Computational Geometry: Young Researchers Forum 2025

— book of abstracts —

This volume contains the abstracts of presentations given at “Computational Geometry: Young Researchers Forum” (CG:YRF), a satellite event of the 41st International Symposium on Computational Geometry, held in Kanazawa, Japan, on June 23-27, 2025.

The CG:YRF program committee consisted of the following people:

- Eyal Ackerman, University of Haifa at Oranim
- Sayan Bandyapadhaya, Portland State University
- Chaya Keller, Ariel University
- Benjamin Raichel, University of Texas at Dallas
- Salman Parsa, DePaul University,
- Eunjin Oh (chair), POSTECH
- Frank Staals, Utrecht University
- Lori Ziegelmeier, Macalester College

There were 24 abstracts submitted to CG:YRF. Of these, 22 were accepted with revisions after a limited refereeing process. Two abstracts were withdrawn. The abstracts have been made public for the benefit of the community and should be considered preprints rather than formally reviewed papers. Thus, these works are expected to appear in conferences with formal proceedings and/or in journals.

More information about the event and about previous and future editions is available online at http://www.computational-geometry.org

CG:Young Researchers Forum — Program

Tuesday, 24 June 2024, 09:40–10:00, CG:YRF Fast Forward. Chair: Eunjin Oh

Tuesday, 24 June 2024, 16:00–17:45, CG:YRF Session A. Chair: Frank Staals

Tuesday, 24 June 2024, 16:00–17:45, CG:YRF Session B. Chair: Eunjin Oh

Tuesday, 24 June 2024, 16:00–17:20, CG:YRF Session C. Chair: Woojin Kim

Property Testing of Curve Similarity

Peyman Afshani
Department of Computer Science, Aarhus University, Denmark

Maike Buchin
Faculty of Computer Science, Ruhr University Bochum, Germany

Anne Driemel
Institute for Computer Science, University of Bonn, Germany

Marena Richter
Institute for Computer Science, University of Bonn, Germany

Sampson Wong
Department of Computer Science, University of Copenhagen, Denmark

Abstract

We propose a probabilistic testing algorithm that determines with constant probability if two curves are similar w.r.t. the discrete Fréchet distance or if they are 'ε-far' (for $0 < \varepsilon < 2$) from being similar, i.e., more than an ε-fraction of the two curves must be ignored for them to become similar. The algorithm performs $O(\frac{t}{\varepsilon} \log \frac{t}{\varepsilon})$ queries where a query returns the set of vertices of the curve that lie within a radius δ of a specified vertex of the other curve and t corresponds to a property of the two curves. We present a class of curves for which t is suffiently small so that the algorithm is sublinear.

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

Keywords and phrases Fréchet distance, Trajectory Analysis, Curve Similarity, Property Testing, Monotonicity Testing

Related Version arXiv:2502.17277

Funding *Anne Driemel*: Affiliated with Lamarr Institute for Machine Learning and Artificial Intelligence.

Acknowledgements Anne Driemel would like to thank Ronitt Rubinfeld and Ilan Newman for helpful discussions in the early stages of this research. We thank Sariel Har-Peled for feedback on an earlier version of this manuscript.

1 Introduction

We initiate a study of property testing for the discrete Fréchet distance. Typically in property testing, we are given access to a (large) data set and the goal is to very quickly assess whether the data has a certain property. Instead of the classical notation of correctness, a property testing algorithm is considered correct if it can satisfy the following two conditions, with a probability close to 1: First, if the input has the desired property, the algorithm must return *accept* and second, if the input is 'far' from having the property, the algorithm should *reject* the input. For more details on property testing, see [13, 1, 7]. Computational geometry has a long tradition of using randomization and sampling to speed up algorithmic approaches [12, 9, 10, 8]. Property testing has received some attention within computational geometry [6, 3, 4, 5, 11, 2], but is much less explored compared to other areas.

2 Preliminaries and problem definition

Let (M, d) be a metric space. We say a *curve* P in (M, d) is an ordered point sequence $\langle p_1, \ldots, p_n \rangle$ with *vertices* $p_i \in M$ for all $i = 1, \ldots, n$. We define the *length* of P to be

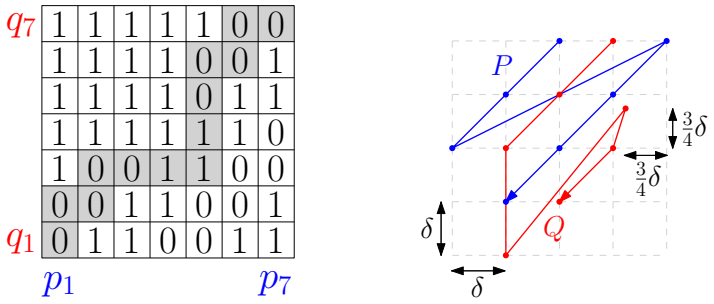


Figure 1 Left: M_δ for the curves on the right with a minimum cost monotone Manhattan path.

$\ell(P) = \sum_{i=1}^{n-1} d(p_i, p_{i+1})$. The *subcurve* of P between p_i and p_j is denoted by $P[i, j]$. A curve P is called *t-straight* if for any two vertices p_i and p_j in P, we have $\ell(P[i, j]) \leq t \cdot d(p_i, p_j)$. Given two curves $P = \langle p_1, \ldots, p_n \rangle$ and $Q = \langle q_1, \ldots, q_n \rangle$, we say that an ordered sequence $\mathcal{C}$ of elements in the n times n integer lattice $[n] \times [n]$ is a *coupling* of P and Q, if it starts in $(1, 1)$, ends in (n, n) and for any consecutive tuples $(i, j), (i', j')$ in $\mathcal{C}$ it holds that $(i', j') \in \{(i + 1, j), (i, j + 1)\}$. We define the *discrete Fréchet distance*[1] between P and Q as

$$D_{\mathcal{F}}(P, Q) := \min_{\text{coupling } \mathcal{C}} \max_{(i,j)\in\mathcal{C}} d(p_i, q_j).$$

The *free space matrix* of P and Q with distance value δ is an $n \times n$ matrix M_δ, where the i-th column corresponds to the vertex p_i of P and the j-th row corresponds to the vertex q_j of Q. The entry $M_\delta[i, j]$ has the value 0 if $d(p_i, q_j) \leq \delta$ and 1 otherwise.[2] A *monotone Manhattan path* $\mathcal{C}$ is a path through the free space matrix that always moves one step up or one step to the right. We define the *cost* of such a path as $c(\mathcal{C}) = \sum_{(i,j)\in\mathcal{C}} M_\delta[i, j]$. Note that $D_{\mathcal{F}}(P, Q) \leq \delta$ if and only if there exists a monotone Manhattan path with cost 0 from $(1, 1)$ to (n, n). Our analysis is based on a property of the free space matrix. We first define this property and then link the property to a certain class of well-behaved input curves.

▶ **Definition 1** (t-local). *Let M be a free space matrix of curves P and Q. We say that M is* t-local *if, for any tuples (i_1, j_1) and (i_2, j_2) with $M[i_1, j_1] = 0 = M[i_2, j_2]$, it holds that $|i_1 - i_2| \leq t \cdot (2 + |j_1 - j_2|)$ and $|j_1 - j_2| \leq t \cdot (2 + |i_1 - i_2|)$.*

▶ **Lemma 2.** *Let P and Q be t-straight curves with edge lengths in $[\delta/\alpha, \alpha\delta]$ for some constant $\alpha \geq 1$. Then, M_δ is $\mathcal{O}(t)$-local.*

For a proof, we refer to the full version. In the full version, we show that our approach also works if the lengths of the edges are bounded by a constant multiple of any fixed value.

▶ **Definition 3** (query). *We have access to the free space matrix via an oracle that returns a sorted list of indices of all zero-entries in the queried row or column. We call this a query.*

▶ **Definition 4** ((ε, δ)-far). *Given two curves P and Q consisting of n vertices each*[3]*, we say that P and Q are (ε, δ)-far from each other if there exists no monotone Manhattan path from $(1, 1)$ to (n, n) in the δ-free space matrix of cost εn or less.*

[1] The classical definition of the discrete Fréchet distance allows diagonal steps in the coupling. An easy adaptation of our proofs to the definition with diagonal steps can be found in Appendix **??**.
[2] Note we use 0 and 1 in switched roles compared to the conventions in the literature.
[3] For ease of notation, our analysis assumes the input curves have the same number of vertices.

▶ **Definition 5** (Fréchet-tester). *Assume we are given query-access to two curves P and Q, and we are given values $\delta > 0$ and $0 < \varepsilon < 2$. If the two curves have discrete Fréchet distance at most δ, we must return 'yes', and if they are (ε, δ)-far from each other w.r.t. the discrete Fréchet distance, the algorithm must return 'no', with probability at least $\frac{4}{5}$.*

Our goal is to design a Fréchet-tester that performs as few (sublinear in n) queries as possible.

3 Testing the discrete Fréchet distance

The idea of Algorithm 1 is to sample a number of columns and rows and check whether there is locally a monotone Manhattan path of cost zero possible. For proofs, see the full version.

▶ **Definition 6** (Permeability). *We say a block $[i, i']$ of consecutive columns (resp., rows) from index i to index i' is* permeable *if there exists a monotone Manhattan path of cost zero that starts in column (resp., row) i and ends in column (resp., row) i'.*

If a column or row contains only one-entries, we call it a *barrier-column* or *barrier-row*.

Algorithm 1 Fréchet-Tester1(M, t, ε)

If $M[1,1] = 1$ or $M[n,n] = 1$ **then return** 'no'.
repeat $\lceil \frac{24t}{\varepsilon} \rceil$ **times**:
  $j \leftarrow$ sample an index uniformly at random from $[n]$.
  if row j or column j of M is a barrier-column or barrier-row **then return** 'no'.
$K \leftarrow \lceil \frac{\varepsilon n}{32t} \rceil - 1$, $\ell \leftarrow \lceil \frac{128t}{\varepsilon} \rceil$, let $\mathcal{J}$ be a set of intervals and set $\mathcal{J} \leftarrow \emptyset$.
for $i = 0, \ldots, \lfloor \log_2 \ell \rfloor$ **do**:
  $I \leftarrow$ sample $\lceil \frac{16n}{2^{i+1}K} \rceil$ different indices uniformly at random from $\{0, 1, \ldots, \frac{n}{2^{i+1}} - 2\}$.
  for each $j \in I$ **do**: add $[j2^{i+1}, (j+2)2^{i+1}]$ to $\mathcal{J}$.
foreach $[i, j] \in \mathcal{J}$ **do**
  if block $[i, j]$ of consecutive columns is not permeable **then return** 'no'.
  if block $[i, j]$ of consecutive rows is not permeable **then return** 'no'.
return 'yes'.

▶ **Theorem 7.** *Let P and Q be curves with n vertices such that their free space matrix is t-local and t is known. Then, Algorithm 1 is a Fréchet-tester that needs $\mathcal{O}(\frac{t}{\varepsilon} \log \frac{t}{\varepsilon})$ queries.*

References

1 Arnab Bhattacharyya and Yuichi Yoshida. *Property Testing: Problems and Techniques*. Springer Nature, 2022.
2 Bernard Chazelle, Ding Liu, and Avner Magen. Sublinear geometric algorithms. In *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, pages 531–540, 2003.
3 Bernard Chazelle, Ronitt Rubinfeld, and Luca Trevisan. Approximating the minimum spanning tree weight in sublinear time. *SIAM Journal on Computing*, 34(6):1370–1379, 2005. doi:10.1137/S0097539702403244.
4 Artur Czumaj, Funda Ergün, Lance Fortnow, Avner Magen, Ilan Newman, Ronitt Rubinfeld, and Christian Sohler. Approximating the weight of the Euclidean minimum spanning tree in sublinear time. *SIAM Journal on Computing*, 35(1):91–109, 2005. doi:10.1137/S0097539703435297.
5 Artur Czumaj and Christian Sohler. Estimating the weight of metric minimum spanning trees in sublinear time. *SIAM Journal on Computing*, 39(3):904–922, 2009. doi:10.1137/060672121.

6 Artur Czumaj, Christian Sohler, and Martin Ziegler. Property testing in computational geometry. In *European Symposium on Algorithms*, pages 155–166. Springer, 2000.

7 Oded Goldreich. *Introduction to property testing*. Cambridge University Press, 2017.

8 J.E. Goodman, J. O'Rourke, and C.D. Tóth. *Handbook of discrete and computational geometry, third edition*. 01 2017. doi:10.1201/9781315119601.

9 Sariel Har-Peled. *Geometric Approximation Algorithms*. American Mathematical Society, USA, 2011.

10 Jirí Matousek. *Lectures on discrete geometry*, volume 212 of *Graduate texts in mathematics*. Springer, 2002.

11 Morteza Monemizadeh. Facility Location in the Sublinear Geometric Model. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2023)*, volume 275 of *LIPIcs*, pages 6:1–6:24, 2023. doi:10.4230/LIPIcs.APPROX/RANDOM.2023.6.

12 Ketan. Mulmuley. *Computational geometry : an introduction through randomized algorithms*. Prentice-Hall, Englewood Cliffs, N.J, 1994.

13 Dana Ron. Algorithmic and analysis techniques in property testing. *Foundations and Trends® in Theoretical Computer Science*, 5(2):73–205, 2010.

Separation Anxiety: A Well-Separated Pair Decomposition and Separator for c-packed Graphs

Lindsey Deryckere
School of Computer Science, The University of Sydney, Australia

Joachim Gudmundsson
School of Computer Science, The University of Sydney, Australia

André van Renssen
School of Computer Science, The University of Sydney, Australia

Yuan Sha
School of Computer Science, The University of Sydney, Australia

Sampson Wong
Department of Computer Science, The University of Copenhagen, Denmark

Abstract

We prove two fundamental properties of c-packed graphs: that there exists a linear-size well-separated pair decomposition under the graph metric, and there exists a constant size balanced separator. We apply these properties to obtain a tree cover of constant size, an exact distance oracle of near-linear size and an approximate distance oracle of linear size.

2012 ACM Subject Classification Theory of computation → Computational geometry

Keywords and phrases Well-separated pair decomposition, separator, tree cover, distance oracles, realistic graphs

Related Version A full version of the paper is available at: `https://arxiv.org/abs/2505.06884`

Funding This research was partially funded by the Australian Government through the Australian Research Council (project number DP240101353).

1 Introduction

The c-packedness property [6], proposed in 2010, is a geometric property that captures the spatial distribution of the edges in a graph. A graph is c-packed if, for any radius r and any ball of radius r, the length of the edges contained in the ball is at most $c \cdot r$. Driemel, Har-Peled and Wenk [6] introduced the c-packedness property for polygonal curves, and showed that one can compute the Fréchet distance between a pair of c-packed curves in near-linear time. In 2013, Gudmundsson and Smid [11] adapted the c-packedness definition to graphs. So far the study of c-packed graphs has been limited to Frechet distance problems [3, 9, 10]. An open problem is whether they have applications beyond Fréchet distance problems.

We provide the first deterministic construction of a linear-size WSPD and $O(c)$-size balanced separator that are independent of the spread of the c-packed metric. We use the separator and WSPD to obtain a tree cover of constant size, an exact distance oracle of near-linear size and an approximate distance oracle of linear size. Our deterministic construction of the WSPD is, to the best of our knowledge, the first that does not depend on the aspect ratio of the metric space.

1.1 Related Work

Well-Separated Pair Decompositions (WSPD) are used for compact representation of the quadratic distances between pairs of points in a metric. For metrics that allow for a sub-

quadtratic size WSPD, they have therefore been used as fundamental tools to approximate solutions to a range of proximity problems that require looking at the distances between all pairs of points, such as nearest neighbour, diameter, stretch and minimum spanning tree. For a point set in $\mathbb{R}^d$, where d is considered a constant, Callahan and Kosaraju [4] showed that there exists a WSPD with separation factor σ of size $O(\sigma^d n)$ that can be computed in $O(n \log n + \sigma^d n)$ time. In contrast to this, we show that for c-packed graphs, the size of the WSPD is not exponential in d, while maintaining that the size is linear.

Balanced separators of sublinear size have been found for a variety of graphs [1, 8, 14, 16]. They have been used as a fundamental tool in devising efficient algorithms for graphs [1, 8, 15] and in numerical analysis [13, 16]. We show that c-packed graphs admit an $O(c)$-sized balanced separator.

Dvořák and Norin [7] showed that if a graph admits a small size balanced separator, it also has small treewidth. Combined with our separator results this implies that c-packed graphs have treewidth $O(c)$. Chaudhuri and Zaroliagis [5] designed an exact distance oracle whose preprocessing time is single exponential in the treewidth of the graph. In contrast to these results, our algorithms do not incur any terms exponential in c.

2 A Well-Separated Pair Decomposition for c-packed Graphs

We construct a tree that fulfills a similar purpose to split trees but for graph distances (which are metric) between points. We call this new type of tree a *δ-connected tree* (δ-CT). Each cell, corresponding to a cube s, of the δ-connected tree is a δ-connected set, meaning that points contained in the cell are within a graph distance of at most $\delta \cdot diam(s)$ from one another. To construct the δ-CT, we use a bottom up approach. The leaves of the compressed quadtree are already δ-connected sets. At higher levels of the compressed quadtree, we consider the δ-connected sets of its children, and merge together pairs of previously δ-connected sets that are also a δ-connected set in the higher level. To obtain an efficient running time, we make two observations. First, when computing the δ-connected sets for a higher level, it suffices to maintain a vertex representative for each δ-connected set of the lower level. Second, to check if a pair of sets are δ-connected, it suffices to check whether their representatives are path-connected in the cube centered at the cell but with double its radius.

To upper bound the graph diameter of the δ-connected set in each cell of the δ-CT we compute the length of intersection of edges with the cell and the 3^d surrounding cells in a canonical grid. To do this efficiently, we construct a data structure that can be queried for the total length of all edges that can contribute to a δ-connected component contained in a cell. We obtain the following theorem.

▶ **Theorem 1.** *Given a c-packed graph G in $\mathbb{R}^d$, for fixed d, one can construct a $WSPD_G$ with separation factor σ of size $O(c^3\sigma n)$ in $O(cn \log n + c^3\sigma n)$ time, using $O(cn)$ space.*

3 A Separator Theorem for c-packed Graphs

We start with the ring separator of Har-Peled and Mendel [12], which states that for a point set in $\mathbb{R}^d$, one can efficiently compute a pair of balls so that $n/2\lambda^3$ of the points are inside the inner ball, and $n/2\lambda^3$ of the points are outside the outer ball, where λ is the doubling constant of $\mathbb{R}^d$. Using the ring separator, we construct a max-flow instance in a similar fashion to Gudmundsson et al. [10] to locate a cut of size $O(c)$. This cut $(1 - 1/2\lambda^3)$-separates the graph, in that it separates the graph into two components each with at most $n \cdot (1 - 1/2\lambda^3)$ points. We obtain the following theorem.

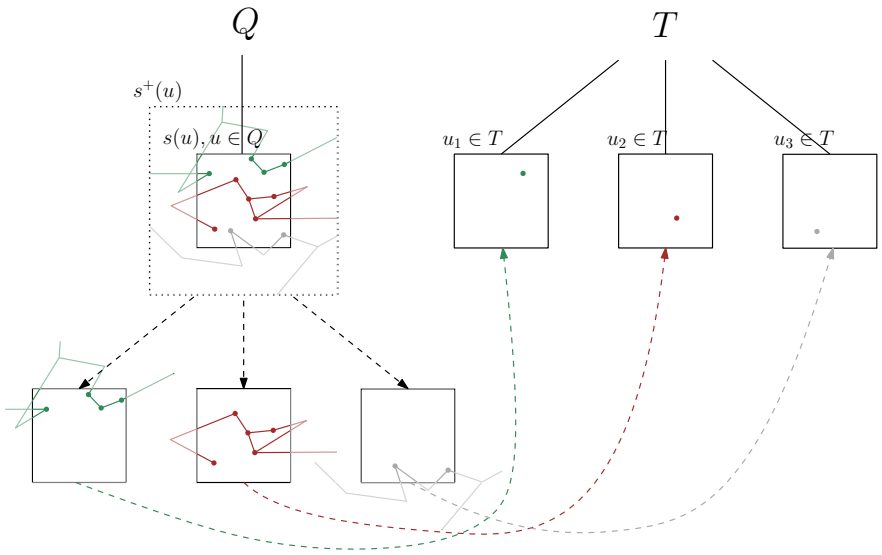


Figure 1 An illustration of iteration j of the algorithm constructing the c-connected tree.

▶ **Theorem 2.** *Given a c-packed graph G in $\mathbb{R}^d$, where d is fixed, with n vertices, one can find a separator of size $O(c)$ that $(1 - \frac{1}{2\lambda^3})$-separates G, in $O(c^2 n)$ time.*

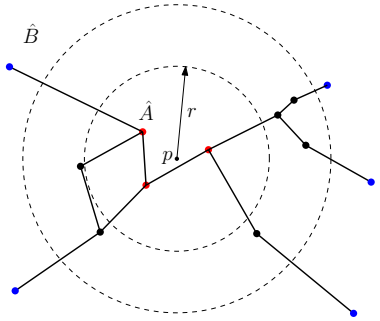


Figure 2 An example of a max-flow instance to locate a cut of size $O(c)$ where the vertices in $\hat{A}$ are the sources and $\hat{B}$ the sinks. The value of the min-cut in the figure is 4.

4 Distance Oracles and a Small Tree Cover for c-packed Graphs

We combine our separator with standard techniques [15] to construct an exact distance oracle. We use this to construct a tree cover following the approach of the celebrated "Dumbbell Theorem" [2]. The main difficulty lies in proving the packing lemmas required for establishing the empty-region property. A dumbbell tree, which connects the dumbbells in a group hierarchically, is built for each group of dumbbells. The c-packedness property and the c-CT enable us to do range searching and efficiently build the dumbbell trees. The tree cover immediately implies a $(1 + \varepsilon)$-approximate distance oracle for the c-packed metric.

References

1 Noga Alon, Paul D. Seymour, and Robin Thomas. A separator theorem for graphs with an excluded minor and its applications. In *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing (STOC)*, pages 293–299. ACM, 1990.

2 Sunil Arya, Gautam Das, David M. Mount, Jeffrey S. Salowe, and Michiel H. M. Smid. Euclidean spanners: short, thin, and lanky. In *Proceedings of the 27th Annual ACM Symposium on Theory of Computing (STOC)*, pages 489–498. ACM, 1995.

3 Karl Bringmann and Marvin Künnemann. Improved approximation for fréchet distance on c-packed curves matching conditional lower bounds. *Int. J. Comput. Geom. Appl.*, 27(1-2):85–120, 2017. doi:10.1142/S0218195917600056.

4 Paul B. Callahan and S. Rao Kosaraju. A decomposition of multidimensional point sets with applications to k-nearest-neighbors and n-body potential fields. *J. ACM*, 42(1):67–90, 1995. doi:10.1145/200836.200853.

5 Shiva Chaudhuri and Christos D. Zaroliagis. Shortest paths in digraphs of small treewidth. part I: sequential algorithms. *Algorithmica*, 27(3):212–226, 2000.

6 Anne Driemel, Sariel Har-Peled, and Carola Wenk. Approximating the Fréchet distance for realistic curves in near linear time. *Discret. Comput. Geom.*, 48(1):94–127, 2012. URL: https://doi.org/10.1007/s00454-012-9402-z, doi:10.1007/S00454-012-9402-Z.

7 Zdenek Dvorák and Sergey Norin. Treewidth of graphs with balanced separations. *J. Comb. Theory B*, 137:137–144, 2019. URL: https://doi.org/10.1016/j.jctb.2018.12.007, doi:10.1016/J.JCTB.2018.12.007.

8 John R. Gilbert, Joan P. Hutchinson, and Robert Endre Tarjan. A separator theorem for graphs of bounded genus. *J. Algorithms*, 5(3):391–407, 1984.

9 Joachim Gudmundsson, Tiancheng Mai, and Sampson Wong. Approximating the fréchet distance when only one curve is c-packed. In Julián Mestre and Anthony Wirth, editors, *35th International Symposium on Algorithms and Computation, ISAAC 2024, December 8-11, 2024, Sydney, Australia*, volume 322 of *LIPIcs*, pages 37:1–37:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024. URL: https://doi.org/10.4230/LIPIcs.ISAAC.2024.37, doi:10.4230/LIPICS.ISAAC.2024.37.

10 Joachim Gudmundsson, Martin P. Seybold, and Sampson Wong. Map matching queries on realistic input graphs under the Fréchet distance. In *Proceedings of the 34th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1464–1492. SIAM, 2023. URL: https://doi.org/10.1137/1.9781611977554.ch53, doi:10.1137/1.9781611977554.CH53.

11 Joachim Gudmundsson and Michiel H. M. Smid. Fast algorithms for approximate Fréchet matching queries in geometric trees. *Comput. Geom.*, 48(6):479–494, 2015. URL: https://doi.org/10.1016/j.comgeo.2015.02.003, doi:10.1016/J.COMGEO.2015.02.003.

12 Sariel Har-Peled and Manor Mendel. Fast construction of nets in low-dimensional metrics and their applications. *SIAM J. Comput.*, 35(5):1148–1184, 2006. doi:10.1137/S0097539704446281.

13 Rasmus Kyng, Richard Peng, Robert Schwieterman, and Peng Zhang. Incomplete nested dissection. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, pages 404–417. ACM, 2018.

14 R. J. Lipton and R. E. Tarjan. A separator theorem for planar graphs. *SIAM J. Appl. Math*, 36(2):177–189, 1979.

15 Richard J. Lipton and Robert E. Tarjan. Applications of a planar separator theorem. *SIAM J. Comput.*, 9(3):615–627, 1980.

16 Gary L. Miller and William P. Thurston. Separators in two and three dimensions. In *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing (STOC)*, pages 300–309. ACM, 1990.

Smallest Intersecting and Enclosing Balls

Jiaqi Zheng
Department of Computer Science, National University of Singapore

Tiow-Seng Tan
Department of Computer Science, National University of Singapore

Abstract

We study the smallest intersecting and enclosing ball problems in Euclidean spaces for input objects that are compact and convex. They link and unify many problems in computational geometry and machine learning. We show that both problems can be modeled as zero-sum games, and propose an approximation algorithm for the former. Specifically, the algorithm produces the first results in high-dimensional spaces for various input objects such as convex polytopes, balls, ellipsoids, etc.

2012 ACM Subject Classification Theory of computation → Computational geometry; Theory of computation → Approximation algorithms analysis; Theory of computation → Convex optimization

Keywords and phrases Geometric optimization, smallest intersecting ball, approximation algorithm

1 Introduction

Given n convex compact objects $\Omega_1, \ldots, \Omega_n$ in d-dimensional Euclidean space, the *smallest intersecting ball* (SIB) problem is to find a ball with the smallest radius r^* that *intersects* every Ω_i, while the *smallest enclosing ball* (SEB) problem is to find the ball with the smallest radius R^* that *encloses* every Ω_i. See Figure 1 for 2D examples of these two problems.

