a triangle in G whose edge weights sum to T .*

Thus, any bound conditioned on this conjecture also holds conditioned on the 3SUM conjecture or the APSP conjecture. We also consider a conjecture introduced by Henzinger et al. [34], which can be thought of as a weakening of the informal conjecture which says that “combinatorial” matrix multiplication on $n \times n$ matrices requires essentially cubic time (note that the term “combinatorial” is not well defined).

► **Conjecture 4** (OMv conjecture). *The following problem (OMv) requires $n^{3-o(1)}$ expected time to solve:*

We are given a $n \times n$ boolean matrix M . We can preprocess this matrix, after which we are given a sequence of n boolean column-vectors of size n denoted by v_1, \dots, v_n , one by one. After seeing each vector v_i , we must output the product Mv_i before seeing v_{i+1} .

The OMv problem can be solved in total time $O(n^3)$ by the naive algorithm. Here the best known method improves this runtime by a subpolynomial factor [43]. The conjecture was originally introduced in the Monte-Carlo setting (i.e. algorithms with a deterministic runtime but which are allowed to err with a small enough probability). We state it in the Las Vegas setting for the sake of uniformity of presentation. All the results of Henzinger et al. carry over to that setting with no difficulty.

While Henzinger et al. showed that most known lower bounds on dynamic problems derived from the 3SUM conjecture can be derived from the OMv conjecture (and often even strengthened), it is not known whether one conjecture implies the other. For most of our problems we derive polynomial lower bounds from both the OMv conjecture and the Exact Triangle conjecture. In such cases, we still get such lower bounds if at least one of the four considered conjectures is true. Moreover, the reductions used here could also give bounds in the case some of these conjecture fail by a small enough polynomial factor (for example if 3SUM requires $\Omega(n^{4/3})$ time).

Note also that recent work by Chan et al. [23] directly implies that the lower bounds we obtain from the APSP conjecture also hold in the so-called Real RAM model (conditioned on the analogous Real-APSP conjecture) and in restricted versions of the model. For the real versions of the 3SUM and Exact Triangle conjectures, combining our reductions with theirs would also imply polynomial lower bounds for many of the problems considered here, although weaker than the ones we obtain in the Word RAM model.

1.2 Main results

In the full version of this paper we obtain (conditional) polynomial lower bounds for a variety of dynamic geometric problems, and an unconditional bound for the incremental Hypervolume Indicator problem in \mathbb{R}^3 . Our bounds are stated as inequalities which imply trade-offs between achievable update and query times. The lower bounds we get on the maximum of both are summarized in Table 1, together with known upper bounds. Note that the bounds we get for squares or square ranges imply the same bounds for rectangles or general orthogonal ranges, although we sometimes get better trade-offs in these cases. Here we focus on some results for Square Range Marking, Counting Extremal Points in \mathbb{R}^3 and unweighted Square Set Cover, in the fully-dynamic setting. The other results can be found in the full version [28].

■ **Table 1** Non-trivial known upper bounds and new (at the time of the first version of this paper being made public) lower bounds on the maximum over update and query time derived from the Exact Triangle conjecture, the OMv conjecture or (in the case of the Hypervolume Indicator problem) unconditionally. The \tilde{O} notation hides polylog factors, while the O^* notation hides factors which are $o(n^\varepsilon)$ for an arbitrarily small constant $\varepsilon > 0$. All upper bounds are for data structures with at most $O^*(n)$ preprocessing. Note that the lower bounds for Square Range Marking also hold in the case of a static set of points (with some assumptions on preprocessing time) and that the lower bound for the Depth Problem derived from the OMv conjecture also holds for amortized runtime in the incremental setting. The lower bound obtained for counting maximal points has since been superseded by the more general result of Jin and Xu [37] who obtain lower bounds also in higher dimension.

Problem	Upper Bound	Lower Bound
Square Range Marking [§2.2,28]	$\tilde{O}(n^{1/2})^{\dagger,\ddagger}$ [15]	From Exact Triangle: $n^{1/4-o(1)}^{\dagger}$ From OMv: $n^{1/2-o(1)}^{\dagger}$
Counting Extremal Points in \mathbb{R}^3 [§3,28]	$O^*(n^{7/8})^{\dagger}$ [16]	From Exact Triangle: $n^{1/5-o(1)}^{\dagger,\ddagger}$ $n^{1/4-o(1)}^{\ddagger,\S}$ From OMv: $n^{1/2-o(1)}^{\dagger,\ddagger}$
Largest Empty Disk in Query Region [28]	$O^*(n^{11/12})^{\ddagger}$ [18]	
Largest Empty Disk in a Set of Disks [28]		
Rectangle Covering with Disks [28]		
Square Covering with Squares [28]	$\tilde{O}(n^{1/2})^{\ddagger}$ [57]	
Convex Layer Size in \mathbb{R}^2 [28]		
Counting Maximal Points in \mathbb{R}^3 [§3,28]	$\tilde{O}(n^{2/3})^{\ddagger}$ [18]	From Exact Triangle: $n^{1/4-o(1)}^{\dagger}$ $n^{1/3-o(1)}^{\dagger}$
$O(n^\alpha)$ -approx. Weighted Square Set Cover [28]		
Klee's Measure Problem with Squares [28]	$\tilde{O}(n^{1/2})^{\ddagger}$ [57]	From OMv: $n^{1/2-o(1)}^{\dagger,\ddagger}$
Discrete KMP with Squares [28]	$O(n^{1/2})^{\dagger,\ddagger}$ [58]	
Depth Problem with Squares [28]	$\tilde{O}(n^{1/2})^{\ddagger}$ [57]	From Exact Triangle: $n^{1/3-o(1)}^{\dagger,\ddagger}$ From OMv: $n^{1/2-o(1)}^{\dagger,\ddagger}$
$O(1)$ -approximate Square Set Cover [§4,28]	$O^*(n^{1/2})^{\ddagger}$ [20]	From OMv: $n^{1/3-o(1)}^{\dagger,\ddagger}$
Hypervolume Indicator in \mathbb{R}^3 [28]	$\tilde{O}(n^{2/3})^{\ddagger}$ [18]	$\Omega(\sqrt{n})^{\#}$

\dagger per-operation runtime in the incremental setting.

\ddagger amortized runtime in the fully-dynamic setting.