The SEB problem has attracted significant attention in the past decades [3, 16, 6], whereas the SIB problem is less discussed and the understanding of SIB lags behind that of SEB. In earlier research [2, 12], SIB are usually considered a variant of SEB. Indeed, they are identical when the input are singleton sets. Nevertheless, as the complexity of the input structure increases, the divergence between these two problems becomes more evident and the SIB problem manifests greater versatility. This is demonstrable even when there are only two objects, Ω_1 and Ω_2: when Ω_1 is a compact convex set and Ω_2 is a single point, the SIB problem is equivalent to finding the nearest point (Euclidean projection) of Ω_2 in the region of Ω_1, and r^* is half the distance from Ω_1 to Ω_2; when Ω_1 and Ω_2 are both convex compact sets, the SIB problem becomes finding the shortest line segment (a.k.a. the shortest connector) that connects these two sets, and r^* is half the minimum distance between them. The dual problem of minimum connector is to find the hyperplane that separates Ω_1 and Ω_2 with the largest margin [7], which corresponds to the support vector machine problems in machine learning [1, 8]. See Figure 2 for examples of SIB in different cases.

Given the diversity of the SIB problem, one can reasonably anticipate that it poses more substantial computational challenges than SEB. Indeed, numerous algorithms have been proposed for solving the SEB problem, including exact and approximation algorithms [15, 13], using optimization or coreset techniques [9, 3], and in parallel or streaming settings [6, 5], but for SIB, most algorithms are merely designed for solving it in fixed dimensions [2, 10].

In this work, we endeavor to narrow the gap in the understanding of these two problems. We show that both the SIB and SEB problems can be modeled as two-player zero-sum games, which is inspired by the seminal work of Clarkson et. al. [6] in sublinear optimization. Based on the new formulation, we propose the first approximation algorithm for the SIB problem in arbitrary dimensions in the unit-cost RAM model, which leverages recent advances in symmetric cone problems [4, 19]. Additional details on the SIB algorithm can be found in the full-length preprint [18]. Software implementing the algorithm is available at [17].

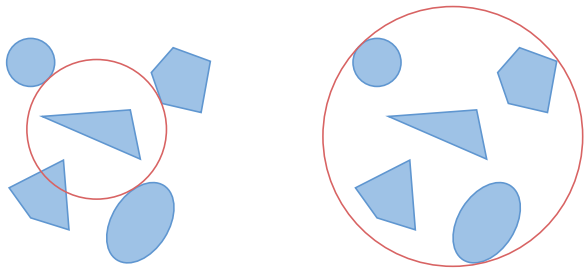


Figure 1 Examples of the problems in 2D spaces, where the blue objects are the input and red circles are the solutions. *Left:* the smallest intersecting ball. *Right:* the smallest enclosing ball.

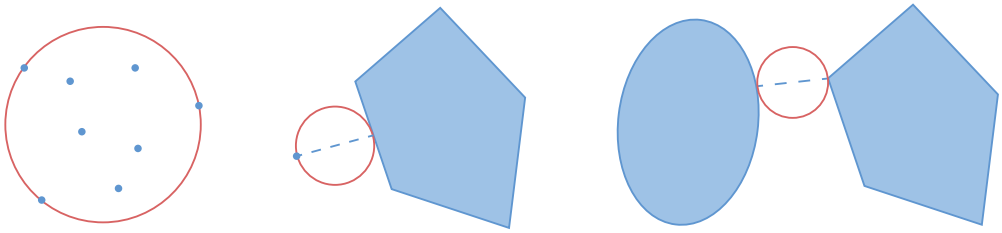


Figure 2 Many faces of the SIB problem. *Left:* the SEB of a point set. *Middle:* the nearest point (Euclidean projection) in a convex set. *Right:* the shortest connector (minimum distance).

2 SIB and SEB as Zero-Sum Games

We use $\oplus$ to denote concatenations of vectors, or Cartesian products of sets and vector spaces. For instance, $\oplus_{i=1}^{n} \boldsymbol{u}_i$ denotes the concatenation of n vectors, namely $(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n)$. Consider the zero-sum game $\min_{\boldsymbol{p}\in\mathcal{P}} \max_{\boldsymbol{q}\in\mathcal{Q}} f(\boldsymbol{p}, \boldsymbol{q})$. We say $(\boldsymbol{p}^*, \boldsymbol{q}^*)$ is a *Nash equilibrium* iff $f(\boldsymbol{p}^*, \boldsymbol{q}^*) \le f(\boldsymbol{p}, \boldsymbol{q}^*), \forall \boldsymbol{p} \in \mathcal{P}$ and $f(\boldsymbol{p}^*, \boldsymbol{q}^*) \ge f(\boldsymbol{p}^*, \boldsymbol{q}), \forall \boldsymbol{q} \in \mathcal{Q}$. Moreover, $f(\boldsymbol{p}^*, \boldsymbol{q}^*)$ is the *value* of the game. Let $\mathcal{V} := \oplus_{i=1}^{n} \Omega_i$, $\mathcal{X}$ be the convex hull of the input, and $\mathcal{Y}$ defined as:

$$\mathcal{Y} := \left\{ \bigoplus_{i=1}^{n} (\boldsymbol{y}_i, s_i) \in \bigoplus_{i=1}^{n} \mathbb{R}^{d+1} : \|\boldsymbol{y}_i\| \le s_i, \forall i \in [n] \text{ and } \sum_{i=1}^{n} s_i = 1 \right\},$$

which can be viewed as the Cartesian product of n Euclidean balls whose radii sum to one.

▶ **Theorem 1.** *The SIB problem can be modeled as the following zero-sum game:*

$$\min_{(\boldsymbol{x},\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n)\in\mathcal{X}\times\mathcal{V}} \max_{\boldsymbol{y}\in\mathcal{Y}} \left(\bigoplus_{i=1}^{n} \begin{pmatrix} \boldsymbol{x} - \boldsymbol{v}_i \\ 0 \end{pmatrix} \right)^{\top} \boldsymbol{y}.$$

A Nash equilibrium (denoted as $(\boldsymbol{x}^*, \boldsymbol{v}_1^*, \ldots, \boldsymbol{v}_n^*, \boldsymbol{y}^*)$) of the game always exist, and the value of the game is r^*. The ball $B(\boldsymbol{x}^*, r^*)$ is intersecting every Ω_i, and $\boldsymbol{v}_i^* \in B(\boldsymbol{x}^*, r^*) \cap \Omega_i$.

▶ **Theorem 2.** *The SEB problem can be modeled as the following zero-sum game:*

$$\min_{\boldsymbol{x}\in\mathcal{X}} \max_{(\boldsymbol{y},\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n)\in\mathcal{Y}\times\mathcal{V}} \left(\bigoplus_{i=1}^{n} \begin{pmatrix} \boldsymbol{x} - \boldsymbol{v}_i \\ 0 \end{pmatrix} \right)^{\top} \boldsymbol{y}.$$

A Nash equilibrium (denoted as $(\boldsymbol{x}^*, \boldsymbol{y}^*, \boldsymbol{v}_1^*, \ldots, \boldsymbol{v}_n^*)$) of the game always exist, and the value of the game is R^*. The ball $B(\boldsymbol{x}^*, R^*)$ is enclosing every Ω_i.

Table 1 Summary of the results for the SIB problem

Input Objects	Previous Work	Our Result
Convex Polytopes	$O(M)^\dagger$ [10]	$O(\frac{R^2(N+nd)\log n}{\varepsilon^2})$
Axis-Aligned Bounding Boxes	$O(n)^\dagger$ [11]	$O(\frac{R^2nd\log n}{\varepsilon^2})$
Euclidean Balls	$O(\frac{n}{\varepsilon^{(d-1)/2}})$ [14]	$O(\frac{R^2nd\log n}{\varepsilon^2})$
Ellipsoids	-	$O(nd^\omega + \frac{R^2nd^2\log n}{\varepsilon^2})$

Note: d is the dimensionality. n is the number of objects. M is the total number of points. N is the number of nonzeros in the input. R is the ratio between D and r^*. ω is the matrix multiplication exponent.
† Running time of exact algorithms for problems in fixed dimensions.

It is worth noting that the SIB game is a bilinear zero-sum game, where the objective function is linear for both min- and max-player. On the other hand, the SEB game is not bilinear as the function is not linear (and neither convex nor concave) for the max-player.

3 Algorithms

Unlike the SEB problem that is extensively studied in the literature, most algorithms for the SIB problem are designed for fixed dimensions with limited types of input objects such as convex polytopes [10] and axis-aligned bounding boxes [11]. The only result for SIB in high-dimensional space is restricted to input of Euclidean balls that are pairwise disjoint [14].

Benefit from our new formulation for the SIB problem, we can utilize techniques for bilinear zero-sum games to design an approximation algorithm for general input objects in arbitrary dimensions. Specifically, we say $(\boldsymbol{x}, r)$ is an $(1+\varepsilon)$-approximate solution of the SIB problem if the ball $B(\boldsymbol{x}, r)$ intersects every Ω_i and $r \le (1+\varepsilon)r^*$. The algorithm works as follows: in each iteration, we update $\boldsymbol{y}$ using an online optimization algorithm over $\mathcal{Y}$, and let $(\boldsymbol{x}, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_n)$ be the best response in $\mathcal{X} \times \mathcal{V}$ against $\boldsymbol{y}$. Then it can be shown that the average point of the past iterates converges to an approximate Nash equilibrium of the SIB game, which provides an approximate solution of the SIB problem.

▶ **Theorem 3.** *Let D be the diameter of the input and let $R = \frac{D}{r^*}$. Suppose the best response can be computed in $O(S)$ time. Then there is an iterative algorithm that computes an $(1+\varepsilon)$-approximate solution of the SIB problem with running time $O(\frac{R^2(S+nd)\log n}{\varepsilon^2})$.*

The complexity results in the unit-cost RAM model for the SIB problem with specific input are shown in Table 1. See [18] for detailed analyses of our results. On the other hand, no existing algorithm can find Nash equilibria for the SEB game due to its non-bilinear nature. We hope for further advancement on the SEB problem under the new formulation.

References

1 Kristin P. Bennett and Erin J. Bredensteiner. Duality and geometry in SVM classifiers. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 57–64, 2000.

2 Binay K. Bhattacharya, Sreesh Jadhav, Asish Mukhopadhayay, and Jean-Marc Robert. Optimal algorithms for some smallest intersection radius problems. In *Proceedings of the Seventh Annual Symposium on Computational Geometry*, SCG '91, pages 81–88, New York, NY, USA, 1991.

3 Mihai Bădoiu and Kenneth L. Clarkson. Smaller core-sets for balls. In *SODA*, volume 3, pages 801–802, 2003.

4 Ilayda Canyakmaz, Wayne Lin, Georgios Piliouras, and Antonios Varvitsiotis. Multiplicative updates for online convex optimization over symmetric cones. *arXiv preprint arXiv:2307.03136*, 2023.

5 Timothy M Chan and Vinayak Pathak. Streaming and dynamic algorithms for minimum enclosing balls in high dimensions. *Computational Geometry*, 47(2):240–247, 2014.

6 Kenneth L. Clarkson, Elad Hazan, and David P. Woodruff. Sublinear optimization for machine learning. *J. ACM*, 59(5), 2012.

7 Achiya Dax. The distance between two convex sets. *Linear Algebra and its Applications*, 416(1):184–213, 2006.

8 Bernd Gärtner and Martin Jaggi. Coresets for polytope distance. In *Proceedings of the twenty-fifth annual symposium on Computational geometry*, pages 33–42, 2009.

9 Bernd Gärtner and Sven Schönherr. An efficient, exact, and generic quadratic programming solver for geometric optimization. In *Proceedings of the sixteenth annual symposium on Computational geometry*, pages 110–118, 2000.

10 Shreesh Jadhav, Asish Mukhopadhyay, and Binay Bhattacharya. An Optimal Algorithm for the Intersection Radius of a Set of Convex Polygons. *Journal of Algorithms*, 20(2):244–267, 1996.

11 Maarten Löffler and Marc van Kreveld. Largest bounding box, smallest diameter, and related problems on imprecise points. *Computational Geometry*, 43(4):419–433, 2010.

12 Boris Mordukhovich, Nguyen Mau Nam, and Cristina Villalobos. The smallest enclosing ball problem and the smallest intersecting ball problem: Existence and uniqueness of solutions. *Optimization Letters*, 7(5):839–853, Apr 2012.

13 Frank Nielsen and Richard Nock. Approximating smallest enclosing balls with applications to machine learning. *International Journal of Computational Geometry & Applications*, 19(05):389–414, 2009.

14 Wanbin Son and Peyman Afshani. Streaming Algorithms for Smallest Intersecting Ball of Disjoint Balls. In *Theory and Applications of Models of Computation*, pages 189–199, 2015.

15 Emo Welzl. Smallest enclosing disks (balls and ellipsoids). In *New Results and New Trends in Computer Science: Graz, Austria, June 20–21, 1991 Proceedings*, pages 359–370. Springer, 2005.

16 E Alper Yildirim. Two algorithms for the minimum enclosing ball problem. *SIAM Journal on Optimization*, 19(3):1368–1391, 2008.

17 Jiaqi Zheng. LIBSIB: A C++ library for computing smallest intersecting balls in arbitrary dimensions, 2024. URL: `https://github.com/orzzzjq/libsib`.

18 Jiaqi Zheng and Tiow-Seng Tan. Approximation algorithms for smallest intersecting balls. *arXiv preprint arXiv:2406.11369*, 2024.

19 Jiaqi Zheng, Antonios Varvitsiotis, Tiow-Seng Tan, and Wayne Lin. A primal-dual framework for symmetric cone programming. *arXiv preprint arXiv:2405.09157*, 2024.

Relax and Merge: A Simple Yet Effective Framework for Solving Fair k-Means and k-sparse Wasserstein Barycenter Problems

Shihong Song
University of Science and Technology of China, School of Computer Science and Technology, Hefei, Anhui 230026, China

Guanlin Mo[1]
University of Science and Technology of China, School of Computer Science and Technology, Hefei, Anhui 230026, China

Qingyuan Yang
University of Science and Technology of China, School of Computer Science and Technology, Hefei, Anhui 230026, China

Hu Ding[2]
University of Science and Technology of China, School of Computer Science and Technology, Hefei, Anhui 230026, China

Abstract

The fairness of clustering algorithms has gained widespread attention across various areas in machine learning. In this paper, we study *fair k-means clustering* in Euclidean space. Given a dataset comprising several groups, the fairness constraint requires that each cluster should contain a proportion of points from each group within specified lower and upper bounds. Due to these fairness constraints, determining the locations of k centers and finding the induced partition are quite challenging tasks. We propose a novel "Relax and Merge" framework that returns a $(1+4\rho+O(\epsilon))$-approximate solution, where ρ is the approximate ratio of an off-the-shelf vanilla k-means algorithm and $O(\epsilon)$ can be an arbitrarily small positive number. If equipped with a PTAS of k-means, our solution can achieve an approximation ratio of $(5+O(\epsilon))$ with only a slight violation of the fairness constraints, which improves the current state-of-the-art approximation guarantee. Furthermore, using our framework, we can also obtain a $(1+4\rho+O(\epsilon))$-approximate solution for the *k-sparse Wasserstein Barycenter* problem, which is a fundamental optimization problem in the field of optimal transport, and a $(2+6\rho)$-approximate solution for the *strictly fair k-means clustering* with no violation, both of which are better than the current state-of-the-art methods. In addition, the empirical results demonstrate that our proposed algorithm can significantly outperform baseline approaches in terms of clustering cost.

2012 ACM Subject Classification Theory of computation → Approximation algorithms analysis

Keywords and phrases clustering, k-means, fairness, approximate algorithm, optimal transport

Digital Object Identifier 10.4230/LIPIcs.CVIT.2016.23

Funding The research of this work was supported in part by the National Natural Science Foundation of China 62272432, the National Key Research and Development Program of China 2021YFA1000900, and the Natural Science Foundation of Anhui Province 2208085MF163.

Acknowledgements We are grateful to Prof. Lingxiao Huang in Nanjing University for inspiring discussions.

[1] The first two authors contribute equally.
[2] Corresponding author.

42nd Conference on Very Important Topics (CVIT 2016).
Editors: John Q. Open and Joan R. Access; Article No. 23; pp. 23:1–23:3
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

In this paper, we consider the problem of (α, β)-***fair*** k-***means clustering*** that was initially proposed by [4] and then generalized by [2].Informally speaking, we assume that the given dataset P consists of m groups of points, and the "fairness" constraint requires that in each obtained cluster, the points from each group should take a fraction between pre-specified lower and upper bounds. [2] showed that a ρ-approximate algorithm for vanilla k-means can provide a $(2+\sqrt{\rho})^2$- approximate solution for (α, β)-fair k-clustering with a slight violation on the fairness constraints. Here, "violation" refers to situations where the fairness constraints are not satisfied in the clustering solution. For example, if a fairness constraint requires each cluster to contain at least 40% of members from a certain group, then a cluster with only 30% would constitute some violations.

Furthermore, [3] studied the "strictly" fair k-means clustering problem, where it requires that the number of points from each group should be uniform in every cluster. Another problem closely related to fair k-means is the so-called "**k-sparse Wassertein Barycenter (WB)**" [1] . This is a fundamental concept in optimal transport theory, and it represents the "average" or central distribution of a set of probability distributions. The formal definitions are shown in appendix. The formal version of this paper has been published at ICLR 2025.

2 Our contributions

Our key idea relies on an important observation, where we find that the fair k-means problem is inherently related to a classic geometric structure, "ϵ-approximate centroid set", which was firstly proposed by [6]. Roughly speaking, given a dataset, an ϵ-approximate centroid set should contain at least one point that approximately represents the centroid location of any subset of this given dataset. It means that the ϵ-approximate centroid set contains not only the approximate centroids based on the Voronoi diagram, but also the approximate centroids of those potential fairness-preserving clusters.

Inspired by the above observation, we illustrate the relationship between fair k-means and ϵ-approximate centroid set first, and then propose a novel *Relax-and-Merge* framework. In this framework, we relax the constraints on the number of clusters k; we focus on utilizing fair constraints to cluster the data into small and fair clusters, which are then merged together to determine the positions of k cluster centers. As shown in Table 1, our result is better than the state of the art works [2, 3]. Equipped with a PTAS for k-means problem (e.g., the algorithm from [5]), our algorithm yields a $5 + O(\epsilon)$ approximation factor.

We also present two important extensions from our work. The first extension is an $(1 + 4\rho + O(\epsilon))$ solution for k-sparse Wasserstein Barycenter. The second one is about strictly fair k-means. We give a refined algorithm that yields a no-violation solution with a $(2 + 6\rho)$ approximation factor, which is better than the state of the art work [3].

In general, there are two stages in clustering with fair constraints. The first stage is to find the proper locations of clustering centers, and the second stage is to assign all the client points to the centers by solving an LP. The previous approaches often use the vanilla k-means in the first stage to obtain the location of centers, and then take the fairness into account in the second stage [2, 3]. In our proposed framework, we aim to shift the consideration of fair constraints to the first stage, so as to achieve a lower approximation factor in the final result. Our algorithm can be summarized in the following two steps:

Relax: We construct a relaxed solution T, *i.e.*, an ϵ-approximate centroid set, as the "potential" set of clustering centers. Here, we relax the size constraint of centers to be polynomial of n rather than exactly k, so as to achieve a sufficiently low cost. Then, we

Table 1 Comparison of the approximation ratios for fair k-means and k-sparse WB. The "general case" includes (α, β)-fair k-means, strictly (α, β)-fair k-means and k-sparse WB.

Algorithms	Approximation ratio	When $\rho = 1 + O(\epsilon)$	Note on the quality
Bera et al.,2019[2]	$(2 + \sqrt{\rho})^2$	$9 + O(\epsilon)$	general case
Schmidt et al.,2020[7]	$5.5\rho + 1$	$6.5 + O(\epsilon)$	two groups only
Böhm et al.,2021[3]	$(2 + \sqrt{\rho})^2$	$9 + O(\epsilon)$	strictly only, no violation
Yang & Ding,2024[8]	$(2 + \sqrt{\rho})^2$	$9 + O(\epsilon)$	k-sparse WB
Algorithm 1, now	$1 + 4\rho + O(\epsilon)$	$5 + O(\epsilon)$	general case
Algorithm 2, now	$2 + 6\rho$	$8 + O(\epsilon)$	strictly only, no violation

solve an LP on T to obtain the optimal assignment matrix ϕ_T^*. T and ϕ_T^* can be viewed as a relaxed solution for (α, β)-fair k-means, *i.e.*, the number of centers may be more than k, and meanwhile, the cost is bounded and the fairness constraints are also preserved. And we adjust the location of T. For each "potential" center $t \in T$, we update the location of t to be the corresponding cluster centroid. The adjusted T is denoted by $\pi(T)$.

Merge: We run a ρ-approximate k-means algorithm on $\pi(T)$ to obtain centers set S. Then, we solve an LP on S to obtain the optimal assignment matrix ϕ_S^* and rounding the solution to an integral solution by our proposed rounding technique.

References

1 Martial Agueh and Guillaume Carlier. Barycenters in the wasserstein space. *SIAM Journal on Mathematical Analysis*, 43(2):904–924, 2011.

2 Suman Bera, Deeparnab Chakrabarty, Nicolas Flores, and Maryam Negahbani. Fair algorithms for clustering. *Advances in Neural Information Processing Systems*, 32, 2019.

3 Matteo Böhm, Adriano Fazzone, Stefano Leonardi, Cristina Menghini, and Chris Schwiegelshohn. Algorithms for fair k-clustering with multiple protected attributes. *Operations Research Letters*, 49(5):787–789, 2021.

4 Flavio Chierichetti, Ravi Kumar, Silvio Lattanzi, and Sergei Vassilvitskii. Fair clustering through fairlets. *Advances in neural information processing systems*, 30, 2017.

5 Vincent Cohen-Addad, Philip N Klein, and Claire Mathieu. Local search yields approximation schemes for k-means and k-median in euclidean and minor-free metrics. *SIAM Journal on Computing*, 48(2):644–667, 2019.

6 Jiří Matoušek. On approximate geometric k-clustering. *Discrete & Computational Geometry*, 24(1):61–84, 2000.

7 Melanie Schmidt, Chris Schwiegelshohn, and Christian Sohler. Fair coresets and streaming algorithms for fair k-means. In *Approximation and Online Algorithms: 17th International Workshop, WAOA 2019, Munich, Germany, September 12–13, 2019, Revised Selected Papers 17*, pages 232–251. Springer, 2020.

8 Qingyuan Yang and Hu Ding. Approximate algorithms for k-sparse wasserstein barycenter with outliers. *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pages 5316–5325, 8 2024. Main Track. doi:10.24963/ijcai.2024/588.

Geodesic k-center in a simple polygon with sites on the boundary

Kien C. Huynh
Linköping University

Joseph S. B. Mitchell
Department of Applied Mathematics and Statistics, Stony Brook University, USA

Anurag Murty Naredla
Institute of Computer Science, University of Bonn, Bonn, Germany

Valentin Polishchuk
Communications and Transport Systems, ITN, Linköping University, Sweden

Abstract

We study the geodesic k-center variant where we are given a simple polygon P with a set of sites S on the boundary of P, and the objective is to cover S with k geodesic disks of minimum radius. Using amortized analysis, we provide an algorithm with a running time of $O(nm\log(nm) + m^4)$ where n is the number of vertices and m is the cardinality of S. We mainly discuss the continuous version of this problem (the centers of the k disks can be anywhere in P), but the results can be applied to the discrete version with minor changes.

2012 ACM Subject Classification Theory of computation → Computational geometry

Keywords and phrases Geodesic k-center

1 Introduction

The k-center problem is a classic optimization problem with applications in many areas, including facility allocation and urban planning. It has been studied extensively in different settings (see [4][5][2][1]). In this work, we consider the following variant: Given a simple polygon P of n vertices with m sites $\{u_1, u_2, \dots u_{m+1} = u_1\} = S \subset \partial P$, find k centers so that the maximum geodesic distance of any site in S to its nearest center is minimized. This objective is equivalent to finding k geodesic disks in P of (the same) minimum radius so that they cover S. The problem is a generalization of [3] in which the authors considered convex polygons, and can be seen as placing centers of interest in a domain (museum, park) so that the maximum distance from each entrance u_i to the nearest center is minimized.

We assume that the sites are ordered clockwise along the boundary. We use $S(i, j)$ to denote the set of sites from u_i to u_j, inclusive. Additionally, we let $\pi(x, y)$ denote the geodesic shortest path between two points x and y with respect to P. When $x = u_i$ and $y = u_j$ are sites in S, we let $\pi(i, j)$ be the geodesic shortest path from u_i to u_j. W.l.o.g., we assume that $i < j$ and the path $\pi(i, j)$ is directed from u_i to u_j, so a site u_k is said to be "on the right" of $\pi(i, j)$ if $j < k < (i + n)$.

2 Dynamic programming algorithm

We start with a simple "no-crossing" lemma, the foundation of our DP algorithm. This lemma claims that we can always find an optimal k-center solution with the following property:

▶ **Lemma 1.** *If a_1, a_2 are any two sites assigned to the center c_a, and b_1, b_2 are assigned to c_b, $c_a \neq c_b$, then no traversal along the boundary ∂P encounters the sites in the order a_1, b_1, a_2, b_2.*

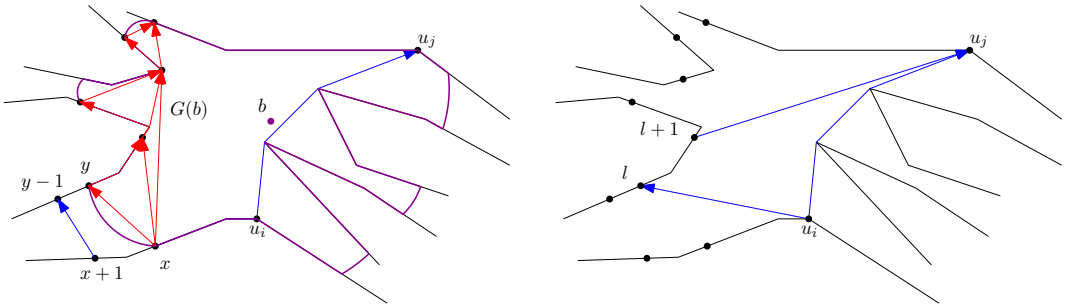


Figure 1 The two cases of the recursion in our algorithm. Left: both sites u_i and u_j belong to the same disk b (purple). The red directed edges show the edges of the graph $G(b)$. Right: u_i and u_j belong to different disks.