\S assuming $n^{1+o(1)}$ expected preprocessing time.

$\#$ unconditional lower bound in the incremental setting on amortized time, assuming at most polynomial time preprocessing, or on worst-case time without preprocessing assumptions.

Some of the lower bounds reveal interesting separations between geometric dynamic problems whose operations can be supported in subpolynomial or $O(n^\varepsilon)$ time and similar problems which require polynomial time with a fixed exponent (under the hardness conjectures we consider).

- Orthogonal range queries with dynamic updates on single points can be done with polylog time operations, while dynamic updates on orthogonal ranges of points require polynomial time.
- Dynamically maintaining maximal points in a point set can be done in polylog time in \mathbb{R}^2 , while maintaining only their number in \mathbb{R}^3 already requires polynomial time.
- The same separation between dimensions 2 and 3 applies for maintaining (the number of) extremal points.
- Related to the previous point, the ability to query for the size of any convex layer on a dynamic set of points in \mathbb{R}^2 requires polynomial time (compared to polylog time when we are only interested in the first convex layer, i.e. the convex hull).
- Maintaining a $O(1)$ -approximation for the size of dynamic unit square set cover can be done in $2^{O(\sqrt{\log n})}$ amortized time per update [20], while maintaining the size of a $O(n^\alpha)$ -approximation (for a constant $0 \leq \alpha < 1$) requires polynomial time for arbitrarily sized squares (with an exponent dependent on α).
- In the weighted case of the previous problem, we also get such a separation: $O(1)$ -approximate weighted unit square set cover can be done in $O(n^\varepsilon)$ time [20] while $O(n^\alpha)$ -approximate weighted dynamic square set cover requires polynomial time, with an exponent independent of α .

2 The general approach

In all the problems we consider, we have a data structure D which maintains a set S of $O(n)$ geometric objects, supporting some form of update and query (a query is any operation which never impacts the result of any subsequent operation). We say that a data structure (or that the set of objects it maintains) is *incremental* when it allows updates which consist of inserting a new object in S . We use the term *fully-dynamic* when both insertions and deletions are allowed. The set S can be initialized in a preprocessing phase.

2.1 General reduction schemes

All our reductions have the same basic structure based on a geometric view of Pătraşcu’s Multiphase problem [50], where we encode a family $\mathcal{F} = \{F_1, \dots, F_k\}$ of subsets of $\{1, \dots, m\}$ as a grid of objects where the presence (or absence) of an object at the grid coordinates (x, y) encodes $x \in F_y$. We can then select some of the columns $I \in \{1, \dots, k\}$ and a row $j \in \{1, \dots, m\}$, allowing us to test if $I \cap F_j \neq \emptyset$ efficiently. We abstract some of the commonalities of the reductions in the following “general” reduction schemes, so we can focus on the specifics of each problem and avoid repetitions later on. Rather than give the original definition of the Multiphase problem, let us define what it means for a data structure to solve it, as this will make the statements of reductions easier, more uniform, and makes the required constraints on the data structure we consider explicit.

► **Definition 5** (Solving the Multiphase problem). *Let $\mathcal{F} = \{F_1, \dots, F_k\}$ be a family of k subsets of $\{1, 2, \dots, m\}$. Let $s_{\mathcal{F}} = \sum_{F \in \mathcal{F}} |F|$. Consider a data structure D with an undo operation⁴ which maintains a set S of $O(n)$ objects with expected preprocessing time $O(t_p)$, expected amortized update time $O(t_u)$ and expected amortized query time $O(t_q)$. Suppose it allows us to do the following.*

- (Step 1) *First, we read \mathcal{F} and store a set of n objects in S using only the preprocessing operation of D .*
- (Step 2) *Then, we receive a subset $J \subset \{1, 2, \dots, m\}$ and perform u_J updates on S .*
- (Step 3) *Finally, we are given an index $1 \leq i \leq k$ and after $O(1)$ updates and queries on S we decide if $J \cap F_i \neq \emptyset$.*

Assume that the time of each of these three steps is dominated by the time of the operations on D and that in each step, the only information available from the previous steps is what is accessible through D . Let $t_{uq} = t_q$ if only queries are performed in Step 3, otherwise let $t_{uq} = t_u + t_q$.

We say that such a data structure solves the Multiphase problem.

As mentioned in the introduction, Pătraşcu gave lower bounds on the time required to solve the Multiphase problem conditioned on the 3SUM conjecture and reduced this problem to various dynamic problems. His reduction from 3SUM has since been tightened by Kopelowitz et al. [41] and reductions from the Exact Triangle and OMv conjectures have been found by Vassilevska Williams and Xu [56] and Henzinger et al. [34] respectively.

We summarize the implications from these works for different parameters in the following theorems. While this results in somewhat verbose statements, we chose this approach in order to streamline the reductions in this paper and to make the lower bounds we obtain explicit in terms n .

► **Theorem 6.** *Let D be a data structure which solves the Multiphase problem. If the Exact Triangle conjecture is true (or in particular if either the 3SUM or APSP conjecture is true), then for any $0 < \gamma < 1$:*

- (Scenario 1) *If $n = O(m \cdot k)$ and $u_J = O(m)$, we have*

$$t_p + t_u \cdot n + t_{uq} \cdot n^{\frac{1+\gamma}{3-2\gamma}} = \Omega\left(n^{\frac{2}{3-2\gamma}-o(1)}\right).$$

- (Scenario 2) *If $n = O(m \cdot k)$ and $u_J = O(|J|)$, we have*

$$t_p + t_u \cdot n^{\frac{2-\gamma}{3-2\gamma}} + t_{uq} \cdot n^{\frac{1+\gamma}{3-2\gamma}} = \Omega\left(n^{\frac{2}{3-2\gamma}-o(1)}\right).$$

- (Scenario 3) *If $n = O(s_{\mathcal{F}})$ and $u_J = O(m)$, we have*

$$t_p + t_u \cdot n^{\frac{3-2\gamma}{2-\gamma}} + t_{uq} \cdot n^{\frac{1+\gamma}{2-\gamma}} = \Omega\left(n^{\frac{2}{2-\gamma}-o(1)}\right).$$

- (Scenario 4) *If $n = O(s_{\mathcal{F}})$ and $u_J = O(|J|)$, we have*

$$t_p + t_u \cdot n + t_{uq} \cdot n^{\frac{1+\gamma}{2-\gamma}} = \Omega\left(n^{\frac{2}{2-\gamma}-o(1)}\right).$$

Note that for incremental (or fully-dynamic) data structures where we can insert objects, we can always assume $t_p = O(t_u \cdot n)$ by inserting the $O(n)$ initial objects individually.

⁴ A data structure is said to have an undo operation if for any update U there is complementary update U' so that if U and U' are executed sequentially the results of subsequent operations are identical to the case where U and U' were never executed. This requirement is easily satisfied in structures that maintain a set and have insertion and deletion update operations.

The results of Henzinger et al. [34] imply that whenever we have such lower bounds from the hardness of Exact Triangle, we can get stronger bounds if we assume hardness of the OMv problem instead.

► **Theorem 7.** *Let D be a data structure which solves the Multiphase problem. Assume $n = O(m^{c_1} \cdot k^{c_2})$ for some constants $c_1, c_2 > 0$, $u_J = O(m)$, and the expected preprocessing time t_p is at most polynomial in n . If the OMv conjecture is true, then for any $0 < \gamma < 1$,*

$$t_u \cdot n^\gamma + t_{uq} \cdot n^{\frac{1-c_1\gamma}{c_2}} = \Omega\left(n^{\frac{1+(c_2-c_1)\gamma}{c_2}-o(1)}\right).$$

In particular if $n = O(m \cdot k)$ (as is the case in the four scenarios of Theorem 6), for any $0 < \gamma < 1$ we have $t_u \cdot n^\gamma + t_{uq} \cdot n^{1-\gamma} = \Omega(n^{1-o(1)})$. For $\gamma = 1/2$, we thus have $t_u + t_q = \Omega(n^{1/2-o(1)})$.

These results follow from straightforward adaptations of Pătraşcu’s proofs [50] together with the more recent results from Williams and Xu [56] and Henzinger et al. [34], and are implicit in the two latter papers. To apply these theorems, we need data structures with an undo operation. When considering structures in the fully-dynamic setting where updates consist of inserting or deleting an object, then this requirement is automatically satisfied. For structures with guarantees on the runtime per operation (rather than amortized guarantees), we can use the following standard technique (see for example [47, Theorem 2.1]).

► **Lemma 8.** *Any data structure with guarantees on the runtime per operation (non-amortized) can be augmented to support an undo operation with the same guarantees.*

From now on, whenever we consider a structure with per-operation runtime guarantees, we assume (without loss of generality) that it has been augmented to support undo.

2.2 An example: Square Range Marking

We illustrate the use of these theorems on the following problem.

Square Range Marking. Preprocess a static set of n initially unmarked points, where an update consists of marking all points in a given axis-aligned square range and a query returns if there is any unmarked point in the set.

Here the dynamic part of the problem is rather limited as only the markings of the points can change after an update, the set of points itself is static. The updates are even monotone in the sense that once a point has been marked it is never unmarked (in particular, the number of unmarked points can never increase). Even for this seemingly simple problem, we can use Theorems 6 and 7 to get the following (conditional) polynomial lower bounds.

► **Theorem 9.** *Let D be a data structure for Square Range Marking with t_p expected preprocessing time and t_u expected time per update (i.e. non-amortized). If the Exact Triangle conjecture holds, then*

$$t_p + t_u \cdot \left(n^{\frac{3-2\gamma}{2-\gamma}} + n^{\frac{1+\gamma}{2-\gamma}}\right) + t_q \cdot n^{\frac{1+\gamma}{2-\gamma}} = \Omega\left(n^{\frac{2}{2-\gamma}-o(1)}\right).$$

If the OMv conjecture holds and t_p is at most polynomial then for any $0 < \gamma < 1$

$$t_u \cdot (n^{1-\gamma} + n^\gamma) + t_q \cdot n^{1-\gamma} = \Omega\left(n^{1-o(1)}\right).$$

In particular, by setting $\gamma = 1/2$, we have $t_u + t_q = \Omega(n^{1/2-o(1)})$.

Proof. It suffices to show that such a data structure fits the conditions of Scenario 3 in Theorem 6. Let $\mathcal{F} = \{F_1, \dots, F_k\}$ be a family of k subsets of $\{1, 2, \dots, m\}$.

We perform Step 1 by initializing D with the following points: for each $1 \leq i \leq k$ and $1 \leq j \leq m$ for which $j \in F_i$, we put a point $p_{i,j}$ at coordinates $((k+2)j+1, i+1)$. The total number of points is $n = s_{\mathcal{F}}$.

To perform Step 2 when given $J \subset \{1, 2, \dots, m\}$, we mark the points inside a square range of side-length $k+2$ whose lower-left corner has coordinates $((k+2)j, 1)$, for all $j \notin J$. This requires $O(m)$ updates on D . The unmarked points are exactly the $p_{i,j}$'s such that $j \in J$.

In Step 3, when given an index $1 \leq i' \leq k$, we mark the points inside the two squares of side-length $(k+2)m$ whose lower-left corners lie at coordinates $(0, i'+1/2)$ and $(0, i'-(k+2)m-1/2)$ respectively. Now there is an unmarked point if and only if there is some point $p_{i,j}$ such that $j \in J$ and $p_{i,j}$ was not marked by these two last updates. This is the case if and only if $i = i'$. By construction, such a point exists if and only if there is some $j \in J$ such that $j \in F_{i'}$ (i.e. $J \cap F_{i'} \neq \emptyset$). Thus, we can answer a Step 3 query after two more updates to D .

By applying Theorems 6 and 7 we get the result. \blacktriangleleft

If we assume truly subquadratic expected preprocessing time we get polynomial lower bounds on t_u or t_q from the Exact Triangle conjecture. The bounds from the OMv conjecture are almost tight, as an upper bound can easily be obtained by taking a two-dimensional kd-tree [10] and augmenting it by adding markers to the nodes indicating if the points in the corresponding subtrees are marked. We then get a data structure with $O(n \log n)$ worst-case preprocessing time and $O(\sqrt{n})$ worst-case time per update. As noted by Cardinal et al. [15], using standard dynamization techniques such a data structure can even be made to support insertion and deletion of points in $O(\log^2 n)$ worst-case time.