Our overall algorithm involves two steps: (1) perform a binary search on the radius r and (2) for each fixed radius we execute a DP algorithm to verify if it is possible to cover all of S using k disks. In our DP, we define $W(i,j), i \leq j$, as the minimum number of disks of radius r needed to cover all sites in S with indices between i and j, including i and j. In step (2), we begin with computing the set of geodesic disks of radius r centered at the sites in S and build an arrangement A of disks in P. Consider any 2-face f with non-zero area in this arrangement whose boundary is comprised of convex or concave arcs of disks in A. Let γ, which we call a cell, be the union of the *interior* of a 2-face f and its convex arcs, then it follows that any disk of radius r whose center is in γ will always contain the same subset of sites in S. Therefore, for each cell γ we can pick an arbitrary disk to represent it. Since there is one disk per cell and A has at most $O(m^2)$ faces, there are at most $O(m^2)$ disks for the algorithm to work with.

We can calculate $W(i,j)$ using recursion which has two separate cases (see Figure 1):

(i) u_i and u_j belong to the same disk. In the trivial case where all sites from u_i to u_j can be covered by one disk, we assign $W(i,j) = 1$. Let $B(i,j)$ be the set of disks that contain both u_i and u_j but no site from u_{j+1} to u_{i-1}, i.e. each disk in $B(i,j)$ does not contain any site in S to the right of $\pi(i,j)$. For each $b \in B(i,j)$, let $G(b)$ be the directed graph whose vertex set is $V(b) = S \cap b$ and edge set is $E(b) = \{e(x,y) \mid i \leq x < y \leq j\} \setminus \{e(i,j)\}$. Each edge $e(x,y)$ has weight $W(x+1, y-1)$ if $x+3 \leq y$ otherwise its weight is 1. Essentially, the weight of edge $e(x,y)$ is the minimum number of disks needed to cover all sites to the left of $\pi(x,y)$, excluding u_x, u_y. Let $\rho(b)$ be the total weight of the shortest path from u_i to u_j within $G(b)$, we will compute the following value for the first case of the recursion:

$$W_1(i,j) = \min_{b \in B(i,j)} \rho(b) + 1$$

which is the best weight over all shortest paths of all disks in $B(i,j)$. If u_i and u_j cannot be contained within one disk of radius r then we can simply skip this step and let $W_1(i,j) = \infty$.

(ii) u_i and u_j belong to two different disks. In this case, we will compute:

$$W_2(i,j) = \min_{l: i \leq l < j} \big(W(i,l) + W(l+1,j)\big)$$

Finally, we let $W(i,j) = \min(W_1(i,j), W_2(i,j))$. To answer the decision question for each fixed r, we only need to check if $W(1,m) \leq k$. The entire process will fill the DP table from the bottom up, i.e., subproblems with a smaller difference between its indices will be computed before "larger" subproblem.

▶ **Lemma 2.** *The above DP algorithm can correctly verify whether it is possible to cover S using at most k disks of radius r.*

A naive analysis of the above algorithm will give us an $O(nm\log(nm) + m^6)$ running time. The preprocessing requires $O(nm\log(nm) + m^2)$ [1] to compute the arrangement A of geodesic disks. For the DP, computing $W_1(i, j)$ takes time $O(m^4)$: there are $O(m^2)$ different disks to consider, and for each disk building the graph with $O(m)$ vertices and compute the shortest path takes time $O(m^2)$. Therefor, the DP has a running time of $O(m^6)$ for all pairs of $(i, j)_{i<j}$. However, we can use amortized analysis to reduce the overall running time to $O(nm\log(nm) + m^4)$. The intuition is that each disk of a cell in the arrangement is visited only once in the entire DP, reducing the running time by a factor of m^2.

▶ **Theorem 3.** *The continuous geodesic k-center problems in a simple polygon where sites are on its boundary can be solved in time $O(nm\log nm + m^4)$.*

The above algorithm can be reused for the discrete version where the centers can only be chosen from a given discrete set of points. Let M be the number of candidate center points in the discrete version, then the total running time will now be $O(nm\log(nm)+\min(M, m^2)m^2)$: this is because we only need to consider cells (γ) that have at least one candidate center in it.

References

1 Magdalene G Borgelt, Marc Van Kreveld, and Jun Luo. Geodesic disks and clustering in a simple polygon. *International Journal of Computational Geometry & Applications*, 21(06):595–608, 2011.

2 Sergio Cabello, Panos Giannopoulos, Christian Knauer, Dániel Marx, and Günter Rote. Geometric clustering: fixed-parameter tractability and lower bounds with respect to the dimension. *ACM Transactions on Algorithms (TALG)*, 7(4):1–27, 2011.

3 Jongmin Choi, Jaegun Lee, and Hee-Kap Ahn. Efficient k-center algorithms for planar points in convex position. In *Algorithms and Data Structures Symposium*, pages 262–274. Springer, 2023.

4 Teofilo F Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical computer science*, 38:293–306, 1985.

5 Dorit S Hochbaum and David B Shmoys. A best possible heuristic for the k-center problem. *Mathematics of operations research*, 10(2):180–184, 1985.

Robust Algorithms for Finding Triangles and Computing the Girth in Disk and Transmission Graphs

Katharina Klost
Institut für Informatik, Freie Universität Berlin, Germany

Kristin Knorr
Institut für Informatik, Freie Universität Berlin, Germany

Wolfgang Mulzer
Institut für Informatik, Freie Universität Berlin, Germany

Abstract

We describe *robust* algorithms for (unit) disk graphs and transmission graphs: the input is not given geometrically, but rather as an abstract graph, and it may or may not be realizable as a (unit) disk graph or a transmission graph. If the graph is realizable, the algorithm must give the correct answer. If not, the algorithm will either give a correct answer or correctly state that the input is not of the required kind. We consider the problem of finding a triangle and of computing the girth.

2012 ACM Subject Classification Theory of computation → Computational geometry

Keywords and phrases robust algorithm, (unit) disk graph, girth

1 Introduction

Given a set $S \subseteq \mathbb{R}^2$ of n *sites* in the plane, where each site $s \in S$ has an *associated radius* $r_s > 0$, the *disk graph* $D(S) = (S, E)$ has an edge $\{u, v\} \in E$ iff the disks defined by u and v intersect. If all associated radii are 1, $D(S)$ is called a *unit disk graph*. The *transmission graph* on S is the *directed* graph with vertex set S and a directed edge st from site s to t iff t lies inside the disk of radius r_s around s. There is plenty of literature on (unit) disk graphs, e.g., [2, 4, 6, 8, 11], whereas transmission graphs are not as widely studied [11, 12]. Although disk graphs and transmission graphs may have up to $\Omega(n^2)$ edges, they can be described with only $O(n)$ numbers: the coordinates of the sites and the associated radii. Most results assume that a geometric realization of the (unit) disk graph or the transmission graph is given. Consider a different scenario: we are given an *abstract* graph G (e.g., as an adjacency list) which may be a (unit) disk graph or a transmission graph. We would like to process G so that if G actually is a special graph, we can still take advantage of the additional structure given by the geometry (and if not, we would like to detect this). One major hurdle (unlike, e.g., in the setting of planar graphs) is that the problem of deciding whether an abstract graph is a (unit) disk graph or a transmission graph is $\exists\mathbb{R}$-hard [10, 13].

To address this problem, Raghavan and Spinrad [16] introduced *robust* algorithms in *restricted domains*. A restricted domain is a subset of the possible inputs. Contrary to the *promise* setting, where the algorithm only gives guarantees for inputs from the restricted domain, the output in the robust setting must always be useful. If the input comes from the restricted domain, the algorithm must return a correct result. If not, the algorithm may either return a correct result, or correctly state that the input does not meet the requirement.

Raghavan and Spinrad [16] give a robust polynomial algorithm for finding maximum *cliques* in unit disk graphs. Here, we revisit the notion of robust algorithms for geometric intersection graphs. Unlike Raghavan and Spinrad [16], we focus on two problems that are polynomial on general graphs, but that can be solved much faster on (unit) disk graphs and

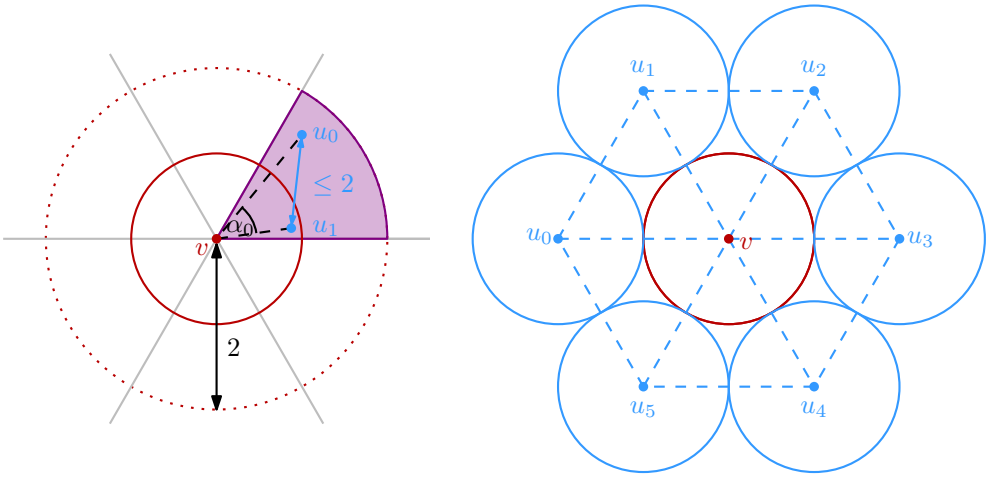


Figure 1 If the mutual distance between the u_i is maximized, 6 disks induce at least one triangle.

transmission graphs: finding a triangle and computing the *girth* (the shortest unweighted cycle). The best known algorithm for general graphs uses matrix multiplication and runs in either $O(n^\omega)$ or $O(n^{2\omega/(\omega+1)})$ time, where $\omega \leq 2.371552$ [1, 7, 9, 17]. For special graph classes, better results are known, e.g. an $O(n)$-time algorithm for *planar* graphs [5, 9, 15] or an $O(n \log n)$-time algorithm for general disk graphs [11].

2 Results

Throughout, the input is an abstract graph $G = (V, E)$, given as an adjacency list.

▶ **Theorem 2.1.** *There is a robust algorithm to find a triangle in a graph from the domain of unit disk graphs that runs in $O(n)$ time.*

The algorithm works as follows: If there is no vertex v with $\deg(v) > 5$, we check explicitly for every vertex if it has two adjacent neighbors. This directly solves the problem. Otherwise, let v be a vertex with $\deg(v) > 5$. Let $N'(v)$ be a set with six neighbors of v. For every neighbor pair $u, w \in N'(v)$, explicitly check if u and w are adjacent. If so, report the triangle u, v, w. Otherwise, report that the input is not a unit disk graph (see Figure 1).

The problem becomes harder in general disk graphs. In particular, every star is a general disk graph (see Figure 2), and hence we cannot work with a bounded degree anymore. However, a result by KKMRSS [11] shows that any non-planar (unit) disk graph contains a triangle at the crossing edges of an embedding (see Figure 3). The proof uses the observation that two intersecting disks form a lens that is intersected by at least one disk of a crossing edge. By contrapositive, if a (unit) disk graph does not contain a triangle, it must be planar.

Hence, we first perform a linear-time planarity test (e.g., [3]) on G. If G is planar, we can find a triangle with the algorithm by Chang and Lu [5], in linear time. If G is not planar, we get a Kuratowski subdivision G_K as a certificate. By the result of KKMRSS there has to be a triangle in the induced subgraph G_{iK}. In order to test for triangles we iterate over all edges $\{u, v\} \in E_K$ and traverse their sorted adjacency lists in an interleaved fashion. (The sorting is achieved in $O(m + n)$ time by transposing.) If the same vertex w is found in both lists, stop and report the triangle u, v, w, see Figure 4. Otherwise, if no such triangle was found after testing all edges in E_K, report that G is not a disk graph. Leading to:

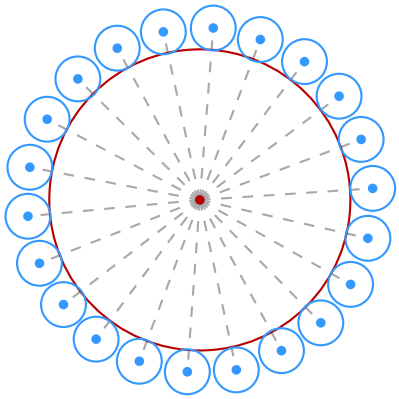


Figure 2 Every star can be realized as a general disk graph.

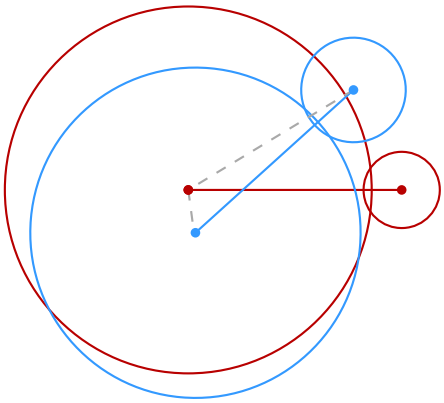


Figure 3 At least three of the disks of crossing edges have a common point.

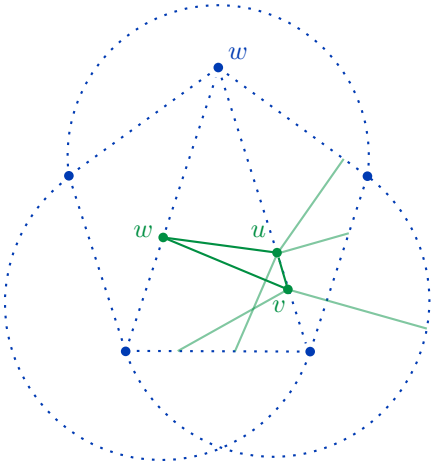


Figure 4 The vertices u, v form an edge in E_K and have a common neighbor. Thus, the triangle u, v, w is found. The green edges are edges in G_{iK}.

▶ **Theorem 2.2.** *There is a robust algorithm to find a triangle in a graph in the domain of general disk graphs that runs in $O(n + m)$ time.*

By a similar strategy to KKMRSS [11], we extend the approach to compute the girth.

▶ **Corollary 2.3.** *There is a robust algorithm to compute the girth of a graph in the domain of unit disk graphs (general disk graphs) that runs in $O(n)$ ($O(n + m)$) time.*

In the first step, we run the robust algorithm of the desired domain to find a triangle. Depending on the outcome we either report the result (non-domain, girth equals 3) or proceed. In case of neither existence of a triangle nor domain violations we test for planarity. If the graph is not planar, the non-existence of a triangle contradicts domain membership. Otherwise, we compute the girth with the algorithm by Chang and Lu [5].

For the directed variant, the transmission graphs, a result by Klost [14] allows us to bound the bidirectional degree by 7 to ensure the existence of a triangle. Given this, we can proceed in a similar manner as before.

▶ **Theorem 2.4.** *There is a robust algorithm that finds a directed triangle in a transmission graph in $O(n + m)$ time.*

References

1 Noga Alon, Raphael Yuster, and Uri Zwick. Finding and counting given length cycles. *Algorithmica*, 17(3):209–223, 1997. doi:10.1007/BF02523189.

2 Alexander Baumann, Haim Kaplan, Katharina Klost, Kristin Knorr, Wolfgang Mulzer, Liam Roditty, and Paul Seiferth. Dynamic connectivity in disk graphs. *Discrete Comput. Geom.*, 71(1):214–277, 2024. doi:10.1007/s00454-023-00621-x.

3 John Boyer and Wendy Myrvold. On the Cutting Edge: Simplified O(n) Planarity by Edge Addition. 8(3):241–273. URL: https://jgaa.info/index.php/jgaa/article/view/paper91, doi:10.7155/jgaa.00091.

4 Sergio Cabello and Miha Jejčič. Shortest paths in intersection graphs of unit disks. *Comput. Geom. Theory Appl.*, 48(4):360–367, 2015. doi:10.1016/j.comgeo.2014.12.003.

5 Hsien-Chih Chang and Hsueh-I Lu. Computing the girth of a planar graph in linear time. *SIAM J. Comput.*, 42(3):1077–1094, 2013. doi:10.1137/110832033.

6 Brent N. Clark, Charles J. Colbourn, and David S. Johnson. Unit disk graphs. *Discrete Mathematics*, 86(1):165–177, 1990. doi:10.1016/0012-365X(90)90358-O.

7 François Le Gall. Powers of tensors and fast matrix multiplication. In *Proc. 39th International Symposium on Symbolic and Algebraic Computation (ISSAC)*, pages 296–303, 2014. doi:10.1145/2608628.2608664.

8 A. Gräf, M. Stumpf, and G. Weißenfels. On coloring unit disk graphs. *Algorithmica*, 20(3):277–293, 1998. doi:10.1007/PL00009196.

9 Alon Itai and Michael Rodeh. Finding a minimum circuit in a graph. *SIAM J. Comput.*, 7(4):413–423, 1978.

10 Ross J. Kang and Tobias Müller. Sphere and dot product representations of graphs. *Discrete Comput. Geom.*, 47(3):548–568, 2012. doi:10.1007/s00454-012-9394-8.

11 Haim Kaplan, Katharina Klost, Wolfgang Mulzer, Liam Roditty, Paul Seiferth, and Micha Sharir. Triangles and girth in disk graphs and transmission graphs. In *Proc. 27th Annu. European Sympos. Algorithms (ESA)*, pages 64:1–64:14, 2019. Full version available at https://arxiv.org/abs/1907.01980. doi:10.4230/LIPIcs.ESA.2019.64.

12 Haim Kaplan, Wolfgang Mulzer, Liam Roditty, and Paul Seiferth. Spanners and reachability oracles for directed transmission graphs. In *Proc. 31st Int. Sympos. Comput. Geom. (SoCG)*, pages 156–170, 2015. doi:10.4230/LIPIcs.SOCG.2015.156.

13 Katharina Klost. Complexity of recognizing generalized transmission graphs. Master's thesis, Freie Universität Berlin, 2017.

14 Katharina Klost. *Geometric graphs: reachability, long trees and short cycles*. PhD thesis, Freie Universität Berlin, 2021.

15 Christos H. Papadimitriou and Mihalis Yannakakis. The clique problem for planar graphs. *Inform. Process. Lett.*, 13(4):131–133, 1981. doi:10.1016/0020-0190(81)90041-7.

16 Vijay Raghavan and Jeremy Spinrad. Robust algorithms for restricted domains. *J. Algorithms*, 48(1):160–172, 2003. doi:10.1016/S0196-6774(03)00048-8.

17 Virginia Vassilevska Williams, Yinzhan Xu, Zixuan Xu, and Renfei Zhou. New bounds for matrix multiplication: From alpha to omega. In *Proc. 35th Annu. ACM-SIAM Sympos. Discrete Algorithms (SODA)*, pages 3792–3835, 2024. doi:10.1137/1.9781611977912.

Minimum Ply Geometric Set Cover in the Continuous Setting

Sathish Govindarajan
Department of Computer Science and Automation, IISc, Bengaluru, India

Mayuresh Patle
Department of Computer Science and Automation, IISc, Bengaluru, India

Siddhartha Sarkar
Department of Computer Science and Automation, IISc, Bengaluru, India

Abstract

In the minimum ply geometric set cover problem (continuous setting), the goal is to cover a given set of points with copies of a given geometric object, while minimizing the ply of the cover. For different objects like unit squares, unit disks, etc. we present a simple and fast algorithm for the minimum ply cover. We show that the problem of minimizing the number of objects in a 1-ply cover for unit squares is NP-hard.

For fixed-sized hyperboxes in d-dimensional space, our algorithm computes an optimal 1-ply cover, which is up to 2^{d-1} times the size of a minimal 1-ply cover in $O(dn \log n)$ time. We show that for unit disks, a ply of 2 is sufficient to cover any point set. We also provide an algorithm producing a 2-ply cover of size at most 7 times the minimal cover in $O(n \log n)$ time. For arbitrary convex polygons with m vertices, we present an algorithm producing a 4-ply cover in $O(n \log n + nm)$ time.

2012 ACM Subject Classification Theory of computation → Computational geometry

Keywords and phrases Geometric Set Cover, Minimum Ply Covering, Minimum-Membership Covering, Approximation Algorithms.

1 Introduction

We study the Minimum Ply Geometric Set Cover, i.e., MPGSC problem in the continuous setting [2]. In this problem, the input consists of a set P of n points in $\mathbb{R}^2$, and a geometric object t, the goal is to find a set $\mathcal{S}$ of translated copies of t that covers all the points in P while minimizing $\mathsf{ply}(\mathcal{S})$, where $\mathsf{ply}(\mathcal{S}) = \max_{p \in \mathbb{R}^2} |\{s \in \mathcal{S} : p \in s\}|$. A related problem is the Minimum-Membership Geometric Set Cover (MMGSC), where we minimize the membership, defined as, $\mathsf{memb}(P, \mathcal{S}) = \max_{p \in P} |\{s \in \mathcal{S} : p \in s\}|$.

The MPGSC problem in the discrete setting was introduced by Biedl et al [2], who presented NP-hardness and approximation results. In the discrete setting, Durocher et al. presented the first constant approximation for MPGSC with unit squares [3]. Whereas, MMGSC was introduced by Erlebach et al [4]. Bandyapadhyay et al. introduced the Generalized Minimum-Membership Geometric Set Cover problem, which generalizes MPGSC and MMGSC. They gave a polynomial-time constant approximation for unit squares [1].

Research on geometric set cover often focuses on the discrete setting, where both the points to be covered and the covering objects are confined to predefined positions. Several variants of this problem are used to model interference minimization in cellular networks, among other applications. However, many real-world scenarios require continuous flexibility in the placement of covering objects, leading to our study of the continuous variant of the problem. Below, we define a minimum size variant of the continuous 1-ply cover.

▶ **Definition 1** (Minimum Size 1-Ply Geometric Set Cover of Unit Squares)**.** *Given a set of n points P on $\mathbb{R}^2$, the goal in* MS1P-GSC-US *is to cover P with the minimum number of non-overlapping unit squares.*

2 Main Results

We first present efficient algorithms for constructing constant-ply covers for various geometric objects, and provide approximation results on their size. Then we establish the NP-hardness of MS1P-GSC-US (refer to Definition 1).

2.1 Algorithms

For a given set P of n points, we summarize our algorithmic results for MPGSC below.

Object	Ply (Sufficient)	Approx. Ratio	Running Time
Unit Interval	1	1	$O(n \log n)$
Unit Square	1	2	$O(n \log n)$
d-dimensional Hyperbox	1	2^{d-1}	$O(dn \log n)$
Unit Disk	2	7	$O(n \log n)$
Convex Polygon (m vertices)	4	–	$O(n \log n + nm)$

In the third column, we approximate the size of the solution. We also show that a 1-ply cover can be constructed if the geometric object t tiles the plane. Moreover, for objects that do not tile the plane, we show a set of points for which a 1-ply cover does not exist.

In this work, we prove the bounds on ply, which implies the same bounds for membership. For example, we construct a 1-ply hypercube cover. Since a 1-ply cover is a set of non-overlapping objects, the membership of any point is at most 1. Hence, this cover is also a 1-membership cover.

2.2 Hardness

▶ **Theorem 2.** MS1P-GSC-US *is NP-hard.*

Proof. (Sketch) We reduce from PLANAR3SAT, which is known to be NP-hard [7]. Given a PLANAR3SAT formula φ with n variables and m clauses, we construct a corresponding instance P_φ of MS1P-GSC-US. The variable-clause incidence graph of φ can be embedded as a planar graph with variables along a horizontal axis [6]. The clauses are represented as non-crossing three-legged "combs" above or below the axis [5]. We refer to Figure 1.

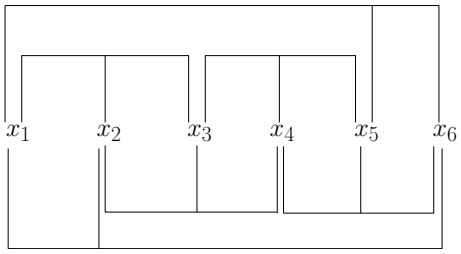


Figure 1 A rectilinear planar embedding of a PLANAR3SAT instance.

We construct P_φ, which is a set of points for which a 1-ply cover of at most k (defined later) unit squares exists if and only if φ is satisfiable. Each variable x_i is represented by a chain of unit squares, colored red and blue alternately. This forms two possible configurations:

1. True assignment: Select red squares (one configuration).
2. False assignment: Select blue squares (alternative configuration).

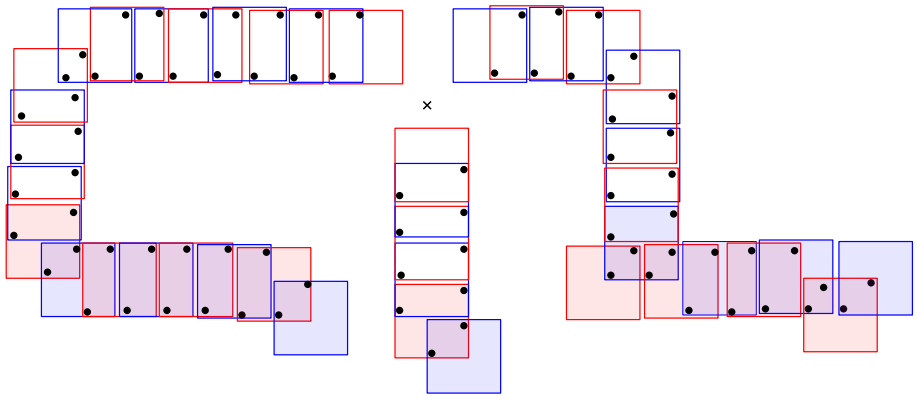


Figure 2 The variable and clause gadgets for a clause $\alpha = (\neg x_i \vee \neg x_j \vee x_k)$. Each variable gadget is a set of points determined by a chain of red and blue squares (shown as shaded). The clause gadget is composed of a clause point (shown as a cross) and some other points determined by three chains of (unshaded) squares. The interaction between a variable gadget and a clause gadget depends on whether the variable appears in its positive or negative form in the clause.

We refer to Figure 2. These two choices ensure that each variable is assigned exactly one truth value. On the other hand, each clause C_j is represented by a *clause* point and other points determined by a three-legged structure composed of unit squares. If at least one literal is satisfied, a covering square for the clause point can be placed efficiently. If no literal is satisfied, additional squares are needed, increasing the size of the solution.

We set k to be $m + c$, where c is half the number of squares placed for the variable and clause gadgets taken together. If φ is satisfiable, we can select a corresponding set of k non-overlapping unit squares covering all points in P_φ. If there exists a 1-ply unit square cover of size at most k, it corresponds to a satisfying assignment of φ. Since PLANAR3SAT is NP-hard, it follows that MS1P-GSC-US is also NP-hard. ◀

References

1 Sayan Bandyapadhyay, William Lochet, Saket Saurabh, and Jie Xue. Minimum-membership geometric set cover, revisited. In *International Symposium on Computational Geometry*, 2023.

2 Therese Biedl, Ahmad Biniaz, and Anna Lubiw. Minimum ply covering of points with disks and squares. *Computational Geometry*, 94:101712, 2021. doi:10.1016/j.comgeo.2020.101712.

3 Stephane Durocher, J. Mark Keil, and Debajyoti Mondal. Minimum ply covering of points with unit squares. In *WALCOM: Algorithms and Computation: 17th International Conference and Workshops, WALCOM 2023, Hsinchu, Taiwan, March 22–24, 2023, Proceedings*, page 23–35, Berlin, Heidelberg, 2023. Springer-Verlag. doi:10.1007/978-3-031-27051-2_3.

4 Thomas Erlebach and Erik Jan van Leeuwen. Approximating geometric coverage problems. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '08, page 1267–1276, USA, 2008. Society for Industrial and Applied Mathematics.

5 Xavier Goaoc, Jan Kratochvíl, Yoshio Okamoto, Chan-Su Shin, and Alexander Wolff. Moving vertices to make drawings plane. In Seok-Hee Hong, Takao Nishizeki, and Wu Quan, editors, *Graph Drawing*, pages 101–112, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

6 Donald E. Knuth and Arvind Raghunathan. The problem of compatible representatives. *SIAM Journal on Discrete Mathematics*, 5(3):422–427, 1992. arXiv:https://doi.org/10.1137/0405033, doi:10.1137/0405033.

7 David Lichtenstein. Planar formulae and their uses. *SIAM J. Comput.*, 11(2):329–343, May 1982. doi:10.1137/0211025.

Strong k-Metric Embeddings

Brennon Treadwell
Oregon State University, United States

Amir Nayyeri
Oregon State University, United States

Abstract

Strong k-metrics were introduced recently (SoCG 2024) as a generalization of traditional metrics, defining a metric-like function on k-tuples of points. We continue the study of strong k-metrics, examining generalized metric embeddings and possible applications. In particular we present results analogous to the embeddability of metrics into spaces with ℓ_2 norms.

2012 ACM Subject Classification Theory of Computation

Keywords and phrases metric spaces, metric embeddings, k-metrics

Digital Object Identifier 10.4230/LIPIcs...

Funding This research is supported by NSF grants CCF-1941086, and CCF-2311179.

1 Introduction and Background

A metric space consists of a set of points X and a metric function d assigning a non-negative value to all pairs of points. They generalize the notion of distance and have been applied in various sub-fields of computer science with much success. For instance, metric embeddings which relate different metrics to each other are used in the design of approximation algorithms for problems involving flows and cuts [1]. We study a higher-order generalization of metric spaces known as *strong* k-metric spaces. A strong k-metric is a set X and a function d which assigns non-negative values to all k-tuples of points in addition to satisfying certain axioms. Strong k-metric spaces are not the only generalization of metric spaces, but they enforce a stronger condition on the k-metric function d which proves more useful in generalizing results in metric embeddings [2, 4, 5]. A natural example of strong k-metrics is $(k-1)$-dimensional volume – the volume of the convex hull of k vectors in $\mathbb{R}^{k-1}$.

Below we describe ongoing research into strong k-metrics, referred to hereafter as k-metrics, including results in strong k-metric embeddings. It is hoped that these results and ongoing research will result in the development of approximation algorithms for problems like the topological sparsest cut problem [8] or the combinatorial simplicial min-cut which, unlike its graph variant, is NP-hard [7].

We refer the reader to [6] for an overview of metric spaces and metric embeddings and [2] for an overview of k-metric spaces. For the sake of brevity, we highlight only the most important background information for understanding the following results.

A k-metric embedding is a map from one k-metric space to another. Ideally such a map preserves the metric value, so that if some k-tuple has metric value x, then the image of that k-tuple also has metric value x. It is common, though, for embeddings to allow some distortion of the metric values and in such cases we naturally focus on bounding the distortion. In particular, we study embeddings into k-metric spaces with ℓ_p norms, denoted $\mathcal{C}^m_{k,p}$, where m is the dimension of the embedding and p denotes the ℓ_p norm used. A metric is in $\mathcal{C}^m_{k,p}$ if and only if there exists a matrix F such that $d = \|\delta_{k-2}F\|_p$, where d is viewed as a vector and δ_{k-2} represents the simplicial coboundary operator [2].

2 Metric Embeddings into $\mathcal{C}_{3,2}$ with Possible Applications

It is known for 2-metrics that given an $(n+1)$-point metric space, there is a polynomial-time algorithm which determines to any desired precision the smallest distortion for which the metric embeds in ℓ_2 [6]. Further, a 2-metric d isometrically embeds in ℓ_2 if and only if there exists vectors $g_i, g_j \in \mathbb{R}^m$ such that for some fixed point $x_0 \in X$ and all $x_i, x_j \in X$ we have:

$$g_i \cdot g_j = \frac{d(x_0,x_i)^2 + d(x_0,x_j)^2 - d(x_i,x_j)^2}{2}.$$

We present analogous results for 3-metrics in Lemmas 1 and 2.

▶ **Lemma 1.** *Given a 3-metric space $(X = \{x_0, \ldots, x_n\}, d : X^3 \to \mathbb{R}_{\geq 0})$, $d \in \mathcal{C}^m_{3,2}$ if and only if there exists $y_{i,j} \in \mathbb{R}^m$, with $y_{i,j} = -y_{j,i}$, such that*

$$y_{i,j}y_{j,k} + y_{i,j}y_{k,i} + y_{j,k}y_{k,i} = \frac{d(x_i,x_j,x_k)^2 - (d(x_0,x_i,x_j)^2 + d(x_0,x_i,x_k)^2 + d(x_0,x_j,x_k)^2)}{2}.$$

▶ **Lemma 2.** *There exists a polynomial time $(1+\varepsilon)$-approximation algorithm for computing minimum distortion embedding of a 3-metric into $\mathcal{C}_{3,2}$.*

The proof for both of these lemmas is similar to the classic proofs for the corresponding results. For Lemma 1, we show that given a metric in $\mathcal{C}^m_{3,2}$ one can construct $y_{i,j}$ from the matrix F which defines the metric. Conversely, given a collection of $y_{i,j}$ satisfying the above hypothesis, one can construct F as the matrix whose rows are given by $y_{i,j}$ which satisfies $d = \|\delta_1 F\|$. For Lemma 2, we express the problem as a relaxation of the isometric embedding seen in Lemma 1. In particular, we construct an SDP with constraints given by the metric values, variables representing the embedding, and with an objective function that minimizes the distortion. The success in generalizing these initial results for ℓ_2 embeddings, particularly the immediate similarity to the standard proofs, motivates further study in this area. It is the hope that a generalization of Bourgain's theorem [3] to k-metrics, for instance, will lead to an initial approximation algorithm for the topological sparsest cut problem. In fact, the seminal $O(\sqrt{\lg n})$-approximation algorithm from [1] makes use of a unit ℓ_2^2-representation of a graph which appears to be generalizable to k-metrics given the results above.

3 Determining Embeddability of 3-metrics in 1 dimension

Among the most basic results in metric embedding, is a linear time algorithm to determine if a metric space (X, d) isometrically embeds into the line metric, i.e. ℓ_2^1. Note that for any two points on the line their ℓ_p and ℓ_q distances are identical. Somewhat surprisingly, such a result is not easily obtained for k-metrics in general. We show certain conditions on the space (X, d) for which a polynomial time algorithm for determining whether $d \in \mathcal{C}^1_{3,2}$ is known. The simplest case is the all 1's metric. That is, (X, d) such that $d(x_i, x_j, x_k) = 1$ for all $x_i, x_j, x_k \in X$. Interestingly, for 2-metrics this is not isometrically embeddable in ℓ_2^1 when $n \geq 3$, in fact, by a simple packing argument it takes $\Omega(\log n)$ dimensions to realize this metric. But in the case of 3-metrics, or any k odd, this is trivially in $\mathcal{C}^1_{k,2}$ by taking $F = \mathbf{1} \in \mathbb{R}^n$.

At the opposite extreme, when all values of a metric are distinct, there is a polynomial time algorithm to determine if the metric is in $\mathcal{C}^1_{3,2}$.

▶ **Lemma 3.** *Given a 3-metric d which takes distinct values on all 3-tuples, there is a polynomial time algorithm to determine if $d \in \mathcal{C}^1_{3,2}$.*

We can relax the hypothesis to: given a point x_i, for all $x_j \neq x_i$, we require that $d(x_i, x_j, x_k)$ is distinct from all $d(x_i, x_j, x_{k'})$. This relaxation allows one to determine $\mathcal{C}^1_{3,2}$ embeddability for a larger class of k-metrics and demonstrates that the core of the previous argument is based on arguing that all 3-tuples adjacent to x_i need only have distinct values from 3-tuples they have 2 common points with. Further research aims to determine a global poly-time algorithm to determine 3-metric embeddability in $\mathcal{C}^1_{3,2}$, or prove one does not exist.

References

1 Sanjeev Arora, Satish Rao, and Umesh V. Vazirani. Expander flows, geometric embeddings and graph partitioning. *J. ACM*, 56(2):5:1–5:37, 2009. doi:10.1145/1502793.1502794.

2 Willow Barkan-Vered, Huck Bennett, and Amir Nayyeri. Topological k-metrics. In Wolfgang Mulzer and Jeff M. Phillips, editors, *40th International Symposium on Computational Geometry, SoCG 2024, June 11-14, 2024, Athens, Greece*, volume 293 of *LIPIcs*, pages 13:1–13:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024. URL: https://doi.org/10.4230/LIPIcs.SoCG.2024.13, doi:10.4230/LIPICS.SOCG.2024.13.

3 J. Bourgain. On lipschitz embedding of finite metric spaces in hilbert space. *Israel Journal of Mathematics*, 52(1), 1985. doi:10.1007/BF02776078.

4 Michel Deza and Ivo G. Rosenberg. n-semimetrics. *Eur. J. Comb.*, 21(6):797–806, 2000. URL: https://doi.org/10.1006/eujc.1999.0384, doi:10.1006/EUJC.1999.0384.

5 Siegfried Gähler. 2-metrische räume und ihre topologische struktur. *Mathematische Nachrichten*, 26(1-4):115–148, 1963. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/mana.19630260109, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/mana.19630260109, doi:10.1002/mana.19630260109.

6 Jiří Matoušek. Lecture notes on metric embeddings, January 2013. URL: https://kam.mff.cuni.cz/~matousek/ba-a4.pdf.

7 William Maxwell and Amir Nayyeri. Generalized max-flows and min-cuts in simplicial complexes. *CoRR*, abs/2106.14116, 2021. URL: https://arxiv.org/abs/2106.14116, arXiv:2106.14116.

8 Ori Parzanchevski, Ron Rosenthal, and Ran J. Tessler. Isoperimetric inequalities in simplicial complexes. *Comb.*, 36(2):195–227, 2016. URL: https://doi.org/10.1007/s00493-014-3002-x, doi:10.1007/S00493-014-3002-X.

Reconfigurations of Plane Caterpillars and Paths

Todor Antić
Department of Applied Mathematics, Faculty of Mathematics and Physics, Charles University, Czech Republic

Jelena Glišić
Department of Applied Mathematics, Faculty of Mathematics and Physics, Charles University, Czech Republic

Guillermo Gamboa Quintero
Computer Science Institute of Charles University, Prague, Czech Republic

Abstract

Let S be a point set in the plane, $\mathcal{P}(S)$ and $\mathcal{C}(S)$ sets of all plane spanning paths and caterpillars on S. We study reconfiguration operations on $\mathcal{P}(S)$ and $\mathcal{C}(S)$. In particular, we prove that all of the commonly studied reconfigurations on plane spanning trees still yield connected reconfiguration graphs for caterpillars when S is in convex position. If S is in general position, we show that the rotation, compatible flip and flip graphs of $\mathcal{C}(S)$ are connected while the slide graph is disconnected. For paths, we prove the existence of a connected component of size at least 2^{n-2} in the flip graph on $\mathcal{P}(S)$, and that no component of size at most 7 can exist in this graph.

2012 ACM Subject Classification Theory of computation → Randomness, geometry and discrete structures; Theory of computation → Computational geometry; Mathematics of computing → Combinatorics

Keywords and phrases reconfiguration graph, caterpillar, path, geometric graph

Digital Object Identifier 10.4230/LIPIcs...

Related Version arXiv:2410.07419

Funding *Todor Antić*: supported by grant no. 23-04949X of the Czech Science Foundation (GAČR) and by SVV–2023–260699.
Jelena Glišić: supported by grant no. 23-04949X of the Czech Science Foundation (GAČR) and by SVV–2023–260699.
Guillermo Gamboa Quintero: supported by the GAČR grant 22-17398S and the Charles University project PRIMUS/24/SCI/012.

Acknowledgements We would like to thank Jan Kynčl and Pavel Valtr for proposing the problems and for all of the helpful discussions during the combinatorial problems seminar at Charles University. We also thank Daniel Perz for pointing out the existence of an isolated vertex in $G_{\mathcal{C}}^{\text{slide}}(S)$, for his help in finding literature for this paper and helpful comments.

1 Introduction

Given a set of structures C, and a reconfiguration operation that transforms one object in C to another, the *reconfiguration graph* is a graph with vertex set C in which two vertices form an edge if one can be transformed into the other using a single reconfiguration operation. Often, in computer science, objects are solutions to a problem and reconfigurations are local changes that transform one solution into another. Then, to understand the solution space, it is important to study the reconfiguration graph. For an introduction to the topic, see [14]. Given a point set S in the plane, a *plane spanning tree* on S is a spanning tree of S whose edges are straight line segments that do not cross. Let $\mathcal{T}(S)$ be the set of all plane spanning trees on S. We consider five reconfigurations on $\mathcal{T}(S)$ (see Figure 1). For the following, we


Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

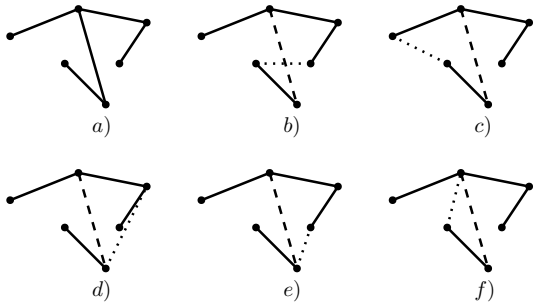


Figure 1 a) A plane spanning tree. Replacing the dashed line with the dotted line corresponds to: b) a flip, c) a compatible flip, d) a rotation, e) an empty triangle rotation and f) a slide.

are given $T_1 = (S, E_1), T_2 = (S, E_2) \in \mathcal{T}(S)$, and we say that:

1. T_1 and T_2 are connected by a *flip* if $E_2 = E_1 \setminus \{e\} \cup \{f\}$ for some edges e, f.
2. T_1 and T_2 are connected by a *compatible flip* if $E_2 = E_1 \setminus \{e\} \cup \{f\}$ for some edges e, f which do not cross.
3. T_1 and T_2 are connected by a *rotation* if $E_2 = E_1 \setminus \{e\} \cup \{f\}$ for some edges e, f which share an endpoint.
4. T_1 and T_2 are connected by an *empty triangle rotation* if $E_2 = E_1 \setminus \{e\} \cup \{f\}$ for some edges e, f which share an endpoint and the triangle spanned by their endpoints is empty.
5. T_1 and T_2 are connected by a *slide* if $E_2 = E_1 \setminus \{e\} \cup \{f\}$ for some edges e, f which are as in 4. and if $e = ab$ and $f = ac$ then $bc \in E_1 \cap E_2$.

Note that every slide is an empty triangle rotation, every empty triangle rotation is a rotation, and so on. This hierarchy will be useful when studying the structural properties of the corresponding reconfiguration graphs.

2 Background

Reconfigurations of plane spanning trees have been well studied, both for points in convex position [6, 7, 11] and in general position [1, 2, 13]. The questions that have been studied are that of connectivity of the reconfiguration graphs [1, 5, 13] and diameter [2, 7, 8]. The reconfigurations of paths have proven to be another interesting topic of study [3, 4, 9, 10, 12, 15]. While the case when points are in convex position has been well studied and many properties of the flip graph are known, for general position it is still not known if the flip graph is even connected, except in some restricted cases [3, 4].

3 Our contribution

We further the study of induced subgraphs of reconfiguration graphs of plane spanning trees by exploring reconfigurations of plane spanning caterpillars, and by expanding on the topic of reconfigurations of plane spanning paths. A *caterpillar* is a tree in which all non-leaf vertices form a path. We call this path the *spine* of the caterpillar. A *plane spanning caterpillar* of a point set S is a plane spanning tree of S which is a caterpillar. For a set S, we will denote by $\mathcal{C}(S)$ the set of all plane spanning caterpillars on S. We will denote the reconfiguration graphs on $\mathcal{C}(S)$ by $G_{\mathcal{C}}^{\text{flip}}(S), G_{\mathcal{C}}^{\text{comp-flip}}(S), G_{\mathcal{C}}^{\text{rot}}(S), G_{\mathcal{C}}^{\text{emp-rot}}(S), G_{\mathcal{C}}^{\text{slide}}(S)$. First, we focus on the case where S is in convex position. We show that the *slide graph* $G_{\mathcal{C}}^{\text{slide}}$ is connected,

which implies that all of the reconfiguration graphs are connected in this case. For compatible flips, we prove a stronger bound for the diameter.

▶ **Theorem 1.** *Let S be a set of $n \geq 3$ points in convex position in the plane. Then, the graph $G_{\mathcal{C}}^{slide}(S)$ is connected with diameter at most $3n - 8$.*

▶ **Theorem 2.** *Let S be a set of $n \geq 3$ points in convex position in the plane. Then, the graph $G_{\mathcal{C}}^{comp\text{-}flip}(S)$ is connected with diameter at most $2n - 5$.*

Next, we consider the case when S is a point set in general position. For a caterpillar $C \in \mathcal{C}(S)$, and consecutive spine vertices $v_i, \ldots, v_j$ of C, we write $S_{i,j}$ for the point set consisting of the spine vertices and all of the leaves attached to them. We call $C \in \mathcal{C}(S)$ with spine $v_1, v_2, \ldots, v_k$ a *well-separated* caterpillar if for each $i \geq 1$, the convex hull of $S_{1,i}$ is disjoint from the rest of S. We show that all caterpillars in this class are mutually connected in $G_{\mathcal{C}}^{\text{slide}}(S)$. This large connected component is significant, as we later show that $G_{\mathcal{C}}^{\text{slide}}(S)$ is disconnected. On the other hand, we show that the rotation graph $G_{\mathcal{C}}^{\text{rot}}(S)$ is connected. We leave open the question of the connectivity of $G_{\mathcal{C}}^{\text{emp-rot}}$.

▶ **Theorem 3.** *Any two well-separated caterpillars are connected in $G_{\mathcal{C}}^{slide}(S)$.*

▶ **Theorem 4.** *The graph $G_{\mathcal{C}}^{slide}(S)$ is connected for every set S of n points in the plane if $n \leq 7$. If $n \geq 8$, there exists a set S of n points such that $G_{\mathcal{C}}^{slide}(S)$ has isolated vertices.*

▶ **Proposition 5.** *The graph $G_{\mathcal{C}}^{rot}(S)$ is connected.*

Lastly, we focus on connected components of the reconfiguration graph of plane spanning paths. Given a set of points in general position S, we will call the corresponding *flip graph of plane spanning paths* $G_{\mathcal{P}}(S)$. Currently, the main open problem is deciding whether $G_{\mathcal{P}}(S)$ is connected. Here we find a large connected component consisting of what we call *generalized peeling paths*. We introduce this subclass in Section **??**. We note that Theorem 6 was independently discovered by Kleist, Kramer and Rieck [12]. We still include it since we use the number of these paths to prove that there are at least $\frac{1}{4}(3^n - 1)$ well-separated caterpillars on S. Finally, we investigate the minimal size of components in $G_{\mathcal{P}}(S)$.

▶ **Theorem 6.** *Let S be a set of n points in general position. Then $G_{\mathcal{P}}(S)$ contains a connected component of size $\Omega(2^{n-2})$.*

▶ **Theorem 7.** *Let S be a point set of $n \geq 5$ points in general position. Then, $G_{\mathcal{P}}(S)$ contains no connected component of size at most 7.*

References

1 Oswin Aichholzer, Franz Aurenhammer, and Ferran Hurtado. Sequences of spanning trees and a fixed tree theorem. *Comput. Geom.*, 21(1-2):3–20, 2002.

2 Oswin Aichholzer, Brad Ballinger, Therese Biedl, Mirela Damian, Erik D. Demaine, Matias Korman, Anna Lubiw, Jayson Lynch, Josef Tkadlec, and Yushi Uno. Reconfiguration of non-crossing spanning trees. *ArXiv*, 2022. URL: `https://arxiv.org/abs/2206.03879`.

3 Oswin Aichholzer, Kristin Knorr, Wolfgang Mulzer, Johannes Obenaus, Rosna Paul, and Birgit Vogtenhuber. Flipping plane spanning paths. In *WALCOM*, volume 13973 of *Lecture Notes in Computer Science*, pages 49–60. Springer, 2023.

4 Selim G. Akl, Kamrul Islam, and Henk Meijer. On planar path transformation. *Inf. Process. Lett.*, 104(2):59–64, 2007.

5 David Avis and Komei Fukuda. Reverse search for enumeration. *Discret. Appl. Math.*, 65(1-3):21–46, 1996. doi:10.1016/0166-218X(95)00026-N.

6 Håvard Bakke Bjerkevik, Linda Kleist, Torsten Ueckerdt, and Birgit Vogtenhuber. *Flipping Non-Crossing Spanning Trees*, pages 2313–2325. Society for Industrial and Applied Mathematics, 2025. URL: https://epubs.siam.org/doi/abs/10.1137/1.9781611978322.77, arXiv:https://epubs.siam.org/doi/pdf/10.1137/1.9781611978322.77, doi:10.1137/1.9781611978322.77.

7 Nicolas Bousquet, Lucas de Meyer, Théo Pierron, and Alexandra Wesolek. Reconfiguration of Plane Trees in Convex Geometric Graphs. In Wolfgang Mulzer and Jeff M. Phillips, editors, *40th International Symposium on Computational Geometry (SoCG 2024)*, volume 293 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 22:1–22:17, Dagstuhl, Germany, 2024. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. URL: https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.SoCG.2024.22, doi:10.4230/LIPIcs.SoCG.2024.22.

8 Nicolas Bousquet, Valentin Gledel, Jonathan Narboni, and Théo Pierron. A note on the flip distance between non-crossing spanning trees. *Comput. Geom. Topol.*, 2(1):8:1–8:7, 2023. URL: https://www.cgt-journal.org/index.php/cgt/article/view/36.

9 Jou-Ming Chang and Ro-Yu Wu. On the diameter of geometric path graphs of points in convex position. *Inf. Process. Lett.*, 109(8):409–413, 2009.

10 Ruy Fabila-Monroy, David Flores-Peñaloza, Clemens Huemer, Ferran Hurtado, Jorge Urrutia, and David R. Wood. On the chromatic number of some flip graphs. *Discrete Mathematics & Theoretical Computer Science*, Vol. 11 no. 2, January 2009. URL: https://dmtcs.episciences.org/460, doi:10.46298/dmtcs.460.

11 M. Carmen Hernando, Ferran Hurtado, Alberto Márquez, Mercè Mora, and Marc Noy. Geometric tree graphs of points in convex position. *Discret. Appl. Math.*, 93(1):51–66, 1999.

12 Linda Kleist, Peter Kramer, and Christian Rieck. On the connectivity of the flip graph of plane spanning paths. In Daniel Kráľ and Martin Milanič, editors, *Graph-Theoretic Concepts in Computer Science*, pages 327–342, Cham, 2025. Springer Nature Switzerland.

13 Torrie L. Nichols, Alexander Pilz, Csaba D. Tóth, and Ahad N. Zehmakan. Transition operations over plane trees. *Discrete Mathematics*, 343(8):111929, 2020. doi:10.1016/j.disc.2020.111929.

14 Naomi Nishimura. Introduction to reconfiguration. *Algorithms*, 11(4), 2018. doi:10.3390/a11040052.

15 Eduardo Rivera-Campo and Virginia Urrutia-Galicia. Hamilton cycles in the path graph of a set of points in convex position. *Comput. Geom.*, 18(2):65–72, 2001. doi:10.1016/S0925-7721(00)00026-2.

Segment Intersection Representations and Constrained Level Planarity

Simon D. Fink
Algorithms and Complexity Group, Technische Universität Wien, Austria

Matthias Pfretzschner
Faculty of Computer Science and Mathematics, University of Passau, Germany

Peter Stumpf
Department of Theoretical Computer Science, Faculty of Information Technology, Czech Technical University in Prague, Czech Republic
Department of Applied Mathematics, Charles University Prague, Czech Republic

Abstract

In a seminal work from 1989, Kratochvíl and Matoušek initiated the study of intersection graphs of segments forming the class SEG [11]. A graph is in SEG if its vertices can be represented by straight line segments in the plane that intersect if and only if the corresponding vertices are adjacent. The authors not only showed that testing membership in SEG is NP-hard, but also by showing that a representation of a graph in SEG might require coordinates with exponential precision [11], laid the foundation for the study of a new complexity class denoted as $\exists\mathbb{R}$ [15]. They studied further variants of SEG (see Fig. 1 of [11]), in one direction further allowing the segments to have a bounded number of bends [13, 14], and in the other direction restricting the segments to a bounded number of slopes [10]. In the latter setting, even the case where only two directions, say horizontal and vertical, are allowed (in their work denoted as 2-DIR, and otherwise also called Grid Intersection Graph or GIG [7]) turned out NP-complete [10].

Kratochvíl and Nešetril observed that further prescribing the orders of horizontal and vertical segments, separately along both axes, finally turned the problem tractable [10, 12]. They also posed the following open problem: what if only one of the two orders (e.g. the vertical order of horizontal segments) is fixed? In this work, we will give an efficient algorithm finally resolving this question after roughly 30 years, which can formally be stated as follows; see also Figures 1a and 1b.

Problem Semi-Fixed HV-Segment Intersection Graph Recognition (SF-HV-SEG)
Input A bipartite graph $G = (H \cup V, E)$ and a linear order σ_H of H.
Question Is G the intersection graph of horizontal segments H and vertical segments V, where the horizontal segments have the fixed vertical order σ_H?