3 Counting the number of extremal points in \mathbb{R}^3

We show polynomial lower bounds for the following problem, which in the plane can be solved in polylog worst-case time per operation by known techniques [48].

Counting Extremal Points in \mathbb{R}^3 . Maintain a dynamic set of $O(n)$ points in \mathbb{R}^3 and allow for queries counting the number of extremal points in the set.

► **Theorem 10.** *Let D be a fully-dynamic data structure for Extremal Points in \mathbb{R}^3 with t_p expected preprocessing time, t_u expected amortized update time and t_q expected amortized query time. If the Exact Triangle conjecture holds, then*

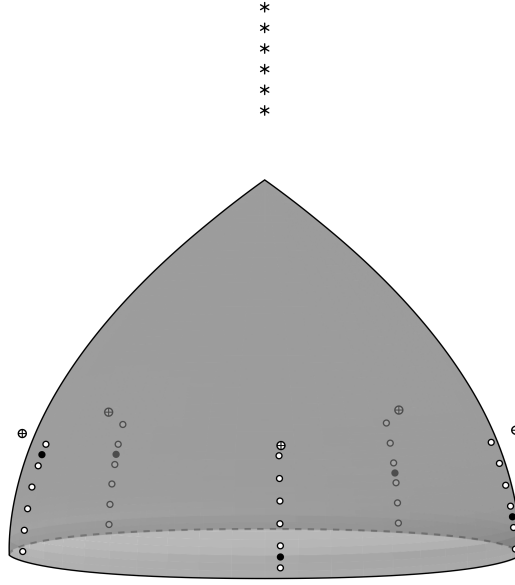
$$t_p + t_u \cdot \left(n^{\frac{2-\gamma}{3-2\gamma}} + n^{\frac{1+\gamma}{3-2\gamma}} \right) + t_q \cdot n^{\frac{1+\gamma}{3-2\gamma}} = \Omega \left(n^{\frac{2}{3-2\gamma}-o(1)} \right).$$

If the OMv conjecture holds, then for any $0 < \gamma < 1$, $t_u \cdot (n^{1-\gamma} + n^\gamma) + t_q \cdot n^{1-\gamma} = \Omega(n^{1-o(1)})$. In particular, by setting $\gamma = 1/2$, we have $t_u + t_q = \Omega(n^{1/2-o(1)})$.

Because we can assume $t_p = O(t_u \cdot n)$ this also implies that under the Exact Triangle conjecture we have $t_u = \Omega(n^{1/5-o(1)})$.

Before proving this theorem, let us introduce some notation. We let $\mathcal{F} = \{F_1, \dots, F_k\}$ be a family of k subsets of $\{1, \dots, m\}$, where $m, k \geq 5$. Here we work in cylindrical coordinates (r, θ, z) (where this would denote the point $(r \cos \theta, r \sin \theta, z)$ in Cartesian coordinates).

Let $R = 4k^2m^2$. For all $0 \leq i \leq k$ and $1 \leq j \leq m$, let $q_{i,j}$ be the point with cylindrical coordinates $(R - (2i+1)^2, \frac{2\pi}{m}j, 2i+1)$. Similarly, for all $1 \leq j \leq m$ and $1 \leq i \leq k$ such that $j \in F_i$, let $p_{i,j}$ be the point with cylindrical coordinates $(R - (2i)^2, \frac{2\pi}{m}j, 2i)$. We let $S_{\mathcal{F}}$ denote the set consisting of all these points.



■ **Figure 1** Illustration (not to scale) of the set of points obtained for $m = k = 5$ and the family of sets $\mathcal{F} = \{\{1\}, \{2\}, \dots, \{5\}\}$. The points $q_{i,j}$ are represented in white, the points $p_{i,j}$ in black and all these points lie on the translucent gray surface. The points b_j are represented with a cross in a white circle and are above the translucent gray surface. The asterisks represent the points of the form t_i and lie on the axis of rotational symmetry of the translucent gray surface.

For all $1 \leq j \leq m$, we let b_j denote the point with cylindrical coordinates $(R-1, \frac{2\pi}{m}j, 2k+2)$. For all $1 \leq i \leq k+1$, we let t_i denote the point lying on the longitudinal axis (the z -axis) at height $\frac{R+(2i-1)^2}{2(2i-1)}$. Note that for all $1 \leq i_1 \leq i_2 \leq k+1$, the point t_{i_1} is higher on the z -axis than t_{i_2} and that all points t_i have a larger z -coordinate than all other previously defined points. See Figure 1 for an illustration.

► **Lemma 11.** *Let S be a set of points such that $S_{\mathcal{F}} \subset S \subset S_{\mathcal{F}} \cup \{b_j \mid 1 \leq j \leq m\} \cup \{t_i \mid 1 \leq i \leq k\}$. Let $1 \leq j' \leq m$ and $1 \leq i' \leq k$. Then:*

- *The point $q_{0,j'}$ is extremal.*
- *The point $q_{i',j'}$ is extremal if and only if $b_{j'} \notin S$ and for all $1 \leq i \leq i'$, $t_i \notin S$. If $p_{i',j'} \in S$ (i.e. $j' \in F_{i'}$), then the same holds for $p_{i',j'}$.*
- *If $t_{i'} \in S$, then $t_{i'}$ is extremal if and only if for all $1 \leq i < i'$, $t_i \notin S$.*

Moreover, this remains true even if all points are arbitrarily perturbed by moving them a distance of at most $1/R^2$.

See the full version of the paper [28] for the proof of this lemma.

Proof of Theorem 10. It suffices to show that such a data structure fits the conditions of Scenario 2 in Theorem 6. Let $\mathcal{F} = \{F_1, \dots, F_k\}$ be a family of k subsets of $\{1, 2, \dots, m\}$. We use the notation of Lemma 11. We first describe the procedure without discussing issues of finite precision and later show how this can be carried out on a Word RAM machine with words of $O(\log n)$ bits.

We perform Step 1 by initializing D with all points of the form $q_{i,j}$, $p_{i,j}$ and b_j . This costs t_u expected time, for a total number of points $n = \Theta(m \cdot k)$.

To perform Step 2 when given $J \subset \{1, 2, \dots, m\}$, we delete the points b_j for all $j \in J$. This requires $O(|J|)$ updates on D .

In Step 3, when given an index $1 \leq i' \leq k$ we start by inserting the point $t_{i'+1}$ to D and getting the count c of extremal points. By Lemma 11, the extremal points of S at this point are exactly those of the following 6 types:

1. the point $t_{i'+1}$,
2. the points b_j for all j such that $j \notin J$,
3. the points $q_{i,j}$ for all i, j such that $j \in J$ and $i < i'$,
4. the points $p_{i,j}$ for all i, j such that $j \in F_i$, $j \in J$ and $i < i'$.
5. the points $q_{i',j}$ for all j such that $j \in J$,
6. the points $p_{i',j}$ for all j such that $j \in F_{i'}$ and $j \in J$.

To answer the query, we want to know if the number of points of the last type is greater than 0. We know that the number of points of the fifth category is exactly $|J|$. Notice that if we now insert $t_{i'}$ to D and get the new count c' of extremal points, we are counting exactly the first four categories of points, where we have replaced $t_{i'+1}$ with $t_{i'}$. Thus, we can test if the number of points of the last category is 0 simply by testing if $c - c' = |J|$. We can thus perform Step 3 with $O(1)$ updates.

We can adapt this to work on a Word RAM machine with words of length $w \geq \log n$ by moving the points to vertices of the integer lattice (after appropriate scaling). This is detailed in the full version of the paper [28]. By applying Theorems 6 and 7 we get the result. \blacktriangleleft

In the fully-dynamic setting, Chan [18] gives a data structure for this problem with $O(n^{1+\varepsilon})$ preprocessing time and $O(n^{11/12+\varepsilon})$ amortized update and query time, for an arbitrary $\varepsilon > 0$. In the more restricted semi-online setting (which generalizes the incremental case), another paper by the same author [16] gives a data structure with $O(n^{1+\varepsilon})$ preprocessing time and $O(n^{7/8+\varepsilon})$ worst-case time per operation.

4 Dynamic geometric Set Cover with squares

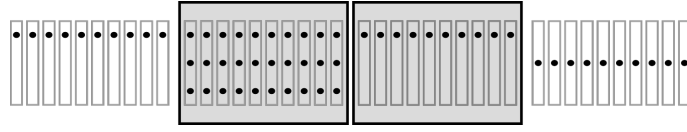
In this section we answer a question by Chan et al. [20], by giving a conditional polynomial lower bound on the time required to approximately maintain (the size of) a dynamic square set cover in the plane under range updates.

Dynamic Square Set Cover. Maintain a set S of n points and axis-aligned squares in the plane to support queries asking for the size of the smallest subset of squares which covers all points.

Even the static version of this problem with unit squares is NP-complete [29], thus the focus on approximations. Chan et al. [20] recently gave a $O(1)$ -approximate solution in the fully dynamic case where both squares and points may be inserted or deleted. This (Monte Carlo randomized) solution achieves $O(n^{1/2+\varepsilon})$ amortized update and query time. The authors ask if there is a conditional polynomial lower bound for this problem. We show the following.

► **Theorem 12.** *Let $0 \leq \alpha < 1$ be an efficiently computable⁵ constant. If there is a fully-dynamic data structure for $O(n^\alpha)$ -approximate Dynamic Square Set Cover with t_u expected amortized update time and t_q expected query time, then the Multiphase problem can be solved with $n = O\left(k^{1/(1-\alpha)} \cdot m^{2/(1-\alpha)^2}\right)$, $t_{uq} = t_u + t_q$ and $u, J = O(m)$.*

⁵ We say that a number α is efficiently computable if there is an algorithm which, for any $k \geq 0$, can output the first k bits of α in $O(\text{poly}(k))$ time.



■ **Figure 2** Illustration (not to scale) of the instance obtained after Step 2 in the proof of Theorem 12, for a data structure with a constant approximation ratio of $\beta = 3/2$. The illustrated instance has $m = 4$, $k = 3$, $\mathcal{F} = \{\{1, 2, 3\}, \{2, 4\}, \{2\}\}$, $J = \{1, 4\}$ and $c = 10$.

Together with Theorem 7 this implies polynomial lower bounds under the OMv conjecture, for all $0 \leq \alpha < 1$. In particular, for $\alpha = 0$ (i.e. for a constant approximation factor) it implies $t_u + t_q = \Omega(n^{1/3-o(1)})$.

Proof. We first prove the result in the case of arbitrary axis-aligned rectangles and then show how to adapt it to use only squares. Let $\mathcal{F} = \{F_1, \dots, F_k\}$ be a family of k subsets of $\{1, 2, \dots, m\}$. Suppose D achieves an approximation ratio of at most $\beta \cdot n^\alpha$ for some constant integer $\beta > 1$ (we can assume this without loss of generality). Let c be an integer value which we specify later.

We perform Step 1 by initializing D with the following points and rectangles:

- for each $1 \leq i \leq k$ and $1 \leq j \leq m$ for which $j \in F_i$, we put a point $p_{i,j}^a$ at coordinates $(c \cdot j + a, i)$ for each $0 \leq a < c$;
- for each $1 \leq j \leq m$ and each $0 \leq a < c$, we put a thin vertical rectangle which covers exactly the points of the form $p_{\bullet,j}^a$.

The total number of points and rectangles at this point is $n_1 = s_{\mathcal{F}} + m \cdot c \leq m \cdot (k + 1) \cdot c$. Set c to be the smallest integer such that $c > \beta(n_1 + m + 2)^\alpha \cdot (m + 2)$ (note that the value of n_1 depends on c , but c is nonetheless well defined and can be computed in $O(\text{polylog}(m \cdot k))$ time). We then have $n_1 = O(k^{1/(1-\alpha)} \cdot m^{2/(1-\alpha)^2})$.

To perform Step 2 when given $J \subset \{1, 2, \dots, m\}$, we insert a rectangle covering all points of the form $p_{\bullet,j}^a$ for each $j \notin J$. This requires $O(m)$ updates on D . See Figure 2 for an illustration of the first two steps.