2012 ACM Subject Classification Mathematics of computing → Graph algorithms; Human-centered computing → Graph drawings

Keywords and phrases segment intersection representation, grid intersection graph, level planarity

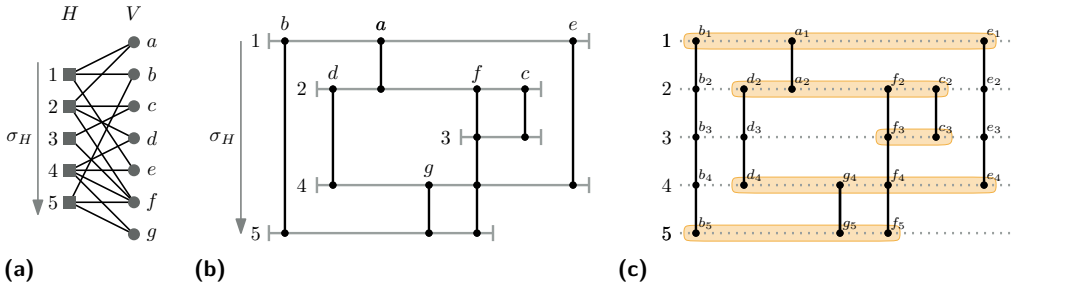


Figure 1 **(a)** A SF-HV-SEG instance, highlighting the fixed order σ_H. **(b)** A corresponding HV-segment representation, showing that it is a yes-instance. **(c)** An equivalent instance of proper $\mathcal{T}$-Level Planarity. The consecutive sets of every level are highlighted in orange.

Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Digital Object Identifier 10.4230/LIPIcs...

Related Version *Full Version*: arXiv:2502.16621 [5]

Funding *Simon D. Fink*: Vienna Science and Technology Fund (WWTF) [10.47379/ICT22029]
Matthias Pfretzschner: Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 541433306
Peter Stumpf: Czech Science Foundation grant no. 23-04949X

Acknowledgements We want to thank Jan Kratochvíl for introducing us to the "Volkswagen-Problem" at HOMONOLO'22 and, together with Jiří Fiala, also for organizing this very enjoyable workshop. We also want to thank Marie Diana Sieper and Felix Klesen who joined us at the workshop for developing early approaches towards the SF-HV-SEG problem.

1 Reducing SF-HV-SEG to proper $\mathcal{T}$-Level Planarity

To solve instances of SF-HV-SEG, we will use a reduction to a constrained planar graph drawing problem. For the Level Planarity problem, we are given a graph together with prescribed y-coordinates for its vertices (also called *levels*), and seek a crossing-free drawing that respects these y-coordinates and has all edges drawn y-monotone. This problem has been thoroughly researched in the field of Graph Drawing, e.g. in terms of efficient and also simple solutions [3, 4, 6, 8, 9] and also its relationship to other planar graph drawing problems [2, 16]. We call the tuple (G, ℓ) consisting of a graph $G = (V, E)$ and a *leveling function* $\ell : V \to \mathbb{N}$ a *level graph*. A level graph (and especially its leveling function) is called *proper* if all edges have their endpoints on adjacent levels, i.e., if we have $\ell(v) = \ell(u) + 1$ for all $uv \in E$.

We will use a further constrained variant that also allows us to require sets of vertices on the same layer to be *consecutive*, i.e., ordered such that no other vertices of that level lie between them. We will require these *consecutivity constraint* sets to form a laminar family, i.e., any two sets are either disjoint or one contains the other.[1] This problem is then known to have a quadratic-time solution if the instance is proper, while it is NP-complete on non-proper instances [1, 2]. Formally, the former, tractable case of the problem is defined as follows.

Problem proper $\mathcal{T}$-Level Planarity
Input A graph $G = (V, E)$, a proper leveling function $\ell : V \to \mathbb{N}$, and for each level i a laminar family T_i of consecutivity constraints on $\ell^{-1}(i)$.[2]
Question Is there a planar drawing of G where each vertex $v \in V$ has y-coordinate $\ell(v)$, all edges are drawn y-monotone, and the horizontal order of vertices on each level i satisfies the consecutivity constraints of T_i?

To now reduce SF-HV-SEG to this problem, we create a level for each horizontal segment along the given order; see Figure 1. For each vertical segment, we create an edge that spans from the first to the last level of horizontal segments that shall be intersected. We subdivide the edges to make the instance proper. The crucial insight to SF-HV-SEG is now that all

[1] The reduction even only generates one consecutivity constraint per level, i.e., families containing only a single set. Still, our reduction in the converse direction [5] also covers this more general case
[2] Using the containment-relationship of laminar families allows us to equivalently represent T_i as a tree of linear size, thus obtaining the original formulation of the problem [2].

vertical segments $\mathcal{I}(i)$ that intersect a given horizontal segment h_i need to be consecutive on the horizontal line corresponding to h_i. This means that any other vertical segment that spans, but does not intersect h_i, needs to cross level i before or after $\mathcal{I}(i)$, but not within. This can easily be modeled using the per-level consecutivity constraints of PROPER $\mathcal{T}$-LEVEL PLANARITY, yielding the following theorem (see the full version [5] for a formal proof).

▶ **Theorem 1.** *There exists a quadratic-time reduction from SF-HV-SEG to* PROPER $\mathcal{T}$*-LEVEL PLANARITY where the input graph is the disjoint union of level-monotone paths.*

Since PROPER $\mathcal{T}$-LEVEL PLANARITY admits a quadratic-time algorithm [1,2] and our reduction increases the size to be at most quadratic, we can thereby solve the initial instance in quartic time overall.

▶ **Corollary 2.** *SF-HV-SEG can be solved in $O(n^4)$ time.*

2 Conclusion

When posing the SF-HV-SEG question, Kratochvíl and Neŝetril already suggested two reformulations of the problem in terms of forbidden "volkswagen" patterns in 2-layer drawings or forbidden "cross" patterns in the permuted adjacency matrix [10,12]. Still, the problem remained open for 30 years until yet another different approach allowed us to show tractability via a reduction to a constrained planar graph drawing problem. In the full version [5], we show that the reduction also works in the other direction, allowing us to view both problems as different perspectives on the same underlying question. Interestingly, the PROPER $\mathcal{T}$-LEVEL PLANARITY problem is itself a reformulation of another problem called k-ARY TANGLEGRAMS [2,17], which arose in computational biology to visualize evolutionary histories of species. In addition to this very visual solution, in the full version [5], we also give another equivalent characterization as a constrained ordering problem that allows a direct solution without the need for reductions. Through the links we established, we now know that we can view the same tractable problem through 6 different-yet-equivalent formulations, some of which arose naturally and independently, and taking different perspectives from either a graph drawing or entirely combinatorial standpoint. It will be interesting to see whether further problems turn out to join this order of equivalent formulations in the future.

References

1 Patrizio Angelini, Giordano Da Lozzo, Giuseppe Di Battista, Fabrizio Frati, and Vincenzo Roselli. The importance of being proper – (in clustered-level planarity and t-level planarity). In Christian A. Duncan and Antonios Symvonis, editors, *Proceedings of the 22nd International Symposium on Graph Drawing (GD'14)*, volume 8871 of *Lecture Notes in Computer Science*, pages 246–258. Springer, 2014. doi:10.1007/978-3-662-45803-7_21.

2 Patrizio Angelini, Giordano Da Lozzo, Giuseppe Di Battista, Fabrizio Frati, and Vincenzo Roselli. The importance of being proper: (in clustered-level planarity and t-level planarity). *Theor. Comput. Sci.*, 571:1–9, 2015. doi:10.1016/J.TCS.2014.12.019.

3 Guido Brückner, Ignaz Rutter, and Peter Stumpf. Level planarity: Transitivity vs. even crossings. In Therese Biedl and Andreas Kerren, editors, *Proceedings of the 26th International Symposium on Graph Drawing and Network Visualization (GD '18)*, volume 11282 of *Lecture Notes in Computer Science*, pages 39–52. Springer, 2018. doi:10.1007/978-3-030-04414-5_3.

4 Guido Brückner, Ignaz Rutter, and Peter Stumpf. Level-planarity: Transitivity vs. even crossings. *Electronic Journal of Combinatorics*, 29(4), 2022. doi:10.37236/10814.

5 Simon D. Fink, Matthias Pfretzschner, and Peter Stumpf. Segment intersection representations, level planarity and constrained ordering problems, 2025. arXiv:2502.16621.

6 Radoslav Fulek, Michael J. Pelsmajer, Marcus Schaefer, and Daniel Štefankovič. Hanani-Tutte, monotone drawings, and level-planarity. In János Pach, editor, *Thirty Essays on Geometric Graph Theory*, pages 263–287. Springer, 2013. doi:10.1007/978-1-4614-0110-0_14.

7 I. Ben-Arroyo Hartman, Ilan Newman, and Ran Ziv. On grid intersection graphs. *Discrete Mathematics*, 87(1):41–52, January 1991. doi:10.1016/0012-365X(91)90069-E.

8 Michael Jünger and Sebastian Leipert. Level planar embedding in linear time. *Journal of Graph Algorithms and Applications*, 6(1):67–113, 2002. doi:10.7155/jgaa.00045.

9 Michael Jünger, Sebastian Leipert, and Petra Mutzel. Level planarity testing in linear time. In Sue Whitesides, editor, *Proceedings of the 6th International Symposium on Graph Drawing (GD'98)*, volume 1547 of *Lecture Notes in Computer Science*, pages 224–237. Springer, 1998. doi:10.1007/3-540-37623-2_17.

10 Jan Kratochvíl. A special planar satisfiability problem and a consequence of its np-completeness. *Discrete Applied Mathematics*, 52(3):233–252, 1994. doi:10.1016/0166-218X(94)90143-0.

11 Jan Kratochvíl and Jiří Matoušek. Intersection graphs of segments. *Journal of Combinatorial Theory, Series B*, 62(2):289–315, 1994. doi:10.1006/jctb.1994.1071.

12 Jan Kratochvíl and Jaroslav Nešetril. Problems proposed at the problem session of the prachatice conference on graph theory. In Jaroslav Nešetril and Miroslav Fiedler, editors, *Proceeding of the Fourth Czechoslovakian Symposium on Combinatorics, Graphs and Complexity*, volume 51 of *Annals of Discrete Mathematics*, pages 375–384. Elsevier, 1992. doi:10.1016/S0167-5060(08)70659-9.

13 Jan Kratochvíl. String graphs. II. recognizing string graphs is NP-hard. *Journal of Combinatorial Theory, Series B*, 52(1):67–78, May 1991. doi:10.1016/0095-8956(91)90091-W.

14 Jan Kratochvíl and Jiří Matoušek. NP-hardness results for intersection graphs. *Commentationes Mathematicae Universitatis Carolinae*, 030(4):761–773, 1989. Publisher: Charles University in Prague, Faculty of Mathematics and Physics. URL: https://eudml.org/doc/17790.

15 Marcus Schaefer. Complexity of Some Geometric and Topological Problems. In David Eppstein and Emden R. Gansner, editors, *Graph Drawing*, pages 334–344, Berlin, Heidelberg, 2010. Springer. doi:10.1007/978-3-642-11805-0_32.

16 Marcus Schaefer. Toward a theory of planarity: Hanani-tutte and planarity variants. *Journal of Graph Algorithms and Applications*, 17(4):367–440, 2013. doi:10.7155/jgaa.00298.

17 Andreas Wotzlaw, Ewald Speckenmeyer, and Stefan Porschen. Generalized k-ary tanglegrams on level graphs: A satisfiability-based approach and its evaluation. *Discret. Appl. Math.*, 160(16-17):2349–2363, 2012. doi:10.1016/J.DAM.2012.05.028.

Edge Isoperimetry of Lattices

Cameron Strachan
London School of Economics and Political Science, United Kingdom

Konrad Swanepoel
London School of Economics and Political Science, United Kingdom

Abstract

We present two results related to an edge-isoperimetric question for Cayley graphs on the integer lattice asked by Ben Barber and Joshua Erde. For any (undirected) graph G, the edge boundary of a subset of vertices S is the number of edges between S and its complement in G. Barber and Erde asked whether for any Cayley graph on $\mathbb{Z}^d$, there is always an ordering of $\mathbb{Z}^d$ such that for each n, the first n terms minimize the edge boundary among all subsets of size n. Our first result answers this question in the negative by presenting an example of a Cayley graph on $\mathbb{Z}^d$ (for all $d \geq 2$) for which there is no such ordering. Our second result is a positive example of a Cayley graph on $\mathbb{Z}^2$ that has such an ordering. This is the most complicated example known to us of a two-dimensional Cayley graph for which such an ordering exists.

2012 ACM Subject Classification Theory of computation → Computational geometry

Keywords and phrases Cayley graph, discrete isoperimetric problem, extremal combinatorics

Digital Object Identifier 10.4230/LIPIcs...

Related Version A full version of the paper is available at: `https://arxiv.org/abs/2503.09591`.

Acknowledgements We thank Peter Allen for his insightful suggestions.

1 Introduction and Background

Given a graph G, the *edge boundary* of $S \subseteq V(G)$ is $\partial(S) := |\{uv \in E(G) : u \in S, v \notin S\}|$. The *edge isoperimetric problem* (EIP) of a graph G is, for a given n, to minimize $\partial(S)$ over all $S \subseteq V(G)$ where $|S| = n$. We call such minimizing sets *solutions to the EIP of* G. This classical problem has been extensively studied since the 1960s (see [12]). Although it is NP-hard in general, some special cases are known. One aspect that has received particular attention is whether nested solutions exist. A *nested solution* for the EIP of G is an ordering $v_1, v_2, \ldots$ of the vertex set $V(G)$ such that for each n, the set $\{v_1, v_2, \ldots, v_n\}$ is a solution to the EIP of G.

One of the first cases of the EIP that has been solved is the d-dimensional cube graph, which has nested solutions, and where the optimal shapes include subcubes [3, 10, 13, 14].

For $p = 1, \infty$, denote by $(\mathbb{Z}^d, \ell_p)$ the graph with vertex set $\mathbb{Z}^d$ where pairs of vertices have an edge if their ℓ_p distance is 1. Bollobás and Leader [4] solved the EIP for $(\mathbb{Z}^d, \ell_1)$. They proved that the solutions include cubes and that $(\mathbb{Z}^d, \ell_1)$ has nested solutions.

Bollobás and Leader [4] also considered the EIP on finite grids $\{1, 2, \ldots, k\}^d$, considered as induced subgraphs of $(\mathbb{Z}^d, \ell_1)$. It turned out that there are two types of solutions: cubes if n is small relative to the size of the grid and half-grids for large n. Furthermore, they showed the transition between these two types of solutions is not smooth, giving the first example of a graph without nested solutions.

If G is an undirected k-regular graph, for any $S \subseteq V(G)$ we have $|E(G[S])| = \frac{k|S| - \partial(S)}{2}$. If G is a directed k-regular graph, then for any $S \subseteq V(G)$ we have $|E(G[S])| = k|S| - \partial(S)$. Thus, the problem of minimizing $\partial(S)$ over all subsets with size n is the same as maximizing $|E(G[S])|$ over all subsets of size n.

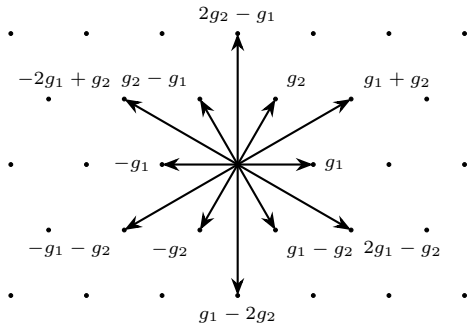


Figure 1 Generating set of Theorem 3

In terms of this formulation, Brass [5] solved the EIP of $(\mathbb{Z}^2, \ell_\infty)$, where the optimal shapes include certain octagons. Additionally, he showed $(\mathbb{Z}^2, \ell_\infty)$ has nested solutions. For $d \geq 3$, the EIP of $(\mathbb{Z}^d, \ell_\infty)$ remains open.

Let $g_1 = (1, 0)$ and $g_2 = (1/2, \sqrt{3}/2)$. The *triangular lattice* is the set $\Lambda := \{mg_1 + ng_2 : m, n \in \mathbb{Z}\}$. We can turn Λ into a graph by joining a pair of vertices if their Euclidean distance is 1. For this graph, the EIP is solved [9] (see also [8, 11]) with solutions that include regular hexagons. Again, the graph has nested solutions. This graph is isomorphic to $\mathbb{Z}^2$, where two vertices are joined if their difference is in $\{(\pm 1, 0), (0, \pm 1), \pm(1, 1)\}$, hence it can be thought of as a graph in between $(\mathbb{Z}^2, \ell_1)$ and $(\mathbb{Z}^2, \ell_\infty)$.

The above examples are all special cases of Cayley graphs on the group $\mathbb{Z}^d$ [1].

▶ **Definition 1.** *Let U be a finite set that generates $\mathbb{Z}^d$ as a group and does not contain the identity. The (directed) Cayley graph $\mathbb{Z}^d_U$ is the graph on the vertex set $\mathbb{Z}^d$ where (u, v) is an edge whenever $v - u \in U$. When U is symmetric (that is, $-u \in U$ for all $u \in U$), we consider $\mathbb{Z}^d_U$ to be undirected.*

Given a generating set U of $\mathbb{Z}^d$, let $Z \subseteq \mathbb{R}^d$ be the zonotope $\sum_{u \in U}[0, u]$ generated by the line segments $[0, u]$, $u \in U$. Barber and Erde [1] showed that the edge boundary of $tZ \cap \mathbb{Z}^d$ for large t, asymptotically approximates the edge boundary of solutions to the EIP of $\mathbb{Z}^d_U$. Barber, Erde, Keevash and Roberts [2] showed that additionally, $tZ \cap \mathbb{Z}^d$ asymptotically approximates the shape of solutions to the EIP of $\mathbb{Z}^d_U$.

Barber and Erde [1] asked if every Cayley graph of $\mathbb{Z}^d$ has nested solutions. Despite the positive examples already given, Briggs and Wells [6] gave counterexamples for the case $d = 1$. On the other hand, they also gave a partial positive answer: for any Cayley graph of $\mathbb{Z}$, there exists an $m \in \mathbb{N}$ and an ordering $v_1, v_2 \ldots$ of $\mathbb{Z}$ such that for any $n \geq m$, the set $\{v_1, v_2, \ldots, v_n\}$ is a solution to the EIP. In other words, they showed that any Cayley graph on $\mathbb{Z}$ has nested solutions starting at a sufficiently large size.

2 Results

We give a negative answer to the question of Barber and Erde for all $d \geq 2$ by giving an explicit example of a Cayley graph without nested solutions. Furthermore, we show this example does not have nested solutions regardless of any starting point. Thus, in dimensions 2 and higher there are stronger counterexamples than in $\mathbb{Z}^1$.

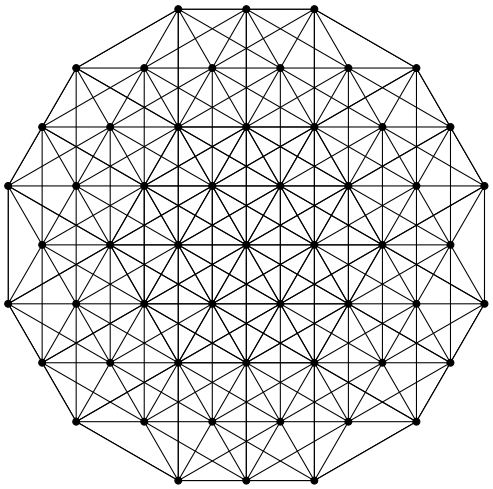


Figure 2 The extremal subgraph of Λ_U with $24k^2 - 24k + 7$ vertices ($k = 2$)

▶ **Theorem 2.** *The EIP for $\mathbb{Z}^d_U$, where U is the generating set $\{\pm e_i : i = 1, \dots, d\} \cup \{\pm 2e_1\}$ of $\mathbb{Z}^d$, does not have nested solutions starting at any size. In other words, for all n and each n-element subset S_n of $\mathbb{Z}^d$ for which $\partial(S_n)$ is the minimum among all n-element subsets of $\mathbb{Z}^d_U$, there does not exist a sequence $S_n \subset S_{n+1} \subset S_{n+2} \subset \cdots$ of i-element subsets S_i of $\mathbb{Z}^d$, such that for each $i \geq n$, $\partial(S_i)$ is the minimum among all i-element subsets of $\mathbb{Z}^d_U$.*

Our second result is a solution of the EIP for another Cayley graph on $\mathbb{Z}^2$ with nested solutions. The generating set for this graph is $U = \{(\pm 1, 0), (0, \pm 1), \pm(1, 1), \pm(1, -1), \pm(-1, 2), \pm(-2, 1)\}$. Thus, it contains $(\mathbb{Z}^2, \ell_\infty)$ as a subgraph. In fact, it is more suitable to consider this to be the graph on the triangular lattice with edges for all pairs at distance 1 or $\sqrt{3}$. As a Cayley graph on Λ, the generating set is depicted in Figure 1. We denote it by Λ_U.

▶ **Theorem 3.** *Let Λ_U be the undirected Cayley graph with vertex set Λ and symmetric generating set $U = \{\pm g_1, \pm(g_1 + g_2), \pm g_2, \pm(2g_2 - g_1), \pm(g_2 - g_1), \pm(g_2 - 2g_1)\}$, where $g_1 = (1, 0)$ and $g_2 = (1/2, \sqrt{3}/2)$. The maximum number of edges of a subgraph of Λ_U with $n \geq 3$ vertices is*

$$e(n) := \begin{cases} 6n - 4\sqrt{6n - 6} & \text{if } n = 24k^2 - 24k + 7 \text{ for some } k \in \mathbb{N} \\ \lfloor 6n - \sqrt{96n - 63} \rfloor & \text{otherwise.} \end{cases}$$

Additionally, Λ_U has nested solutions.

In Figure 2 we depict the unique (up to translation) extremal subgraph of Λ_U with 55 vertices. The subgraphs of Λ_U with n vertices and $e(n)$ edges are candidate extremal graphs for a problem of Erdős and Vesztergombi [7] on the maximum number of occurrences of the smallest and second smallest distances in a set of n points in the plane. Let S be a set of n points in the plane, and denote by $m_1(S)$ and $m_2(S)$ the number of occurrences of the smallest and second smallest distance in S. Let $f(n)$ be the maximum value of $m_1(S) + m_2(S)$, where the maximum is taken over all sets S of n points. It is known that $f(n) \leq 6n$ [15]. (See also [7] for further results.) Theorem 3 implies that $f(n) \geq e(n)$, with the lower bound given by subsets of the triangular lattice, with smallest distance 1 and second smallest distance $\sqrt{3}$.

▶ **Conjecture 4.** *For any sufficiently large n, $f(n) = e(n)$, with the only sets S of n points attaining $f(n) = m_1(S) + m_2(S)$ being similar to the extremal sets on the triangular lattice.*

References

[1] Ben Barber and Joshua Erde, *Isoperimetry in integer lattices*, Discrete Anal. (2018), Paper No. 7, 16. MR3819052

[2] Ben Barber, Joshua Erde, Peter Keevash, and Alexander Roberts, *Isoperimetric stability in lattices*, Proc. Amer. Math. Soc. **151** (2023), no. 12, 5021–5029. MR4648905

[3] A. J. Bernstein, *Maximally connected arrays on the n-cube*, SIAM J. Appl. Math. **15** (1967), 1485–1489. MR223260

[4] Béla Bollobás and Imre Leader, *Edge-isoperimetric inequalities in the grid*, Combinatorica **11** (1991), no. 4, 299–314. MR1137765

[5] Peter Brass, *Erdős distance problems in normed spaces*, Comput. Geom. **6** (1996), no. 4, 195–214. MR1392310

[6] Joseph Briggs and Chris Wells, *Phase transitions in isoperimetric problems on the integers*, 2024.

[7] György Csizmadia, *The multiplicity of the two smallest distances among points*, Discrete Math. **194** (1999), no. 1-3, 67–86. MR1654972

[8] Frank Harary and Heiko Harborth, *Extremal animals*, J. Combin. Inform. System Sci. **1** (1976), no. 1, 1–8. MR457263

[9] Heiko Harborth, *Solution to problem 664 A*, Elem. Math **29** (1974), 14–15.

[10] L. H. Harper, *Optimal assignments of numbers to vertices*, J. Soc. Indust. Appl. Math. **12** (1964), 131–135. MR162737

[11] ———, *The edge-isoperimetric problem for regular planar tesselations*, Ars Combin. **61** (2001), 47–63. MR1863367

[12] ———, *Global methods for combinatorial isoperimetric problems*, Cambridge Studies in Advanced Mathematics, vol. 90, Cambridge University Press, Cambridge, 2004. MR2035509

[13] Sergiu Hart, *A note on the edges of the n-cube*, Discrete Math. **14** (1976), no. 2, 157–163. MR396293

[14] John H. Lindsey II, *Assignment of numbers to vertices*, Amer. Math. Monthly **71** (1964), 508–516. MR168489

[15] K. Vesztergombi, *Bounds on the number of small distances in a finite planar set*, Studia Sci. Math. Hungar. **22** (1987), no. 1-4, 95–101. MR913897

Topological simplification guided by the depth poset

Bartosz Furmanek
Department of Computational Mathematics, Jagiellonian University, Profesora Stanisława Łojasiewicza 6, 30-348 Kraków, Poland

Jakub Leśkiewicz
Department of Computational Mathematics, Jagiellonian University, Profesora Stanisława Łojasiewicza 6, 30-348 Kraków, Poland

Michał Lipiński
Intitute of Science and Technology Austria, Am Campus 1, 3400 Klosterneuburg, Austria,
Second affiliation, Department of Computational Mathematics, Jagiellonian University, Poland

Abstract

Topological simplification is the process of reducing complexity of the structure of a topological space while maintaining the essential features. It may especially be seen as reordering cells of a complex, in a way, which eliminates some persistent homology groups, without affecting the rest. There are multiple solutions for 2-dimensional complexes, however, for the general case the problem becomes more complicated. We present a new approach based on the concept of *depth posets*.

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

Keywords and phrases Topological data analysis, Topological simplification, Discrete Morse Theory

Funding *Bartosz Furmanek*: The research was funded by the program Excellence Initiative – Research University at the Jagiellonian University in Kraków
Jakub Leśkiewicz: The research was funded by the program Excellence Initiative – Research University at the Jagiellonian University in Kraków
Michał Lipiński: This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie Grant Agreement No. 101034413.

Acknowledgements The authors would like to thank Mateusz Przybylski for helpful input during early conversations of this work.

1 Introduction

Let us first introduce the main concepts.

▶ **Definition 1.** *A* Lefschetz complex *is a triplet* $(X, \dim, \Delta)$*, where* X *is a finite set of elements called cells,* $\dim : X \to \mathbb{Z}$ *is a map assigning a dimension to each cell, and* $\Delta : X \times X \to \mathbb{Z}_2$ *is the boundary coefficient such that* $\sum_{z \in X} \Delta(y, z) \cdot \Delta(z, x) = 0$ *for all* $x, y \in X$ *and* $\Delta(y, x) \neq 0$ *only if* $\dim y = \dim x + 1$*. Whenever* $\Delta(y, x) = 1$*, we say that* x *is a* facet *of* y*. A well-known example of a Lefschetz complex is a simplicial complex. When it does not lead to confusion, we will shorten the notation and refer to* X *as the Lefschetz complex.*

▶ **Definition 2.** *Let* $f : X \to \mathbb{R}$ *be a filter on a Lefschetz complex* X*. A pair* $(s, t) \in X \times X$ *is* shallow *if* $f(s)$ *is the maximum among facets of* t *and* $f(t)$ *is the minimum among cofacets of* s *in the filter. We write* $SH(X, f)$ *for the set of shallow pairs. Every shallow pair is a birth-death pair, but not every birth-death pair is shallow.*

We consider the following, slightly simplified variant of a discrete Morse function [2].