In Step 3, when given an index $1 \leq i' \leq k$, we insert two rectangles: the first covers all points $p_{i,\bullet}^a$ with $i < i'$ and the second covers all points $p_{i,\bullet}^a$ with $i > i'$. The total number of points and rectangles at this point is $n \leq n_1 + m + 2$. Now, if $J \cap F_{i'} = \emptyset$ then all points can be covered with at most $m + 2$ rectangles: the (at most) m rectangles inserted in step 2 together with the two rectangles inserted in Step 3. On the other hand, if there is some $j \in J \cap F_{i'}$, then the points of the form $p_{i,j}^a$ can only be covered by choosing c thin rectangles created in Step 1. Thus, any approximation of the set cover with a ratio better than $\frac{c}{m+2}$ suffices to distinguish between the two cases. Moreover, we have $\frac{c}{m+2} > \beta \cdot (n_1 + m + 2)^{\alpha \frac{m+2}{m+2}} \geq \beta \cdot n^\alpha$. We can then answer a Step 3 query by asking the data structure for a $\beta \cdot n^\alpha$ approximation of the size of the minimum set cover.

To see how to adapt this reduction using only squares, notice that we can increase the height of any rectangle without affecting the results. Thus, we can stretch the whole configuration of points and rectangles horizontally until the thinnest vertical rectangles become squares, and adjust the heights of the other rectangles to make them squares as well. ◀

Contrast this lower bound with the case of *unit* axis-aligned squares, for which Chan et al. [20] give a data structure for $O(1)$ -approximation achieving $2^{O(\sqrt{\log n})}$ amortized update and query time.

References

- 1 Amir Abboud and Søren Dahlgaard. Popular conjectures as a barrier for dynamic planar graph algorithms. In Irit Dinur, editor, *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 477–486. IEEE Computer Society, 2016. doi:10.1109/FOCS.2016.58.
- 2 Amir Abboud and Virginia Vassilevska Williams. Popular conjectures imply strong lower bounds for dynamic problems. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pages 434–443. IEEE Computer Society, 2014. doi:10.1109/FOCS.2014.53.
- 3 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.
- 4 Pankaj K. Agarwal, Sathish Govindarajan, and S. Muthukrishnan. Range searching in categorical data: Colored range searching on grid. In Rolf H. Möhring and Rajeev Raman, editors, *Algorithms - ESA 2002, 10th Annual European Symposium, Rome, Italy, September 17-21, 2002, Proceedings*, volume 2461 of *Lecture Notes in Computer Science*, pages 17–28. Springer, 2002. doi:10.1007/3-540-45749-6_6.
- 5 Josh Alman, Matthias Mnich, and Virginia Vassilevska Williams. Dynamic parameterized problems and algorithms. In Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl, editors, *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10-14, 2017, Warsaw, Poland*, volume 80 of *LIPICs*, pages 41:1–41:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017. doi:10.4230/LIPICs.ICALP.2017.41.
- 6 Amihod Amir, Timothy M. Chan, Moshe Lewenstein, and Noa Lewenstein. On hardness of jumbled indexing. In Javier Esparza, Pierre Fraigniaud, Thore Husfeldt, and Elias Koutsoupias, editors, *Automata, Languages, and Programming - 41st International Colloquium, ICALP 2014, Copenhagen, Denmark, July 8-11, 2014, Proceedings, Part I*, volume 8572 of *Lecture Notes in Computer Science*, pages 114–125. Springer, 2014. doi:10.1007/978-3-662-43948-7_10.
- 7 Amihod Amir, Tsvi Kopelowitz, Avivit Levy, Seth Pettie, Ely Porat, and B. Riva Shalom. Mind the gap! - online dictionary matching with one gap. *Algorithmica*, 81(6):2123–2157, 2019. doi:10.1007/s00453-018-0526-2.
- 8 Ilya Baran, Erik D. Demaine, and Mihai Patrascu. Subquadratic algorithms for 3SUM. *Algorithmica*, 50(4):584–596, 2008. doi:10.1007/s00453-007-9036-3.
- 9 Surender Baswana, Shreejit Ray Chaudhury, Keerti Choudhary, and Shahbaz Khan. Dynamic DFS in undirected graphs: breaking the $o(m)$ barrier. In Robert Krauthgamer, editor, *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 730–739. SIAM, 2016. doi:10.1137/1.9781611974331.ch52.
- 10 Jon Louis Bentley. Multidimensional binary search trees used for associative searching. *Commun. ACM*, 18(9):509–517, 1975. doi:10.1145/361002.361007.
- 11 Christoph Berkholz, Jens Keppeler, and Nicole Schweikardt. Answering conjunctive queries under updates. In Emanuel Sallinger, Jan Van den Bussche, and Floris Geerts, editors, *Proceedings of the 36th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2017, Chicago, IL, USA, May 14-19, 2017*, pages 303–318. ACM, 2017. doi:10.1145/3034786.3034789.
- 12 Christoph Berkholz, Jens Keppeler, and Nicole Schweikardt. Answering UCQs under updates and in the presence of integrity constraints. In Benny Kimelfeld and Yael Amerdamer, editors, *21st International Conference on Database Theory, ICDT 2018, March 26-29, 2018, Vienna, Austria*, volume 98 of *LIPICs*, pages 8:1–8:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPICs.ICDT.2018.8.