▶ **Definition 3.** *Let X be a Lefschetz complex. A map $h : X \to \mathbb{R}$ is called a* discrete Morse function *(dMf, for short) if it fulfils the following conditions for all $x, y \in X$:*

1. *if $\Delta(y, x) = 1$ then $h(x) \leq h(y)$ (weak monotonicity),*
2. *if $h(x) = h(y)$ then either $\Delta(y, x) = 1$ or $\Delta(x, y) = 1$ (pairing),*
3. *the preimage of any singleton under h contains at most two elements (almost injective).*

A dMf h induces a combinatorial Forman gradient vector field (later referred to as cvf), denoted by V_h (see [5]), consisting of pairs of cells with the same value (called vectors) and the remaining singletons (called critical cells). The above definition allows us to interpret h also as a filter on X, and therefore, takes advantage of both, the persistence and dynamical perspectives. By $\mathrm{BD}(X, h)$, we will denote the family of off-diagonal persistence birth-death pairs of Lefschetz complex X induced by h.

One of the problems in *topological simplification* (see [3]), is reordering cells to eliminate certain persistent groups with a minimal number of side effects. It is especially interesting to study such simplifications that leave $\mathrm{BD}(X, h)$ unchanged, except for the removed one.

In [1], the authors propose a solution for this problem based on Forman's vector fields. They noticed, that by a proper interpretation of a filter as a dMf, we may identify elements of pairs in $\mathrm{BD}(X, h)$ as critical points of a combinatorial gradient. This perspective allows us to utilize the classical combinatorial dynamic theorem:

▶ **Theorem 4** ([6]). *Let σ and τ be two critical cells of a gradient vector field V, with exactly one path ρ from τ to σ. Then there is a gradient vector field $V^{-\rho}$ obtained by reversing V along the V-path ρ. The critical cells of V are exactly the critical cells of V apart from $\{\sigma, \tau\}$. Moreover, $V^{-\rho} = V$ except along the path ρ .*

However, the theorem alone gives no guarantee that elimination of a pair of critical cells will not affect the remaining pairs in $\mathrm{BD}(X, h)$. The authors of [1] proposed a special partial order on pairs in $\mathrm{BD}(X, h)$ indicating which critical cells are "safe" to be removed. However, the order was provided only for 0 and 1-dimensional persistence pairs. In our work we apply similar idea, but using a partial order on $\mathrm{BD}(X, h)$ called the *depth poset* [4], which is well defined in any dimension.

2 Main results

▶ **Definition 5.** *Let $(s, t) \in X \times X$ be a pair in a Lefschetz complex (X, dim, Δ) such that s is facet of t. A* cancellation *of (s, t) is the operation producing the new Lefschetz complex (X', dim', Δ'), called the* quotient*, such that $X' = X \setminus \{s, t\}$, $\dim'$ is a restriction of* dim *to X' and $\Delta'(y, x) := \Delta(y, x) + \Delta(y, s) \cdot \Delta(t, x)$. In particular, the homology of X and X' are the same.*

Note, that cancellation of a shallow pair may create new shallow pairs in X'. However, the following theorem guarantees that, apart from the cancelled one, none of them disappears.

▶ **Theorem 6** ([4]). *Let h be dMf and let X be the Lefschetz complex filtered by h. Fix $(s, t) \in SH(X, h)$ and let h' be the dMf on the quotient after canceling (s,t). Then $SH(X', h') \supseteq SH(X, h) \setminus \{(s, t)\}$ and $BD(X', h') = BD(X, h) \setminus \{(s, t)\}$.*

The key idea is to consider the Morse complex of cvf V_h, denoted as $\mathcal{M}(V_h)$, as another Lefschetz complex. We can construct an injective filter h' on $\mathcal{M}(V_h)$ by restricting dMf h to critical cells of V_h. In particular, we have $SH(\mathcal{M}(V_h), h') \subset \mathrm{BD}(\mathcal{M}(V_h), h') = \mathrm{BD}(X, h)$. We notice that, given that there exists exactly one path between the cancelled elements, the

persistence-aware cancellation of a shallow pair in the Morse complex $\mathcal{M}(V_h)$ provided by Theorem 6 can be translated back to the original space X.

▶ **Theorem 7.** *Let V_h be a cvf on a Lefschetz complex X filtered by dMf h and ρ be a unique path between critical points t and s. Then, for every x, y, cells from $\mathcal{M}(V_h^{-\rho})$, we have:*

$$\Delta_{\mathcal{M}(V_h^{-\rho})}(y,x) = \Delta_{\mathcal{M}(V_h)}(y,x) + \Delta_{\mathcal{M}(V_h)}(y,s) \cdot \Delta_{\mathcal{M}(V_h)}(t,x).$$

▶ **Definition 8.** *A linear order on the birth-death pairs is a* shallow order *if, when performing cancellations according to this order, each cancelled pair is guaranteed to be shallow at the current stage. The* depth poset *is the intersection of all possible shallow orders.*

Thus, by combining Theorems 4 and 7, we can use the structure of the depth poset (reflecting all possible shallow orders) at the level of the Morse complex to systematically simplify the dMf h on X in a controlled way, by removing only a single persistence pair from $\mathrm{BD}(X,h)$ at a time. This process is illustrated by a diagram in Figure 1, while a specific example of topological simplification is shown in Figure 2.

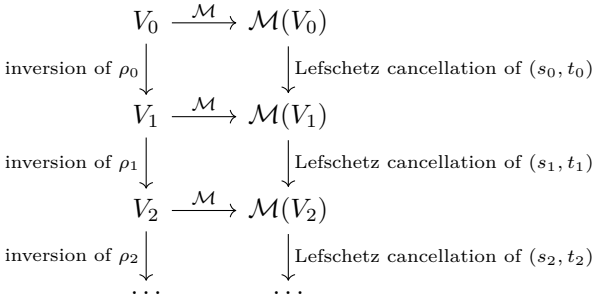


Figure 1 A schematic depiction of the proposed procedure.

Since the depth poset can be calculated in any dimension, this method is completely independent of the dimension of X.

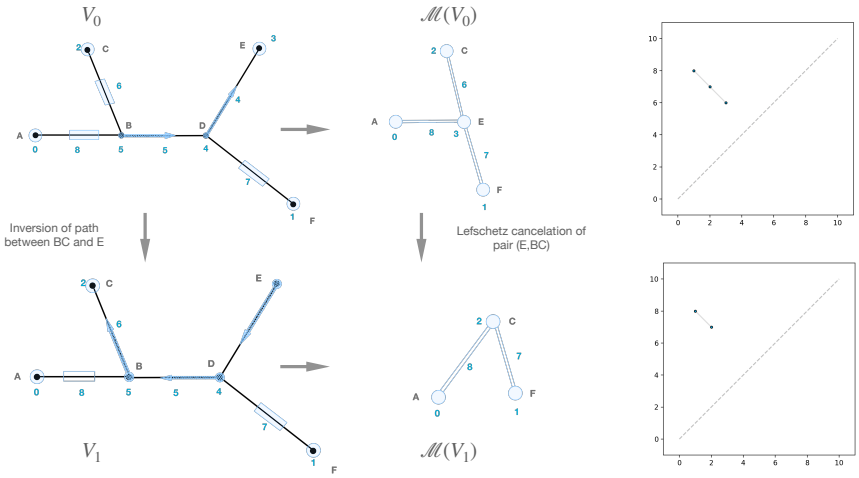


Figure 2 An example of topological simplification.

References

1 Ulrich Bauer, Carsten Lange, and Max Wardetzky. Optimal topological simplification of discrete functions on surfaces. *Discrete & Computational Geometry*, 47(2):347–377, 2012.

2 Bruno Benedetti. Discrete morse theory for manifolds with boundary. *Transactions of American Mathematical Society*, 364(12), 2012.

3 Herbert Edelsbrunner, David Letscher, and Afra Zomorodian. Topological persistence and simplification. *Discrete & Computational Geometry*, 28(4):511–533, 2002. doi:10.1007/s00454-002-2885-2.

4 Herbert Edelsbrunner, Michał Lipiński, Marian Mrozek, and Manuel Soriano Trigueros. The poset of cancellations in a filtered complex. arXiv:2311.14364v3, 2024. arXiv:2311.14364.

5 Robin Forman. Morse theory for cell complexes. *Advances in Mathematics*, 134(1):90–145, 1998.

6 Robin Forman. Morse theory for cell complexes. *Advances in Mathematics*, 134(1):90–145, 1998. URL: https://www.sciencedirect.com/science/article/pii/S0001870897916509, doi:10.1006/aima.1997.1650.

Improved Approximation Algorithms for Three-Dimensional Bin Packing

Debajyoti Kar
Department of Computer Science and Automation, Indian Institute of Science, Bengaluru, India

Arindam Khan
Department of Computer Science and Automation, Indian Institute of Science, Bengaluru, India

Malin Rau
Department of Mathematical Sciences, Chalmers University of Technology and University of Gothenburg, SE-412 96 Göteborg Sweden

Abstract

We study three fundamental three-dimensional (3D) geometric packing problems: 3D (Geometric) Bin Packing (3D-BP), 3D Strip Packing (3D-SP), and Minimum Volume Bounding Box (3D-MVBB). The previous best (absolute) approximation for all three problems is by Li and Cheng (SICOMP, 1990), who gave algorithms with approximation ratios of 13, 46/7, and $46/7 + \varepsilon$, respectively, for 3D-BP, 3D-SP, and 3D-MVBB. We provide improved approximation ratios of 6, 6, and $3 + \varepsilon$, respectively, for the three problems, for any constant $\varepsilon > 0$.

For 3D-BP, in the asymptotic regime, Bansal et al. (Math. Oper. Res., 2006) showed that there is no APTAS even when all items have the same height. Caprara (Math. Oper. Res., 2008) gave an asymptotic approximation ratio of $T_\infty^2 + \varepsilon \approx 2.86$, where T_∞ is the well-known Harmonic constant in Bin Packing. We provide an algorithm with an improved asymptotic approximation ratio of $3T_\infty/2 + \varepsilon \approx 2.54$. Further, we show that unlike 3D-BP (and 3D-SP), 3D-MVBB admits an APTAS.

2012 ACM Subject Classification Theory of computation → Packing and covering problems

Keywords and phrases Approximation Algorithms, Geometric Packing, Multidimensional Packing

Related Version *Full Version*: https://arxiv.org/abs/2503.08863

Funding *Debajyoti Kar*: Debajyoti Kar is grateful to the Google PhD Fellowship Program.
Arindam Khan: Arindam Khan's research is supported in part by Google India Research Award, SERB Core Research Grant (CRG/2022/001176) on "Optimization under Intractability and Uncertainty", Ittiam Systems CSR grant, and the Walmart Center for Tech Excellence at IISc (CSR Grant WMGT-23-0001).

1 Introduction

Three-dimensional (3D) packing problems are used to model several practical settings in production and transportation planning – ranging from cargo management, manufacturing, 3D printing and prototyping, to cutting and loading applications. In this work, we study three classical 3D packing problems. In all of these problems, the input is a collection of (rectangular) cuboids (items), each specified by their height, width, and depth. In the 3D Bin Packing (3D-BP) problem, the goal is to output a packing of all the items using the minimum number of bins, where each bin is a unit cube. In the 3D Strip Packing (3D-SP) problem, we are given a three-dimensional strip having a 1×1 square base and unbounded height, and we have to pack all items minimizing the height of the strip. Finally, in the Minimum Volume Bounding Box (3D-MVBB) problem, we seek to obtain a cuboidal box of minimum volume that can accommodate all input items. In all these problems, the items cannot be rotated about any axis, and they must be packed non-overlappingly. Further, we assume that all items and bins/boxes are axis-aligned.

3D-BP and 3D-SP generalize several classical strongly NP-hard problems in scheduling and packing, including (1D) bin packing, multiprocessor scheduling [22], packing squares into squares [14], and packing cubes into cubes [25]. The survey by Ali, Ramos, Carravilla, and Oliveira [1] provides a comprehensive overview of 3D packing. With the recent exponential growth in transportation and shipping, there has been a plethora of work on empirical procedures and heuristics for 3D-BP [13, 16, 24, 9, 30, 8, 19, 23, 27, 26] and 3D-SP [15, 4, 2, 11, 32, 31]. In contrast, the theoretical exploration of 3D packing has been significantly limited due to its inherently complicated nature. All the three considered problems are NP-hard. Since 3D-BP and 3D-SP generalize 2D Bin Packing, they do not admit an APTAS, as 2D Bin Packing has an asymptotic approximation hardness of $1 + 1/2196$ [6]. Furthermore, it is NP-hard to decide if a set of squares can be packed into a single square bin or not [14] – thus giving an absolute approximation hardness of 2 for 3D-BP and 3D-SP.

The work of Li and Cheng [22] gives algorithms with absolute approximation ratios of $13, 46/7$, and $46/7 + \varepsilon$ for 3D-BP, 3D-SP, and 3D-MVBB, respectively. For 3D-BP, [10, 12] gave an asymptotic approximation ratio of $T_\infty^3 + \varepsilon \approx 4.836$, which was improved to $T_\infty^2 + \varepsilon \approx 2.86$ by Caprara [5]. For 3D-SP, a long line of work [22, 21, 29, 28, 18, 3] culminated in an asymptotic $(3/2 + \varepsilon)$-approximation due to Jansen and Prädel [17]. For general cuboids, there has been no progress on the absolute approximation ratios for any of the three problems since 1990 [22] and for asymptotic approximation ratio of 3D-BP since 2008 [5]. In fact, improved approximability of d-dimensional geometric Bin Packing and Strip Packing, for $d > 2$, was listed as *one of the ten major open problems* in the survey on multidimensional packing by Christensen, Khan, Pokutta, and Tetali [7].

2 Our Results

Due to space constraints, we only present a brief overview of our results and defer the details to the full version [20]. First, for 3D-BP, if OPT is sufficiently large, the algorithm of Caprara [5] already guarantees an absolute approximation ratio less than 3. For the other case, we show how a packing in k bins can be transformed into $6k$ *structured* bins; following which, for constant k, it is possible to find such a structured packing efficiently using a variant of the Generalized Assignment Problem – giving us an absolute approximation ratio of 6.

▶ **Theorem 1.** *There exists a polynomial-time 6-approximation algorithm for 3D-BP.*

This directly implies an absolute $(6 + \varepsilon)$-approximation for 3D-SP. With a more careful analysis, we obtain a slightly better approximation ratio.

▶ **Theorem 2.** *There exists a small absolute constant $\rho > 0$, such that for any $\varepsilon > 0$, there is a polynomial-time $(6 - \rho + \varepsilon)$-approximation algorithm for 3D-SP.*

We next show the existence of an APTAS for 3D-SP when we are allowed to use resource augmentation. Using this, we obtain a $(3 + \varepsilon)$-approximation for 3D-MVBB. Furthermore, surprisingly, unlike 3D-BP and 3D-SP, we show that 3D-MVBB admits an APTAS – settling the asymptotic approximability for the problem.

▶ **Theorem 3.** *For any $\varepsilon > 0$, there exists a polynomial-time $(3+\varepsilon)$-approximation algorithm, and an asymptotic polynomial-time approximation scheme for 3D-MVBB.*

Finally, we turn our attention to the asymptotic approximability of 3D-BP. We use *harmonic rounding* of the item heights, wherein any height larger than ε (*tall* items) is rounded up to the nearest larger number of the form $1/q$, for $q \in \mathbb{N}$. Using a result of [17],

there exists a Strip Packing of these rounded items having height at most a $(3/2+\varepsilon)$-factor larger than the optimal height (asymptotically), in which the strip is partitioned into $O_\varepsilon(1)$ cuboids such that the items packed inside each cuboid are *similar*. By increasing the height of the strip by $O_\varepsilon(1)$, it is possible to pack most items inside these cuboids in such a way that no *tall* item is intersected by any horizontal plane lying at an integral height from the base of the strip. The advantage of this is that it is now possible to obtain a 3D-BP solution of the packed items into about $(1+\varepsilon)H$ bins, where H is the height of the strip. Finally, we show that it is possible to pack the remaining items into $O_\varepsilon(1)$ additional bins, giving us the following result.

▶ **Theorem 4.** *For any $\varepsilon > 0$, there exists a polynomial-time algorithm for 3D-BP with an asymptotic approximation ratio $(3T_\infty/2+\varepsilon) \approx 2.54$.*

References

1 Sara Ali, António Galrão Ramos, Maria Antónia Carravilla, and José Fernando Oliveira. Online three-dimensional packing problems: A review of off-line and on-line solution approaches. *Computers & Industrial Engineering*, 168:108–122, 2022.

2 Sam D Allen, Edmund K Burke, and Graham Kendall. A hybrid placement strategy for the three-dimensional strip packing problem. *European Journal of Operational Research*, 209(3):219–227, 2011.

3 Nikhil Bansal, Xin Han, Kazuo Iwama, Maxim Sviridenko, and Guochuan Zhang. A harmonic algorithm for the 3d strip packing problem. *SIAM Journal on Computing*, 42(2):579–592, 2013.

4 Eberhard E Bischoff and Michael D Marriott. A comparative evaluation of heuristics for container loading. *European Journal of Operational Research*, 44(2):267–276, 1990.

5 Alberto Caprara. Packing d-dimensional bins in d stages. *Mathematics of Operations Research*, 33(1):203–215, 2008.

6 Miroslav Chlebík and Janka Chlebíková. Hardness of approximation for orthogonal rectangle packing and covering problems. *Journal of Discrete Algorithms*, 7(3):291–305, 2009.

7 Henrik I Christensen, Arindam Khan, Sebastian Pokutta, and Prasad Tetali. Approximation and online algorithms for multidimensional bin packing: A survey. *Computer Science Review*, 24:63–79, 2017.

8 Teodor Gabriel Crainic, Guido Perboli, and Roberto Tadei. Extreme point-based heuristics for three-dimensional bin packing. *INFORMS Journal on Computing*, 20(3):368–384, 2008.

9 Teodor Gabriel Crainic, Guido Perboli, and Roberto Tadei. Ts2pack: A two-level tabu search for the three-dimensional bin packing problem. *European Journal of Operational Research*, 195(3):744–760, 2009.

10 János Csirik and André Van Vliet. An on-line algorithm for multidimensional bin packing. *Operations Research Letters*, 13(3):149–158, 1993.

11 Thai Ha Duong. *Heuristics approaches for three-dimensional strip packing and multiple carrier transportation plans*. PhD thesis, University of Nottingham, 2015.

12 Leah Epstein and Rob van Stee. Optimal online bounded space multidimensional packing. In *SODA*, pages 214–223, 2004.

13 Oluf Faroe, David Pisinger, and Martin Zachariasen. Guided local search for the three-dimensional bin-packing problem. *INFORMS journal on Computing*, 15(3):267–283, 2003.

14 Carlos E Ferreira, Flavio K Miyazawa, and Yoshiko Wakabayashi. Packing squares into squares. *Pesquisa Operacional*, 19(2):223–237, 1999.

15 John A George and David F Robinson. A heuristic for packing boxes into a container. *Computers & Operations Research*, 7(3):147–156, 1980.

16 Mhand Hifi, Imed Kacem, Stéphane Nègre, and Lei Wu. A linear programming approach for the three-dimensional bin-packing problem. *Electronic Notes in Discrete Mathematics*, 36:993–1000, 2010.

17 Klaus Jansen and Lars Prädel. A new asymptotic approximation algorithm for 3-dimensional strip packing. In *SOFSEM*, pages 327–338, 2014.

18 Klaus Jansen and Roberto Solis-Oba. An asymptotic approximation algorithm for 3 d-strip packing. In *SODA*, pages 143–152, 2006.

19 Zhihong Jin, Takahiro Ito, and Katsuhisa Ohno. The three-dimensional bin packing problem and its practical algorithm. *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, 46(1):60–66, 2003.

20 Debajyoti Kar, Arindam Khan, and Malin Rau. Improved approximation algorithms for three-dimensional bin packing, 2025. URL: `https://arxiv.org/abs/2503.08863`.

21 Keoin Li and Kam Hoi Cheng. Heuristic algorithms for on-line packing in three dimensions. *Journal of Algorithms*, 13(4):589–605, 1992.

22 Keqin Li and Kam-Hoi Cheng. On three-dimensional packing. *SIAM Journal on Computing*, 19(5):847–867, 1990.

23 Xueping Li, Zhaoxia Zhao, and Kaike Zhang. A genetic algorithm for the three-dimensional bin packing problem with heterogeneous bins. In *IIE Annual Conference*, page 2039, 2014.

24 Andrea Lodi, Silvano Martello, and Daniele Vigo. Heuristic algorithms for the three-dimensional bin packing problem. *European Journal of Operational Research*, 141(2):410–420, 2002.

25 Yiping Lu, Danny Z Chen, and Jianzhong Cha. Packing cubes into a cube is np-complete in the strong sense. *Journal of Combinatorial Optimization*, 29:197–215, 2015.

26 Daniel Mack and Andreas Bortfeldt. A heuristic for solving large bin packing problems in two and three dimensions. *Central European Journal of Operations Research*, 20:337–354, 2012.

27 Batoul Mahvash, Anjali Awasthi, and Satyaveer Chauhan. A column generation-based heuristic for the three-dimensional bin packing problem with rotation. *Journal of the Operational Research Society*, 69(1):78–90, 2018.

28 Flavio Keidi Miyazawa and Yoshiko Wakabayashi. An algorithm for the three-dimensional packing problem with asymptotic performance analysis. *Algorithmica*, 18(1):122–144, 1997.

29 Flavio Keidi Miyazawa and Yoshiko Wakabayashi. Packing problems with orthogonal rotations. In *LATIN*, pages 359–368, 2004.

30 Francisco Parreño, Ramón Alvarez-Valdés, José Fernando Oliveira, and José Manuel Tamarit. A hybrid grasp/vnd algorithm for two-and three-dimensional bin packing. *Annals of Operations Research*, 179:203–220, 2010.

31 Tony Wauters, Jannes Verstichel, and Greet Vanden Berghe. An effective shaking procedure for 2d and 3d strip packing problems. *Computers & Operations Research*, 40(11):2662–2669, 2013.

32 Lijun Wei, Wee-Chong Oon, Wenbin Zhu, and Andrew Lim. A reference length approach for the 3d strip packing problem. *European Journal of Operational Research*, 220(1):37–47, 2012.

On the Complexity of Minimising the Moving Distance for Dispersing Objects

Nicolás Honorato-Droguett
Nagoya University, Japan

Kazuhiro Kurita
Nagoya University, Japan

Tesshu Hanaka
Kyushu University, Japan

Hirotaka Ono
Nagoya University, Japan

Abstract

We study Geometric Graph Edit Distance (GGED), a graph-editing model to compute the minimum edit distance of intersection graphs that uses moving objects as an edit operation. We first show an $O(n \log n)$-time algorithm to render a given unit interval graph (i) edgeless, (ii) acyclic and (iii) k-clique-free. We next show that GGED becomes strongly NP-hard when rendering a weighted interval graph (i) edgeless, (ii) acyclic and (iii) k-clique-free. Lastly, we show that minimising the maximum moving distance for rendering a unit disk graph edgeless is NP-hard over the L_1 and L_2 distances.

2012 ACM Subject Classification Theory of computation → Mathematical optimization

Keywords and phrases Intersection graphs, Optimisation, Graph modification

Related Version *Full Version*: https://doi.org/10.48550/arXiv.2502.12903 [11]

Funding *Kazuhiro Kurita*: This work is partially supported by JSPS KAKENHI Grant Numbers JP21K17812, JP22H03549, and JST ACT-X Grant Number JPMJAX2105.
Tesshu Hanaka: This work is partially supported by JSPS KAKENHI Grant Numbers JP21K17707, JP23H04388, JST CRONOS Grant Number JPMJCS24K2,
Hirotaka Ono: This work is partially supported by JSPS KAKENHI Grant Numbers JP20H05967, JP21K19765, JP22H00513, JST CRONOS Grant Number JPMJCS24K2,

1 Introduction

Graph modification is a fundamental topic for addressing graph (dis)similarity, which involves modifying a given graph at the minimum cost by adding or deleting vertices or edges to satisfy a specific non-trivial graph property. The problem of determining this cost is commonly known as *graph modification problem* (GMP) and has applications in various disciplines [2,8,9]. GMPs are often categorised into vertex and edge modification problems, with edit operations restricted to the vertex and edge sets, respectively. When costs are assumed to be uniform, it is known that GMP is NP-hard for a wide range of graph classes and properties [1, 6, 13, 15].

The negative bounds of GMPs motivate alternative formulations for graph editing that consider domain-specific constraints and cost measures. In particular, *geometric intersection graphs* (hereafter intersection graphs) have drawn a lot of attention on this subject (see e.g. [3–5, 14]). Given a collection of geometric objects $\mathcal{S}$, the *intersection graph* $G(\mathcal{S}) = (V, E)$ is a graph such that there exists a one-to-one correspondence between the vertex set V and $\mathcal{S}$, and the edge set E has an edge if and only if two geometric objects intersect. Fundamental graph classes included in this model are interval graphs and disk graphs.

This paper addresses Geometric Graph Edit Distance, a model for modifying intersection graphs from a geometric perspective. In the intersection graph model, one natural edit operation is to move the objects of $\mathcal{S}$. We consider this moving operation as an edit operation and address minimising the cost of moving objects so that the resulting intersection graph is in a specific graph class. The cost is quantified by the total moving distance, which is the sum of the (possibly weighted) distance values assigned to each object in the collection. When the distances are weighted, we say that the intersection graph is *weighted*. We also present the minimax variation of the problem, in which the maximum moving distance is minimised. More precisely, we formulate the two problems as follows.

Problem Statement Let $\mathcal{S}$ be a collection of geometric objects and Π be a graph class. Geometric Graph Edit Distance [10] asks for the minimum total moving distance of the objects in $\mathcal{S}$ so that $G(\mathcal{S})$ is in Π. Given a real $K > 0$ in addition to the above, minimax-Geometric Graph Edit Distance asks to decide whether a graph in Π can be realised by moving objects so that the distance of moving of S is at most K for all $S \in \mathcal{S}$.

2 Results

We describe our results for Geometric Graph Edit Distance. Our results focus mainly on interval graphs, and we deal with the following graph classes: $\Pi_{\texttt{edgeless}} = \{G : G \text{ is edgeless}\}$, $\Pi_{\texttt{acyc}} = \{G : G \text{ is acyclic}\}$ and $\overline{\Pi_{k\texttt{-clique}}} = \{G : G \text{ does not contain a } k\text{-clique}\}$. Due to space restrictions, we limit ourselves to only giving proof overviews of the results. The full version of the paper can be found in [11].