- 13 Karl Bringmann. Fine-grained complexity theory (tutorial). In Rolf Niedermeier and Christophe Paul, editors, *36th International Symposium on Theoretical Aspects of Computer Science, STACS 2019, March 13-16, 2019, Berlin, Germany*, volume 126 of *LIPICs*, pages 4:1–4:7. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPICs.STACS.2019.4.
- 14 Karl Bringmann. Fine-grained complexity theory: Conditional lower bounds for computational geometry. In Liesbeth De Mol, Andreas Weiermann, Florin Manea, and David Fernández-Duque, editors, *Connecting with Computability - 17th Conference on Computability in Europe, CiE 2021, Virtual Event, Ghent, July 5-9, 2021, Proceedings*, volume 12813 of *Lecture Notes in Computer Science*, pages 60–70. Springer, 2021. doi:10.1007/978-3-030-80049-9_6.
- 15 Jean Cardinal, John Iacono, and Grigorios Koumoutsos. Worst-case efficient dynamic geometric independent set. In Petra Mutzel, Rasmus Pagh, and Grzegorz Herman, editors, *29th Annual European Symposium on Algorithms, ESA 2021, September 6-8, 2021, Lisbon, Portugal (Virtual Conference)*, volume 204 of *LIPICs*, pages 25:1–25:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPICs.ESA.2021.25.
- 16 Timothy M. Chan. Semi-online maintenance of geometric optima and measures. *SIAM J. Comput.*, 32(3):700–716, 2003. doi:10.1137/S0097539702404389.
- 17 Timothy M. Chan. A (slightly) faster algorithm for Klee’s measure problem. *Computational Geometry*, 43(3):243–250, 2010. Special Issue on 24th Annual Symposium on Computational Geometry (SoCG’08). doi:10.1016/j.comgeo.2009.01.007.
- 18 Timothy M. Chan. Dynamic geometric data structures via shallow cuttings. *Discret. Comput. Geom.*, 64(4):1235–1252, 2020. doi:10.1007/s00454-020-00229-5.
- 19 Timothy M. Chan. More logarithmic-factor speedups for 3SUM, (median, +)-convolution, and some geometric 3SUM-hard problems. *ACM Trans. Algorithms*, 16(1):7:1–7:23, 2020. doi:10.1145/3363541.
- 20 Timothy M. Chan, Qizheng He, Subhash Suri, and Jie Xue. Dynamic geometric set cover, revisited. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 3496–3528, 2022. doi:10.1137/1.9781611977073.139.
- 21 Timothy M. Chan and Zhengcheng Huang. Dynamic colored orthogonal range searching. In Petra Mutzel, Rasmus Pagh, and Grzegorz Herman, editors, *29th Annual European Symposium on Algorithms, ESA 2021, September 6-8, 2021, Lisbon, Portugal (Virtual Conference)*, volume 204 of *LIPICs*, pages 28:1–28:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPICs.ESA.2021.28.
- 22 Timothy M. Chan and Yakov Nekrich. Better data structures for colored orthogonal range reporting. In Shuchi Chawla, editor, *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pages 627–636. SIAM, 2020. doi:10.1137/1.9781611975994.38.
- 23 Timothy M. Chan, Virginia Vassilevska Williams, and Yinzhan Xu. Hardness for triangle problems under even more believable hypotheses: reductions from real APSP, real 3SUM, and OV. In Stefano Leonardi and Anupam Gupta, editors, *STOC ’22: 54th Annual ACM SIGACT Symposium on Theory of Computing, Rome, Italy, June 20 - 24, 2022*, pages 1501–1514. ACM, 2022. doi:10.1145/3519935.3520032.
- 24 Timothy M. Chan and Da Wei Zheng. Hopcroft’s problem, log-star shaving, 2d fractional cascading, and decision trees. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 190–210, 2022. doi:10.1137/1.9781611977073.10.
- 25 Lijie Chen, Erik D. Demaine, Yuzhou Gu, Virginia Vassilevska Williams, Yinzhan Xu, and Yuancheng Yu. Nearly optimal separation between partially and fully retroactive data structures. In David Eppstein, editor, *16th Scandinavian Symposium and Workshops on Algorithm Theory, SWAT 2018, June 18-20, 2018, Malmö, Sweden*, volume 101 of *LIPICs*, pages 33:1–33:12. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPICs.SWAT.2018.33.
- 26 Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms, 3rd Edition*. MIT Press, 2009. URL: <http://mitpress.mit.edu/books/introduction-algorithms>.

- 27 Søren Dahlgaard. On the hardness of partially dynamic graph problems and connections to diameter. In Ioannis Chatzigiannakis, Michael Mitzenmacher, Yuval Rabani, and Davide Sangiorgi, editors, *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, volume 55 of *LIPIcs*, pages 48:1–48:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016. doi:10.4230/LIPIcs.ICALP.2016.48.
- 28 Justin Dallant and John Iacono. Conditional lower bounds for dynamic geometric measure problems. *CoRR*, abs/2112.10095, 2022. arXiv:2112.10095.
- 29 Robert J. Fowler, Mike Paterson, and Steven L. Tanimoto. Optimal packing and covering in the plane are np-complete. *Inf. Process. Lett.*, 12(3):133–137, 1981. doi:10.1016/0020-0190(81)90111-3.
- 30 Anka Gajentaan and Mark H. Overmars. On a class of $o(n^2)$ problems in computational geometry. *Comput. Geom.*, 5:165–185, 1995. doi:10.1016/0925-7721(95)00022-2.
- 31 Allan Grønlund and Seth Pettie. Threesomes, degenerates, and love triangles. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pages 621–630. IEEE Computer Society, 2014. doi:10.1109/FOCS.2014.72.
- 32 Prosenjit Gupta, Ravi Janardan, Saladi Rahul, and Michiel H. M. Smid. Computational geometry: Generalized (or colored) intersection searching. In Dinesh P. Mehta and Sartaj Sahni, editors, *Handbook of Data Structures and Applications*, chapter 67, pages 1042–1057. CRC Press, 2nd edition, 2018. URL: <https://www-users.cs.umn.edu/~sala0198/Papers/ds2-handbook.pdf>.
- 33 Prosenjit Gupta, Ravi Janardan, and Michiel H. M. Smid. Further results on generalized intersection searching problems: Counting, reporting, and dynamization. *J. Algorithms*, 19(2):282–317, 1995. doi:10.1006/jagm.1995.1038.
- 34 Monika Henzinger, Sebastian Krinninger, Danupon Nanongkai, and Thatchaphol Saranurak. Unifying and strengthening hardness for dynamic problems via the online matrix-vector multiplication conjecture. In Rocco A. Servedio and Ronitt Rubinfeld, editors, *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 21–30. ACM, 2015. doi:10.1145/2746539.2746609.
- 35 Monika Henzinger, Andrea Lincoln, Stefan Neumann, and Virginia Vassilevska Williams. Conditional hardness for sensitivity problems. In Christos H. Papadimitriou, editor, *8th Innovations in Theoretical Computer Science Conference, ITCS 2017, January 9-11, 2017, Berkeley, CA, USA*, volume 67 of *LIPIcs*, pages 26:1–26:31. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017. doi:10.4230/LIPIcs.ITCS.2017.26.
- 36 Ravi Janardan and Mario Alberto López. Generalized intersection searching problems. *Int. J. Comput. Geom. Appl.*, 3(1):39–69, 1993. doi:10.1142/S021819599300004X.
- 37 Ce Jin and Yinzhan Xu. Tight dynamic problem lower bounds from generalized BMM and OMv. *To appear in STOC'22*. *CoRR*, abs/2202.11250, 2022. arXiv:2202.11250.
- 38 Adam Karczmarz and Jakub Lacki. Fast and simple connectivity in graph timelines. In Frank Dehne, Jörg-Rüdiger Sack, and Ulrike Stege, editors, *Algorithms and Data Structures - 14th International Symposium, WADS 2015, Victoria, BC, Canada, August 5-7, 2015. Proceedings*, volume 9214 of *Lecture Notes in Computer Science*, pages 458–469. Springer, 2015. doi:10.1007/978-3-319-21840-3_38.
- 39 Young Kun Ko and Min Jae Song. Hardness of approximate nearest neighbor search under l -infinity. *CoRR*, abs/2011.06135, 2020. arXiv:2011.06135.
- 40 Tsvi Kopelowitz and Robert Krauthgamer. Color-distance oracles and snippets. In Roberto Grossi and Moshe Lewenstein, editors, *27th Annual Symposium on Combinatorial Pattern Matching, CPM 2016, June 27-29, 2016, Tel Aviv, Israel*, volume 54 of *LIPIcs*, pages 24:1–24:10. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016. doi:10.4230/LIPIcs.CPM.2016.24.