▶ **Theorem 1.** *Given a collection of unit intervals $\mathcal{I}$, Geometric Graph Edit Distance can be solved in $O(n \log n)$ time so that $G(\mathcal{I})$ is in $\Pi_{edgeless}$.*

Proof Overview. Figure 1 illustrates the algorithm. For a given collection $\mathcal{I}$ and a real $s \geq 1$, we give a piecewise-linear convex function $E : \mathcal{I} \times \mathbb{R} \to \mathbb{R}$, $E(\{I_1, \ldots, I_n\}, x) = \sum_{i=1}^{n} f_i(x)$, $\quad f_i(x) = |x - c(I_i) - (n-i)s|$ to disperse intervals by equal distance s, where $c(I)$ is the centre of an interval I and $c(I_{i+1}) \geq c(I_i)$ for all i. An interval collection can be partitioned into subcollections for which E gives an optimal solution. Determining partitions is done in total $O(n \log n)$ time using an unbalanced merge sort approach and minimising E is done in $O(\log n)$ time using binary search on the breakpoints of E. ◀

▶ **Corollary 2.** *Given a collection of unit intervals $\mathcal{I}$, Geometric Graph Edit Distance can be solved in $O(n \log n)$ time so that $G(\mathcal{I})$ is in Π_{acyc} and $G(\mathcal{I})$ is in $\overline{\Pi_{k\text{-}clique}}$.*

▶ **Theorem 3.** *Geometric Graph Edit Distance is strongly* NP*-hard on weighted interval graphs to realise a graph in $\Pi_{edgeless}$.*

Proof Overview. We show a reduction from 3-Partition [7]. Given an instance (A, B, s) of 3-Partition, we construct an interval collection $\mathcal{I}_A$ and show that A can be partitioned into m subsets if and only if $G(\mathcal{I}_A)$ is in $\Pi_{\texttt{edgeless}}$ with at most total moving distance $T = 3Bm^2$. ◀

▶ **Corollary 4.** *Geometric Graph Edit Distance is strongly* NP*-hard on weighted interval graphs to realise a graph in Π_{acyc} and $\overline{\Pi_{k\text{-}clique}}$.*

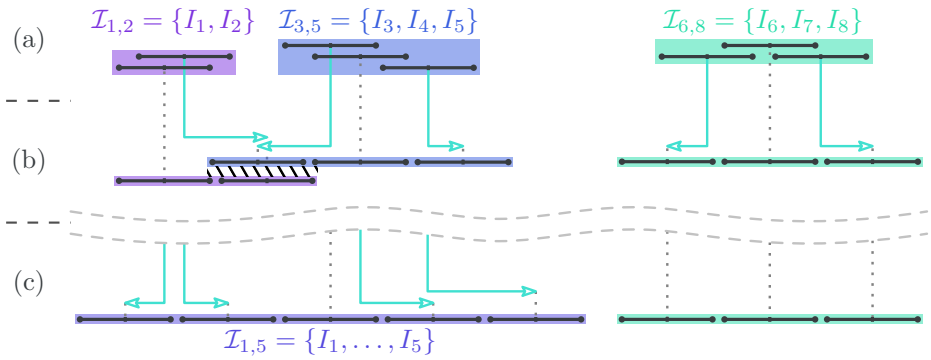


Figure 1 Overview of the algorithm of Theorem 1: (a) The given collection $\mathcal{I}$ is partitioned into three subcollections $\mathcal{I} = \mathcal{I}_{1,2}$, $\mathcal{I}_{3,5}$ and $\mathcal{I}_{6,8}$ for which E gives an optimal solution; (b) The movement is represented by moving each subcollection to its optimal point using E. The subcollections $\mathcal{I}_{1,2}$ and $\mathcal{I}_{3,5}$ intersect; (c) The subcollections $\mathcal{I}_{1,2}$ and $\mathcal{I}_{3,5}$ are merged and then E is minimised for $\mathcal{I}_{1,5}$. The resulting collection is dispersed with minimum total moving distance.

▶ **Theorem 5.** *MINIMAX-GEOMETRIC GRAPH EDIT DISTANCE is strongly* NP*-hard on weighted unit disk graphs and* $\Pi_{\texttt{edgeless}}$ *over the* L_1 *and* L_2 *distances.*

Proof Overview. Reduction from PLANAR 3-SAT [12, 16] (see also Figure 2). We build a disk collection $\mathcal{D}_\Phi$ using the embedding G_Φ of an instance Φ of PLANAR 3-SAT. In particular, for a given instance (Φ, G_Φ) of PLANAR 3-SAT, it can be shown that the minimum maximum moving distance so that $G(\mathcal{D}_\Phi)$ is in $\Pi_{\texttt{edgeless}}$ is at most K if and only if Φ is satisfiable. ◀

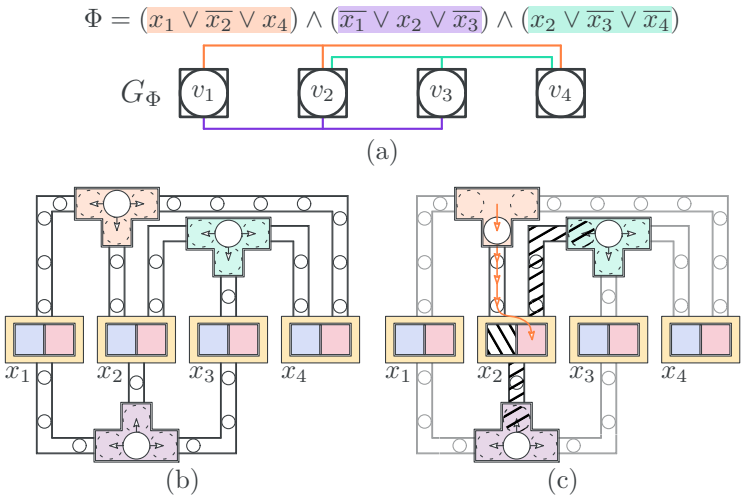


Figure 2 Reduction Overview: (a) An arbitrary instance Φ of PLANAR 3-SAT with its rectilinear embedding G_Φ; (b) The skeleton given by the instance (Φ, G_Φ); (c) The intersection of the gadget for $c = (x_1 \vee \overline{x_2} \vee x_4)$ is removed by moving disks in a way that a free slot of the gadget for x_2 is used. Since c is satisfied when $x_2 = \textit{false}$, the free slots for the other two gadgets become blocked, being unable to remove their intersection using the variable gadget for x_2.

References

1 Pablo Burzyn, Flavia Bonomo, and Guillermo Durán. NP-completeness results for edge modification problems. *Discrete Applied Mathematics*, 154(13):1824–1844, 2006. doi:10.1016/j.dam.2006.03.031.

2 Fan R. K. Chung and David Mumford. Chordal Completions of Planar Graphs. *J. Comb. Theory Ser. B*, 62(1):96–106, 1994. doi:10.1006/jctb.1994.1056.

3 Mark de Berg, Sándor Kisfaludi-Bak, and Gerhard J. Woeginger. The complexity of dominating set in geometric intersection graphs. *Theoretical Computer Science*, 769:18–31, 2019. doi:10.1016/j.tcs.2018.10.007.

4 Fedor V. Fomin, Petr A. Golovach, Tanmay Inamdar, Saket Saurabh, and Meirav Zehavi. Kernelization for spreading points. In *ESA 2023*, volume 274 of *LIPIcs*, pages 48:1–48:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023. doi:10.4230/LIPICS.ESA.2023.48.

5 Fedor V. Fomin, Petr A. Golovach, Tanmay Inamdar, Saket Saurabh, and Meirav Zehavi. Parameterized geometric graph modification with disk scaling. In *ITCS 2025*, volume 325 of *LIPIcs*, pages 51:1–51:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2025. doi:10.4230/LIPICS.ITCS.2025.51.

6 Fedor V. Fomin, Saket Saurabh, and Neeldhara Misra. Graph modification problems: A modern perspective. In *Frontiers in Algorithmics*, pages 3–6. Springer International Publishing, 2015. doi:10.1007/978-3-319-19647-3_1.

7 M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman, 1979.

8 Marc Hellmuth, Manuela Geiß, and Peter F. Stadler. Complexity of modification problems for reciprocal best match graphs. *Theoretical Computer Science*, 809:384–393, 2020. doi:10.1016/j.tcs.2019.12.033.

9 Hung P. Hoang, Stefan Lendl, and Lasse Wulf. Assistance and interdiction problems on interval graphs. *Discrete Applied Mathematics*, 340:153–170, 2023. doi:10.1016/j.dam.2023.06.046.

10 Nicolás Honorato-Droguett, Kazuhiro Kurita, Tesshu Hanaka, and Hirotaka Ono. Algorithms for optimally shifting intervals under intersection graph models. In *IJTCS-FAW 2024*, volume 14752, pages 66–78. Springer, 2024. doi:10.1007/978-981-97-7752-5_5.

11 Nicolás Honorato-Droguett, Kazuhiro Kurita, Tesshu Hanaka, and Hirotaka Ono. On the complexity of minimising the moving distance for dispersing objects, 2025. URL: https://arxiv.org/abs/2502.12903, arXiv:2502.12903.

12 Donald E. Knuth and Arvind Raghunathan. The problem of compatible representatives. *SIAM Journal on Discrete Mathematics*, 5(3):422–427, 1992. doi:10.1137/0405033.

13 John M. Lewis and Mihalis Yannakakis. The node-deletion problem for hereditary properties is NP-Complete. *Journal of Computer and System Sciences*, 20(2):219–230, 1980. doi:10.1016/0022-0000(80)90060-4.

14 Fahad Panolan, Saket Saurabh, and Meirav Zehavi. Contraction decomposition in unit disk graphs and algorithmic applications in parameterized complexity. *ACM Trans. Algorithms*, 20(2):15, 2024. doi:10.1145/3648594.

15 R. Sritharan. Graph modification problem for some classes of graphs. *Journal of Discrete Algorithms*, 38-41:32–37, 2016. doi:10.1016/j.jda.2016.06.003.

16 Craig A. Tovey. A simplified NP-complete satisfiability problem. *Discrete Applied Mathematics*, 8(1):85–89, 1984. doi:10.1016/0166-218x(84)90081-7.

The Reeb Transform

Erin Chambers
Department of Computer Science and Engineering, University of Notre Dame, Indiana, USA

Shankha Shubhra Mukherjee
Department of Computer Science and Engineering, University of Notre Dame, Indiana, USA

Katharine Turner
Mathematical Sciences Institute, Australian National University, Canberra, Australia

Abstract

Reeb Transforms offer a compact representation of how shapes in $\mathbb{R}^n$ change when sliced along different coordinate directions. We define Reeb Transforms as the collection of Reeb graphs obtained by collapsing connected components of each level set in all coordinate directions. In this paper, we develop a rigorous treatment of Reeb Transforms for o-minimal definable sets, emphasizing stratified spaces up to 3-dimensions, embedded in $\mathbb{R}^n$. We establish that for surfaces in $\mathbb{R}^3$, the Reeb Transform is injective, capturing the essential topological features needed to uniquely reconstruct such surfaces. We further show that local modifications, under certain regularity conditions, do not alter the Reeb Transform, thereby demonstrating a form of local stability. However, in dimensions above three, the Reeb Transform ceases to be injective, indicating the limitations of this descriptor in higher-dimensional settings. In addition to these main results, we develop several other properties of Reeb Transforms that underscore their versatility and significance for shape analysis within o-minimal frameworks.

2012 ACM Subject Classification Topology

Keywords and phrases Computational Topology, Algorithms, Reeb Graphs, Directional Transforms

1 Introduction

The present work situates shapes in o-minimal structures for tameness; the definable sets we consider are free from pathological behaviors. This approach follows van den Dries [12] and others who pioneered *tame topology*, ensuring properties such as finiteness of the number of connected components in definable families [6, 10]. In Fujita's work [9], one finds further exposition on how o-minimality restricts the complexity of sets and functions, often permitting finite combinatorial structures such as Reeb graphs to capture an object's global topology.

A crucial aspect of Reeb-based analysis is understanding how level sets evolve. A growing body of recent work [4, 8, 11, 1] formalizes how knowledge of Euler characteristics across all slicing planes in all directions can be sufficient to reconstruct shapes within $Set(d)$ (class of all *compact*, *definable* subsets of $\mathbb{R}^n$ in an o-minimal structure over $\mathbb{R}$). Our paper leverages this perspective to establish injectivity results for the Reeb Transform by relating changes in the Reeb graphs to changes in the connectivity or critical points of the slices.

Within the literature, some authors have focused on transformations (such as isotopies or small perturbations) that preserve the topological features of surfaces [7, 2, 5, 3]. These studies align with the notion that unless critical points of the height function are introduced or removed, the Reeb graphs remain unaltered. Our work expands on this idea by characterizing the conditions under which open sets can be added or removed without affecting the Reeb Transform.

While the Reeb transform is not in general able to distinguish arbitrary shapes, our research indicates it is, in fact, *injective* for certain classes of objects. For example, shapes in low-dimensional settings can be reconstructed from their Reeb transforms under additional

Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

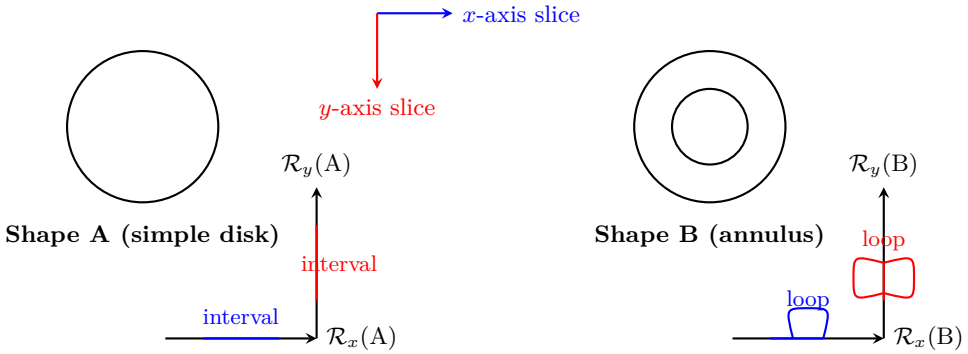


Figure 1 Illustrating injectivity of the Reeb Transform in $Set(2)$. Shape A (a disk) has Reeb graphs along x or y-axes that are simply intervals. Shape B (an annulus) yields Reeb graphs containing loops in each direction. Hence $\mathcal{RT}(A) \neq \mathcal{RT}(B)$.

regularity assumptions. Our work continues this investigation, presenting both positive injectivity results and explicit failure cases in higher-dimensional domains.

2 Results

We briefly summarize our principal statements and define the notation used:

A compact, definable set $A \subset \mathbb{R}^d$ in an o-minimal structure belongs to the class $Set(d)$. In particular, $Set(2)$ denotes the family of such sets in $\mathbb{R}^2$. Likewise, $\mathrm{Surf}(n)$ denotes all compact, definable, smooth manifolds of dimension $(n-1)$ embedded in $\mathbb{R}^n$. For a unit direction $\hat{n} \in S^{d-1}$ and height $t \in \mathbb{R}$, we write $\mathrm{LS}(A, \hat{n}, t) = \{ x \in A \mid \hat{n} \cdot x = t \}$ to denote the level set at height t. The *Reeb Transform* $\mathcal{RT}(A)$ is the collection of Reeb graphs obtained by collapsing connected components of each level set in all coordinate directions. Formally:

$$\mathcal{RT}(A) \;=\; \Big\{ \mathcal{R}_{h_k} \;\Big|\; k = 1, 2, \ldots, n \Big\}.$$

where the *Reeb Graph* $\mathcal{R}_{h_k}$ of h_k is the quotient space obtained by collapsing each connected component of each level set of $h_k|_A$ to a single point. We define $h_k|_A$ as:

$$h_k\big|_A \;:\; A \;\to\; \mathbb{R}, \quad h_k\big|_A(x_1, x_2, \ldots, x_n) = x_k.$$

We define $Set(d, 1)$ as a 1-dimensional stratified space within $Set(d)$.

▶ **Lemma 1.** *The Reeb transform is injective when restricted to* $Set(2)$*.[Fig. 1]*

▶ **Lemma 2.** *Let* $A \in Set(d, 1)$*. Then for any fixed* $\hat{n} \in S^3$ *and height* $t \in \mathbb{R}$*, there exist* $\epsilon_n, \epsilon_t > 0$ *such that for all* $0 < \delta < \epsilon$ *and for all* $v \in S^{d-1}$ *with* $\hat{n} \cdot v \in (1 - \epsilon_t, 1)$*, the Euler characteristic of* $\mathrm{LS}(A, \hat{n}, t)$ *equals that of the corresponding segment of the Reeb graph (in direction* v*) between heights* $t - \delta$ *and* $t + \delta$*.*

▶ **Proposition 1.** *Let* n_k *be the number of vertices in* $\mathcal{RT}(A, v)$ *at height* t *of degree* k*. For* P *the plane with normal* v *and height* t*, the Euler characteristic*

$$\chi(A \cap P) = \sum_v (2 - deg(v))$$

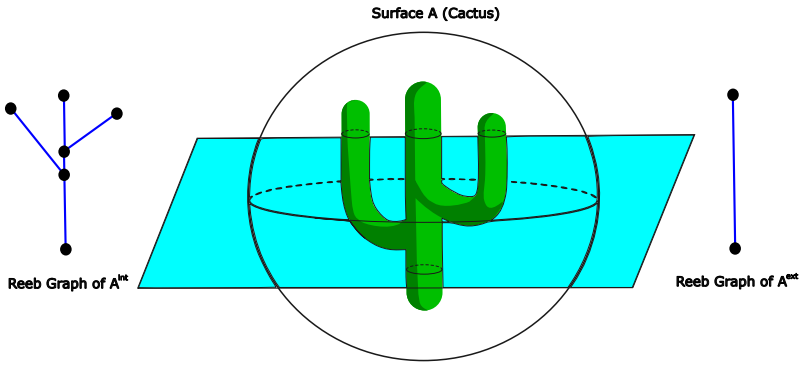


Figure 2 Illustrating Theorem 3. A dividing surface $A \subset \mathbb{R}^3$ splits space into an interior (A^{int}) and exterior (A^{ext}).

We can rewrite this formula in terms of the number of connected components as

$$\chi(A \cap P(\hat{n}, t)) = 2C(A, \hat{n}, t) - C(A, \hat{n}, t - \epsilon) - C(A, \hat{n}, t + \epsilon) \tag{1}$$

where $C(A, \hat{n}, s)$ denotes the number of connected components in $A \cap P(\hat{n}, t)$ and $\epsilon > 0$ is sufficiently small.

▶ **Corollary 2.** *The Reeb transform is injective when restricted to* Surf(3)

▶ **Theorem 3.** *Let $A \in$ Surf(3) be a dividing surface. Let A^{int} denote the A union the interior of A and A^{ext} denote A union the exterior of A. A is encoded by the combination of $\mathcal{RT}(A^{int})$ and $\mathcal{RT}(A^{ext})$. [Fig. 2]*

▶ **Definition 3** (Concave Point). *Let $Y \subset \mathbb{R}^d$ be a set in $Set(d)$ (with no restriction on its intrinsic dimension), and let $y \in Y$. We say that y is a* concave point of Y *if there exist $\epsilon > 0$ and a plane $P(\hat{n}, t) := \{x \in \mathbb{R}^d \mid \hat{n} \cdot x = t\}$ (for some unit vector $\hat{n} \in S^{d-1}$ and real number $t \in \mathbb{R}$) such that:*

$$\big(Y^c \cap B(y, \epsilon)\big) \subseteq \{x \in \mathbb{R}^d \mid \hat{n} \cdot x < t\},$$

where Y^c is the complement of Y in $\mathbb{R}^d$ and $B(y, \epsilon)$ is the open ball of radius ϵ centered at y.

In other words, within some neighborhood around y, the complement of Y lies strictly on one side of the plane $P(\hat{n}, t)$. Equivalently, y is a local minimum of Y in the direction $\hat{n}$.

▶ **Proposition 4.** *Let $A \in Set(d)$ for $d \geq 3$, and let $X \subseteq A$ be an* open, *simply connected subset. Then*

$$\mathcal{RT}(A \setminus X) = \mathcal{RT}(A) \iff \textit{the closure of } X \textit{ contains no concave points of } X.$$

References

1 Erik J. Amezquita, Michelle Y. Quigley, Tim Ophelders, Jacob B. Landis, Daniel Koenig, Elizabeth Munch, and Daniel H. Chitwood. Measuring hidden phenotype: Quantifying the shape of barley seeds using the euler characteristic transform. *In Silico Plants*, 4(1), January 2022. Publisher Copyright: © 2021 The Author(s). Published by Oxford University Press on behalf of the Annals of Botany Company. doi:10.1093/insilicoplants/diab033.

2 S. Biasotti, D. Giorgi, M. Spagnuolo, and B. Falcidieno. Reeb graphs for shape analysis and applications. *Theoretical Computer Science: Computational Algebraic Geometry and Applications*, 392:5–22, 2008. Article 13. URL: http://www.sciencedirect.com/science/article/pii/S0304397507007396, doi:http://dx.doi.org/10.1016/j.tcs.2007.10.018.

3 Silvia Biasotti, Bianca Falcidieno, and Michela Spagnuolo. *Surface Shape Understanding Based on Extended Reeb Graphs*, chapter 6, pages 87–102. John Wiley Sons, Ltd, 2004. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/0470020288.ch6, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/0470020288.ch6, doi:10.1002/0470020288.ch6.

4 Justin Curry, Sayan Mukherjee, and Katharine Turner. How many directions determine a shape and other sufficiency results for two topological transforms. *ArXiv:1805.09782*, 2018. arXiv:1805.09782v1.

5 Tamal K. Dey and Rephael Wenger. Stability of critical points with interval persistence. *Discrete Comput. Geom.*, 38(3):479–512, October 2007.

6 WG Dwyer. Tame homotopy theory. *Topology*, 18(4):321–338, 1979.

7 Herbert Edelsbrunner, John Harer, Ajith Mascarenhas, Valerio Pascucci, and Jack Snoeyink. Time-varying reeb graphs for continuous space–time data. *Computational Geometry*, 41(3):149–166, 2008. URL: https://www.sciencedirect.com/science/article/pii/S0925772107001071, doi:10.1016/j.comgeo.2007.11.001.

8 Brittany Terese Fasy, Samuel Micka, David L. Millman, Anna Schenfisch, and Lucia Williams. Persistence diagrams for efficient simplicial complex reconstruction. *CoRR*, abs/1912.12759, 2019. URL: http://arxiv.org/abs/1912.12759, arXiv:1912.12759.

9 MASATO FUJITA. Locally o-minimal structures with tame topological properties. *The Journal of Symbolic Logic*, 88(1):219–241, October 2021. URL: http://dx.doi.org/10.1017/jsl.2021.80, doi:10.1017/jsl.2021.80.

10 Masato Fujita. Locally o-minimal structures with tame topological properties. *The Journal of Symbolic Logic*, 88(1):219–241, 2023.

11 Robert Ghrist, Rachel Levanger, and Huy Mai. Persistent homology and euler integral transforms, 2018. URL: https://arxiv.org/abs/1804.04740, arXiv:1804.04740.

12 Lou van den Dries. O-minimal structures and real analytic geometry. *Transactions of the American Mathematical Society*, 280(2):795–814, 1983.

Braiding vineyards

Erin Chambers
University of Notre Dame
[South Bend, United States of America]

Mathijs Wintraecken
Inria Sophia Antipolis, Université Côte d'Azur
[Sophia Antipolis, France]

Christopher Fillmore
ISTA (Institute of Science and Technology Austria)
[Klosterneuburg, Austria]

Elizabeth Stephenson
unaffiliated
[Drammen, Norway]

Abstract

Vineyards are a common way to describe persistence diagrams of a data set which is changing, as strong stability means that it is possible to pair points in "nearby" persistence diagrams, yielding a family of point sets which connect into curves when stacked. Recent work has also studied monodromy in the persistent homology transform, demonstrating some interesting connections between an input shape and monodromy in the persistent homology transform for 0-dimensional homology embedded in $\mathbb{R}^2$. In this work, we re-characterize monodromy in terms of periodicity of the associated vineyard of persistence diagrams. We construct a family of objects in any dimension which have non-trivial monodromy for l-persistence of any periodicity and for any l. More generally we prove that any knot or link can appear as a vineyard for a shape in $\mathbb{R}^d$, with $d \geq 3$. This shows an intriguing and, to the best of our knowledge, previously unknown connection between knots and persistence vineyards.

2012 ACM Subject Classification Theory of computation → Computational geometry

Keywords and phrases Monodromy, Knots and links, Persistent homology, Vineyards

Digital Object Identifier 10.4230/LIPIcs...

Related Version *Full version*: `https://arxiv.org/abs/2504.11203` [6]

Funding *Mathijs Wintraecken*: Supported by the Austrian science fund (FWF) grant No. M-3073, the welcome package from IDEX of the Université Côte d'Azur, and the French National Science Agency (ANR) under the StratMesh grant.

Acknowledgements Kate Turner

1 Introduction

This paper explores links between persistent homology [13, 16, 12, 15, 11], in particular the new research direction around monodromy [2], and knot theory [9, 10, 17, 18, 14, 3, 4].

For a continuous one parameter family of filtrations, we can "stack" the persistence diagrams of these filtrations; the resulting object is called a vineyard [8]. Thanks to the stability of persistence, the points in the diagram move continuously (even Lipschitz continuously) with the parameter. This means that we can follow a point in (the stack of) the persistence diagrams; the resulting curve is called a vine. We work with extended persistence [7] in order to retain finite coordinates for vines for each filtration in the family.

Monodromy is the effect where if one makes a loop in a base space of a covering or fibre bundle, the lifted curve may not end up in the same point as you started out with. We say the monodromy is of period $2\pi k$ (with $k > 0$) if the lifted curve returns to the starting point after k revolutions in the base space.

In our context the base space for monodromy is a closed curve or loop, $\gamma : [0, 2\pi] \to \mathbb{R}^d$. A manifold $\mathcal{M}$ (mostly some knot, link, or some offset of a knot or link) is embedded into the same Euclidean space. The fibres are the persistence diagrams of the distance function restricted to the manifold $\mathcal{M}$, that is $d(x, \gamma(t))_{\mathcal{M}}$, for $x \in \mathcal{M}$, see Figure 1. The bundle therefore is the vineyard for t in $[0, 2\pi]$ and identifying persistence diagrams for $t = 0, 2\pi$. The lifted curve is a vine $\tilde{\gamma}_{(b_0,d_0)}(t)$ in the vineyard starting at (b_0, d_0) in the persistence diagram of $d(x, \gamma(0))_{\mathcal{M}}$ and the periodicity is the smallest $k > 0$, such that for all i, $\tilde{\gamma}_{(b_0,d_0)}(0) = \tilde{\gamma}_{(b_i,d_i)}(2\pi k)$ (we also handle the more complicated setting where vines may enter the diagonal in the full paper).