- 41 Tsvi Kopelowitz, Seth Pettie, and Ely Porat. Higher lower bounds from the 3SUM conjecture. In Robert Krauthgamer, editor, *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 1272–1287. SIAM, 2016. doi:10.1137/1.9781611974331.ch89.
- 42 Kasper Green Larsen and Freek van Walderveen. Near-optimal range reporting structures for categorical data. In Sanjeev Khanna, editor, *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2013, New Orleans, Louisiana, USA, January 6-8, 2013*, pages 265–276. SIAM, 2013. doi:10.1137/1.9781611973105.20.
- 43 Kasper Green Larsen and R. Ryan Williams. Faster online matrix-vector multiplication. In Philip N. Klein, editor, *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 2182–2189. SIAM, 2017. doi:10.1137/1.9781611974782.142.
- 44 Joshua Lau and Angus Ritossa. Algorithms and hardness for multidimensional range updates and queries. In James R. Lee, editor, *12th Innovations in Theoretical Computer Science Conference, ITCS 2021, January 6-8, 2021, Virtual Conference*, volume 185 of *LIPICs*, pages 35:1–35:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPICs.ITCS.2021.35.
- 45 Christian W. Mortensen. Generalized static orthogonal range searching in less space. Technical Report TR-2003-33, IT University of Copenhagen, Copenhagen, Denmark, September 2003.
- 46 Yakov Nekrich. Efficient range searching for categorical and plain data. *ACM Trans. Database Syst.*, 39(1):9:1–9:21, 2014. doi:10.1145/2543924.
- 47 Mark H. Overmars. Searching in the past II: General transforms. Technical Report RUU-CS-81-9, Department of Computer Science, University of Utrecht, Utrecht, The Netherlands, May 1981.
- 48 Mark H. Overmars and Jan van Leeuwen. Maintenance of configurations in the plane. *J. Comput. Syst. Sci.*, 23(2):166–204, 1981. doi:10.1016/0022-0000(81)90012-X.
- 49 Maximilian Probst. On the complexity of the (approximate) nearest colored node problem. In Yossi Azar, Hannah Bast, and Grzegorz Herman, editors, *26th Annual European Symposium on Algorithms, ESA 2018, August 20-22, 2018, Helsinki, Finland*, volume 112 of *LIPICs*, pages 68:1–68:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPICs.ESA.2018.68.
- 50 Mihai Pătraşcu. Towards polynomial lower bounds for dynamic problems. In *Proceedings of the Forty-Second ACM Symposium on Theory of Computing, STOC '10*, pages 603–610, New York, NY, USA, 2010. Association for Computing Machinery. doi:10.1145/1806689.1806772.
- 51 Aviad Rubinfeld. Hardness of approximate nearest neighbor search. In Ilias Diakonikolas, David Kempe, and Monika Henzinger, editors, *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. ACM, 2018. doi:10.1145/3188745.3188916.
- 52 Qingmin Shi and Joseph F. JáJá. Optimal and near-optimal algorithms for generalized intersection reporting on pointer machines. *Inf. Process. Lett.*, 95(3):382–388, 2005. doi:10.1016/j.ipl.2005.04.008.
- 53 Jan van den Brand, Danupon Nanongkai, and Thatchaphol Saranurak. Dynamic matrix inverse: Improved algorithms and matching conditional lower bounds. In David Zuckerman, editor, *60th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2019, Baltimore, Maryland, USA, November 9-12, 2019*, pages 456–480. IEEE Computer Society, 2019. doi:10.1109/FOCS.2019.00036.
- 54 R. Ryan Williams. Faster all-pairs shortest paths via circuit complexity. *SIAM J. Comput.*, 47(5):1965–1985, 2018. doi:10.1137/15M1024524.
- 55 Virginia Vassilevska Williams. *On some fine-grained questions in algorithms and complexity*, pages 3447–3487. WORLD SCIENTIFIC, 2019. doi:10.1142/9789813272880_0188.

- 56 Virginia Vassilevska Williams and Yinzhan Xu. Monochromatic triangles, triangle listing and APSP. In Sandy Irani, editor, *61st IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020, Durham, NC, USA, November 16-19, 2020*, pages 786–797. IEEE, 2020. doi:10.1109/FOCS46700.2020.00078.
- 57 Hakan Yıldız, Luca Foschini, John Hershberger, and Subhash Suri. The union of probabilistic boxes: Maintaining the volume. In Camil Demetrescu and Magnús M. Halldórsson, editors, *Algorithms - ESA 2011 - 19th Annual European Symposium, Saarbrücken, Germany, September 5-9, 2011. Proceedings*, volume 6942 of *Lecture Notes in Computer Science*, pages 591–602. Springer, 2011. doi:10.1007/978-3-642-23719-5_50.
- 58 Hakan Yıldız, John Hershberger, and Subhash Suri. A discrete and dynamic version of Klee’s measure problem. In *Proceedings of the 23rd Annual Canadian Conference on Computational Geometry, Toronto, Ontario, Canada, August 10-12, 2011*, 2011. URL: <http://www.cccg.ca/proceedings/2011/papers/paper28.pdf>.