Monodromy was first identified in the context of multidimensional persistence in [5]. In [2], the occurrence of monodromy in the context of the directional persistence transform is studied, more precisely for 0-dimensional persistence modules of objects embedded in $\mathbb{R}^2$. They conclude with several interesting open questions related to the interpretation of monodromy and whether it can be demonstrated in higher dimensions.

2 Our contribution

We answer the latter question by showing that monodromy can occur both in any order homology i, dimension d with $d > i$, and any period $2\pi k$ in a vineyard.

▶ **Theorem 1.** *The vineyard of the distance function in $\mathbb{R}^d$ can exhibit monodromy for persistence up to the $(d-2)$th homology and for extended persistence up to the $(d-1)$th homology. Moreover the periodicity of the monodromy can be $2\pi k$, for any $k \in \mathbb{Z}_{\geq 2}$.*

Additionally, we will prove that for every knot or link there exists an embedded manifold and a family of functions on $\mathcal{M}$ (where each function is induced by the distance to a point in the ambient space, and where in turn each point comes from a curve γ) such that the vineyard of the family of functions yields a braid representing the initial knot or link in the sense of the theorem of Alexander for links [1].

▶ **Theorem 2.** *Given a knot or link, $d, l \in \mathbb{Z}_{>0}$, with, $d \geq 3$ and $l < d - 2$, then there exists an $\mathcal{M} \subset \mathbb{R}^d$ and a closed curve $\gamma \subset \mathbb{R}^d$ such that by identifying the ends of the l-vineyard of $d(x, \gamma(t))_{\mathcal{M}}$, the restriction of the Euclidean distance function to the manifold $\mathcal{M}$, will yield a knot or link, which contains the given knot or link as a subset. That is, it is topologically the initial knot or link after removing some spurious connected components.*

We now provide a brief outline of the construction used in the proofs of both theorems. Thanks to Alexander's theorem [1], a given link can be represented as a closed braid with $\mathcal{I}$ strands. We start with an embedding of the closed braid which lies in a neighbourhood of an annulus in the plane and the braid is planar with the exception of small neighbourhoods of the crossing points, see the first two segments of Firgure 2.

We then modify (if necessary) the braid such that the crossings are equally parsed, by which we mean the angular parameter of the annulus is subdivided into $2\mathcal{I}$ intervals such that each interval together with its opposite interval contain exactly one crossing between them. They are known as crossing and opposing crossing free intervals respectively.

We further modify the embedding of the braid by twisting the annulus 90 degrees in the direction orthogonal to the plane into which the annulus was originally embedded. We define

an observation loop γ to be the curve that follows that twisted annulus on the outside at a constant distance, see the final segment of Firgure 2.

Now for a generic crossing and containing interval, there are two specific strands to consider. These two strands give rise to two or four births in the persistence diagram of $d(\cdot, x)|_{\mathcal{M}}$ for any point in x in the same angular interval of the observation loop. The 0-cycle will die if the strand on which it was created is connected to a strand that was born earlier (a similar statement holds for l-cycles). We can tune the death times of the cycles by pushing the strands, which are non crossing by construction, inward or outward in the opposing non-crossing interval to maintain the appropriate crossings in the vineyard braid to satisfy Theorem 2. Finally, we demonstrate the validity of Theorem 1 by passing the l-dimensional offset of a particular braid through the above construction.

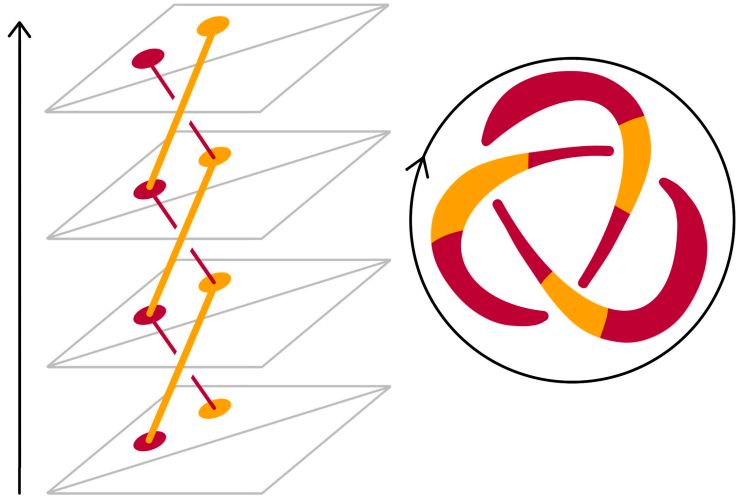


Figure 1 Braided vineyard over a trefoil knot displaying monodromy. The filtration is the distance function from points on the oriented observation loop.

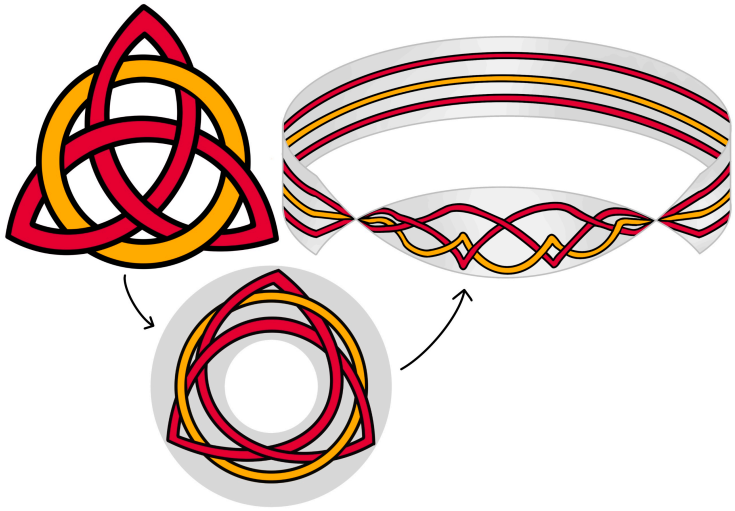


Figure 2 Schematic of the construction used in the proofs of Theorems 1 and 2.

References

1 J. W. Alexander. A lemma on systems of knotted curves. Proceedings of the National Academy of Sciences, 9(3):93–95, March 1923. URL: http://dx.doi.org/10.1073/pnas.9.3.93, doi:10.1073/pnas.9.3.93.

2 Shreya Arya, Barbara Giunti, Abigail Hickok, Lida Kanari, Sarah McGuire, and Katharine Turner. Decomposing the persistent homology transform of star-shaped objects, 2024. URL: https://arxiv.org/abs/2408.14995, doi:10.48550/ARXIV.2408.14995.

3 Joan S Birman and Michael D Hirsch. A new algorithm for recognizing the unknot. Geometry & Topology, 2(1):175–220, 1999.

4 Benjamin A. Burton. The Next 350 Million Knots. In Sergio Cabello and Danny Z. Chen, editors, 36th International Symposium on Computational Geometry (SoCG 2020), volume 164 of Leibniz International Proceedings in Informatics (LIPIcs), pages 25:1–25:17, Dagstuhl, Germany, 2020. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. URL: https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.SoCG.2020.25, doi:10.4230/LIPIcs.SoCG.2020.25.

5 Andrea Cerri, Marc Ethier, and Patrizio Frosini. A study of monodromy in the computation of multidimensional persistence. In Discrete Geometry for Computer Imagery, pages 192–202, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

6 Erin Chambers, Christopher Fillmore, Elizabeth Stephenson, and Mathijs Wintraecken. Braiding vineyards, 2025. arXiv:2504.11203.

7 David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Extending persistence using poincaré and lefschetz duality. Foundations of Computational Mathematics, 9(1):79–103, April 2008. URL: http://dx.doi.org/10.1007/s10208-008-9027-z, doi:10.1007/s10208-008-9027-z.

8 David Cohen-Steiner, Herbert Edelsbrunner, and Dmitriy Morozov. Vines and vineyards by updating persistence in linear time. In Proceedings of the twenty-second annual symposium on Computational geometry, SoCG06, page 119–126. ACM, June 2006. URL: http://dx.doi.org/10.1145/1137856.1137877, doi:10.1145/1137856.1137877.

9 M. Dehn. Über unendliche diskontinuierliche Gruppen. Mathematische Annalen, 71(1):116–144, mar 1911. URL: http://dx.doi.org/10.1007/BF01456932, doi:10.1007/bf01456932.

10 M. Dehn. Transformation der Kurven auf zweiseitigen Flächen. Mathematische Annalen, 72(3):413–421, sep 1912. URL: http://dx.doi.org/10.1007/BF01456725, doi:10.1007/bf01456725.

11 Tamal Krishna Dey and Yusu Wang. Computational Topology for Data Analysis. Cambridge University Press, February 2022. URL: http://dx.doi.org/10.1017/9781009099950, doi:10.1017/9781009099950.

12 Edelsbrunner, Letscher, and Zomorodian. Topological persistence and simplification. Discrete & Computational Geometry, 28(4):511–533, November 2002. URL: http://dx.doi.org/10.1007/s00454-002-2885-2, doi:10.1007/s00454-002-2885-2.

13 Patrizio Frosini. A distance for similarity classes of submanifolds of a euclidean space. Bulletin of the Australian Mathematical Society, 42(3):407–415, December 1990. URL: http://dx.doi.org/10.1017/S0004972700028574, doi:10.1017/s0004972700028574.

14 Joel Hass. Algorithms for recognizing knots and 3-manifolds. arXiv preprint math/9712269, 1997.

15 Steve Oudot. Persistence Theory: From Quiver Representations to Data Analysis. American Mathematical Society, December 2015. URL: http://dx.doi.org/10.1090/surv/209, doi:10.1090/surv/209.

16 Vanessa Robins. Towards computing homology from approximations. Topology Proceedings, 24, 01 1999.

17 J. H. Rubinstein. The solution to the recognition problem for s^3. Lectures in Haifa, Israel, 1992.

18 Abigail Thompson. Thin position and the recognition problem for $\mathbf{S}^3$. Mathematical Research Letters, 1(5):613–630, 1994.

Detecting Toroidal Structure in Data: Implementation and Applications of Persistent Cup-Length

Ekaterina S. Ivshina
SEAS, Harvard University, USA

Galit Anikeeva
Center for Theoretical Physics, MIT, USA

Ling Zhou
Department of Mathematics, Duke University, USA

Abstract

Understanding data structure is key to uncovering hidden patterns and enhancing model expressivity. While persistent homology quantifies holes across scales, recent cohomology-based invariants exploit a richer multiplicative structure. We present the first implementation of persistent cup-length [5], which detects not only the presence but also interaction of holes. We prove our algorithm can identify toroidal structures in data and demonstrate it on grid cells, with applications in ML.

2012 ACM Subject Classification Theory of computation → Computational geometry; Mathematics of computing → Algebraic topology; Mathematics of computing → Topology

Keywords and phrases cohomology, cup product, persistence, cup-length, grid cells

Digital Object Identifier 10.4230/LIPIcs.CVIT.2016.23

Funding *Ekaterina S. Ivshina*: NSF GRFP
Galit Anikeeva: NSF GRFP, Fannie and John Hertz Fellowship

Acknowledgements ESI thanks Jacob Zavatone-Veth, Andrew Yarmola, and Cengiz Pehlevan for helpful discussions. GA acknowledges support from the Fannie and John Hertz Foundation. LZ thanks Jose Perea for valuable discussions.

1 Introduction

Topological Data Analysis (TDA) provides a powerful framework for understanding the "shape" of data by capturing topological features such as connected components, cycles, and voids via persistent homology [9, 10, 15, 18, 4, 8, 2, 3]. However, persistent homology focuses on the additive structure of homology groups and can fail to distinguish spaces with the same Betti numbers but fundamentally different topological structures (e.g., a 2-torus vs. a wedge of two circles and a sphere; see Figure 1).

Recent work has shown that incorporating the cup product from cohomology can overcome these limitations [12, 17, 1, 13, 6, 5, 7, 14]. The cup product encodes interactions between cycles, enabling a *persistent cup-length* invariant that detects nontrivial 2-dimensional structure (e.g., tori) hidden from homology alone [5]. In particular, a cup-length of 2, generated by linearly independent 1-cocycles, indicates the existence of two such cocycles whose cup product forms a nontrivial 2-cocycle. This suggests the presence of a *toroidal component* under suitable conditions. This refined invariant enables us to distinguish certain topologically distinct spaces that share the same homology.

Even though cohomology-based invariants have significant potential, their adoption in TDA has been limited because of computational challenges. We bridge this gap by presenting

42nd Conference on Very Important Topics (CVIT 2016).
Editors: John Q. Open and Joan R. Access; Article No. 23; pp. 23:1–23:4
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

the first implementation and applications of persistent cup-length with polynomial-time guarantees [5]. Further context is provided in the Appendix. Our contributions are:

1. An implementation of a persistent cup-length algorithm for both continuous (Ripser + landmarks) and discrete simplex-wise filtrations. Our code will be available on GitHub.
2. Proof that a persistent cup-length of 2 in a connected closed orientable surface implies the existence of a toroidal structure.
3. Experimental validation on synthetic data and grid cell recordings, confirming that persistent cup-length detects toroidal topology.
4. Discussion of practical implications for neural data, diagnosing recurrent neural network (RNN) dynamics, and understanding quasiperiodicity (see Appendix).

Limitations remaining for future work include computational complexity for large datasets, and sensitivity to the choice of cocycle representatives and the parameters of the filtration. Efficient landmark selection strategies and parameter tuning will be key.

2 Theory

Persistent cup-length detects toroidal structures

We say that a space X has a toroidal component if its cohomology ring contains a subring isomorphic to the cohomology ring of T^2. In the special case where X is a connected, closed, oriented surface, the classification theorem of surfaces implies that if X has a toroidal component, then it is a connected sum of tori. This provides a topological interpretation of the notion of a toroidal component.

Practical considerations: landmark-based subsampling and stability

Computing persistent cohomology on large datasets can be computationally expensive. We mitigate this by employing landmark-based subsampling. Landmarks allow us to approximate the full dataset while preserving key topological features. Further, the stability of persistent cup-length (see [5, Theorem 2]) ensures that our computed persistent cup-length invariant is robust under subsampling.

3 Algorithm

The persistent cup-length algorithm operates in polynomial time [5]. We implemented two versions of the algorithm: one for discrete, simplexwise filtrations and another for continuous Vietoris–Rips filtrations computed using Ripser [16]. Our implementation is summarized in the appendix and addresses multiple bugs in the implementation proposal in [5].

Our implementation uses a distance-matrix representation of the input data. We use Ripser to compute a set of representative cocycles for the persistence barcodes. The main part of our algorithm solves the coboundary condition problem, determining whether a given cocycle is a coboundary at different filtration times. To ensure broad applicability, we incorporate the landmarks feature, reducing the dataset size without significantly compromising the result quality. We validated our implementation by sampling points from a torus and from $S^1 \vee S^2 \vee S^1$ (see Figure 1). The algorithm successfully identifies a cup length of 2 persisting for the torus, but, as predicted by theory, no cup length higher than 1 is observed for $S^1 \vee S^2 \vee S^1$.

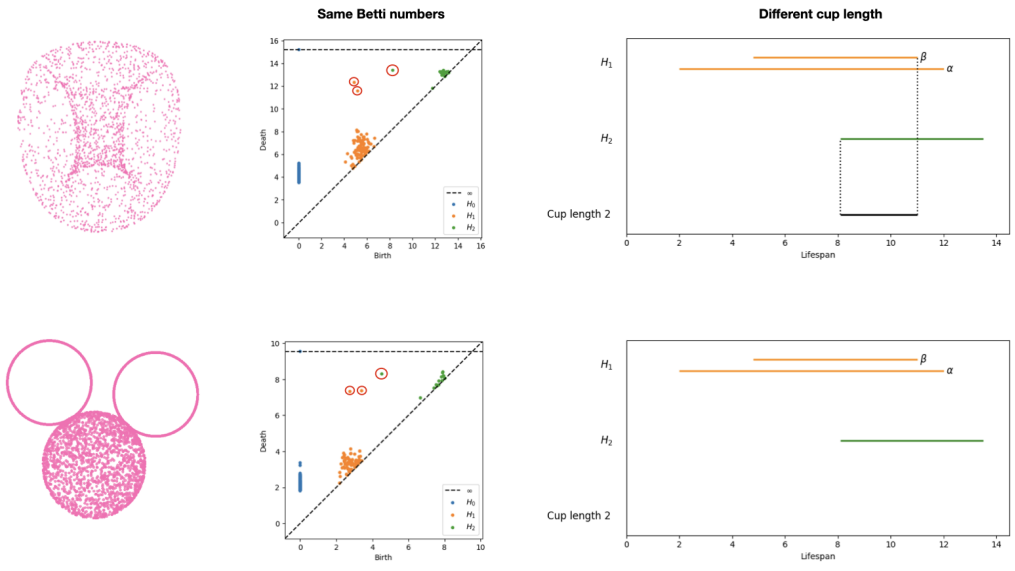


Figure 1 (Top) Torus $S^1 \times S^1$; (Bottom) $S^1 \vee S^2 \vee S^1$: point cloud (left), persistence diagram (middle), and cup-length computation (right). Persistence (via 150 landmarks) reveals two persistent 1D cycles and one 2D void in both datasets, making them indistinguishable by homology alone. Cup-length computation shows the two most persistent 1D bars (orange), the top 2D bar (green), and the cup-length 2 interval (black): cup length 2 is present in the torus, absent in $S^1 \vee S^2 \vee S^1$.

4 Experiments: Grid cells

Recently, [11] used persistent homology to show that activity of grid cells spans a manifold consistent with toroidal topology. However, persistent homology alone does not conclusively prove that the underlying topology is toroidal. By employing our persistent cup-length invariant, we reliably confirm the toroidal topology of grid cell population activity in 17 out of 27 grid modules analyzed in [11] spanning varying behavioral conditions and different environmental contexts. An example analysis of one grid module is shown in Figure 2.

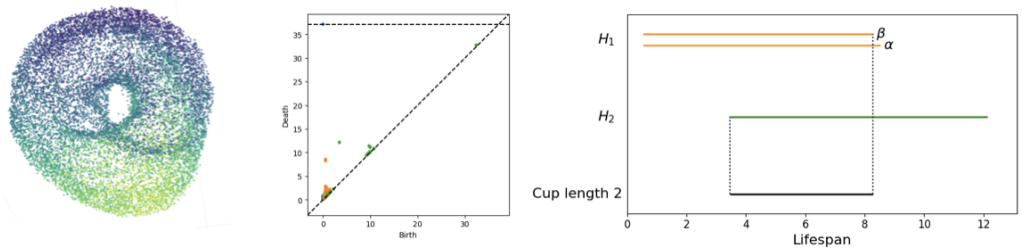


Figure 2 (Left) UMAP-based visualization of grid module activity (149 cells, rat R, day 1, OF, module 2); (Middle) persistence diagram; (Right) cup-length diagram showing top 1D bars (orange), top 2D bar (green), and cup-length 2 interval (black).

References

1 Francisco Belchí and Anastasios Stefanou. A-infinity Persistent Homology Estimates Detailed Topology from Point Cloud Datasets. *Discrete & Computational Geometry*, pages 1–24, 2021. doi:10.1007/s00454-021-00319-y.

2 Gunnar Carlsson. Topology and data. *Bulletin of the American Mathematical Society*, 46(2):255–308, 2009.

3 Gunnar Carlsson. Persistent homology and applied homotopy theory. In *Handbook of Homotopy Theory*, pages 297–329. Chapman and Hall/CRC, 2020.

4 David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of persistence diagrams. *Discrete & computational geometry*, 37(1):103–120, 2007.

5 Marco Contessoto, Facundo Mémoli, Anastasios Stefanou, and Ling Zhou. Persistent cup-length. In *38th International Symposium on Computational Geometry*, volume 224 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 31, 17. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2022. doi:10.4230/lipics.socg.2022.31.

6 Luis G. Polanco Contreras. *Applications of Persistent Cohomology to Dimensionality Reduction and Classification Problems*. PhD thesis, 2022. Thesis (Ph.D.)–Michigan State University. URL: http://gateway.proquest.com/openurl?url_ver=Z39.88-2004&rft_val_fmt=info:ofi/fmt:kev:mtx:dissertation&res_dat=xri:pqm&rft_dat=xri:pqdiss:29206251.

7 Tamal K. Dey and Abhishek Rathod. Cup product persistence and its efficient computation, 2024. URL: https://arxiv.org/abs/2212.01633, arXiv:2212.01633.

8 Herbert Edelsbrunner and John Harer. Persistent homology-a survey. *Contemporary mathematics*, 453:257–282, 2008.

9 Patrizio Frosini. A distance for similarity classes of submanifolds of a Euclidean space. *Bulletin of the Australian Mathematical Society*, 42(3):407–415, 1990.

10 Patrizio Frosini. Measuring shapes by size functions. In *Intelligent Robots and Computer Vision X: Algorithms and Techniques*, volume 1607, pages 122–133. International Society for Optics and Photonics, 1992.

11 Richard J. Gardner, Erik Hermansen, Marius Pachitariu, Yoram Burak, Nils A. Baas, Benjamin A. Dunn, May-Britt Moser, and Edvard I. Moser. Toroidal topology of population activity in grid cells. *Nature*, 602(7895):123–128, January 2022. URL: http://dx.doi.org/10.1038/s41586-021-04268-7, doi:10.1038/s41586-021-04268-7.

12 Jonathan Huang. Cup products in computational topology, 2005. URL: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=04b5ed449bff26197856c8ecb7dd4fbbf93ea846.

13 Umberto Lupo, Anibal M. Medina-Mardones, and Guillaume Tauzin. Persistence Steenrod modules. *J. Appl. Comput. Topol.*, pages 1–28, 2022. doi:10.1007/s41468-022-00093-7.

14 Facundo Mémoli, Anastasios Stefanou, and Ling Zhou. Persistent cup product structures and related invariants. *J. Appl. Comput. Topol.*, 8(1):93–148, 2024. doi:10.1007/s41468-023-00138-5.

15 Vanessa Robins. Towards computing homology from finite approximations. In *Topology proceedings*, volume 24, pages 503–532, 1999.

16 Christopher Tralie, Nathaniel Saul, and Rann Bar-On. Ripser.py: A lean persistent homology library for python. *The Journal of Open Source Software*, 3(29):925, Sep 2018. doi:10.21105/joss.00925.

17 Andrew Yarmola. *Persistence and computation of the cup product*. Undergraduate honors thesis, Stanford University, 2010. URL: https://web.math.princeton.edu/~yarmola/assets/pdf/cup_prod.pdf.

18 Afra Zomorodian and Gunnar Carlsson. Computing persistent homology. In *Proceedings of the twentieth annual symposium on Computational geometry*, pages 347–356, 2004.

Computing the intrinsic Delaunay triangulation of a closed orientable polyhedral surface

Loïc Dubois
LIGM, CNRS, Univ Gustave Eiffel, F-77454 Marne-la-Vallée, France

Abstract

A *triangulation* of a surface is a finite collection of disjoint plane triangles whose sides are partially matched, encoding a compact orientable polyhedral surface. In this paper we consider the *Delaunay* triangulation of a *closed* surface whose vertices are the singularities of the surface (except for flat tori). It is generically unique. We provide an efficient algorithm to compute it from any triangulation of the surface. This allows to pre-process a triangulation before computing shortest paths on its surface.

2012 ACM Subject Classification Theory of Computation → Computational Geometry

Keywords and phrases Polyhedral surfaces, shortest paths, Delaunay triangulations

Digital Object Identifier 10.4230/LIPIcs.CVIT.2016.23

Acknowledgements The author thanks Éric Colin de Verdière and Vincent Despré for discussions that lead to considerable improvements of this article.

1 Introduction

In one of its simplest forms a *triangulation* T is a finite collection of disjoint plane triangles together with a partial matching of the sides of the triangles such that any two matched sides have the same length (Figure 1). This very simple data structure appears under different names in the literature (intrinsic triangulation [3], portalgon [2]). Cutting out the triangles from the plane and identifying the matched sides isometrically, respecting the orientations of the triangles, provides a compact orientable polyhedral surface $\mathcal{S}(T)$. The surface $\mathcal{S}(T)$ is *closed* if, in addition, it is connected and without boundary. Such surfaces can also be obtained from *meshes*, triangles in $\mathbb{R}^3$ glued along their edges. Cutting the edges of a mesh and laying out the resulting triangles in the plane provides a triangulation of its surface. Yet triangulations are more general than meshes: most triangulations cannot be obtained from a mesh this way.

In this paper we consider the *Delaunay* triangulation of a closed surface whose vertices are the singularities of the surface (except for flat tori). It is generically unique. Our contribution is an efficient algorithm to compute it from any triangulation of the surface.

Not all triangulations are suitable for computation. Prominently, shortest path algorithms are affected by the *happiness* [2] of the triangulation, the maximum number of times a shortest path on its surface visits a triangle, which is unbounded, in stark contrast with meshes (whose edges are shortest paths in their surface). Delaunay triangulations have bounded happiness [2], so our algorithm can be used to pre-process a triangulation before computing shortest paths on its surface, answering a problem posed almost 20 years ago in a blog post by Erickson [1], and again in SoCG'23 [2].

We now describe our main result and its proof in more detail.

2 Main result

Replacing triangles by generic polygons in the definition of triangulation gives a *tessellation*. Omitting the case of flat tori here for simplicity, it is classical that on every other closed ori-

42nd Conference on Very Important Topics (CVIT 2016).
Editors: John Q. Open and Joan R. Access; Article No. 23; pp. 23:1–23:3

Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

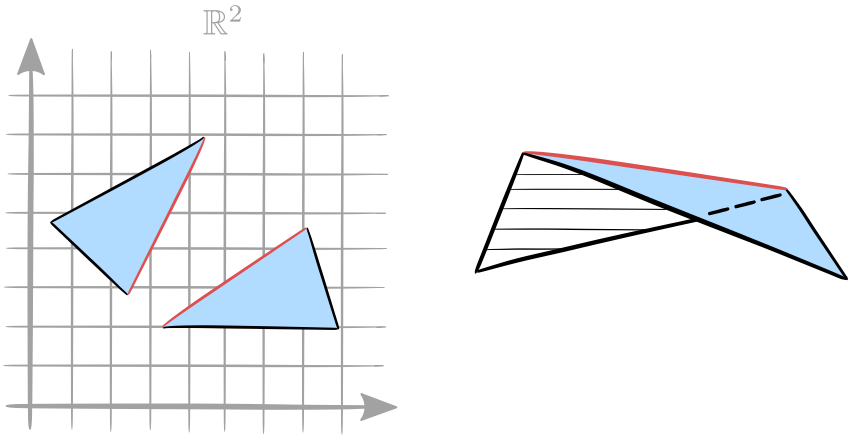


Figure 1 (Left) A triangulation T: two polygons in the plane with two matched sides, in red. (Right) The surface $\mathcal{S}(T)$, and the graph T^1 embedded on $\mathcal{S}(T)$.

entable polyhedral surface S, the Voronoi diagram of the singularities is dual to a tessellation D of S, the *Delaunay tessellation* of S (here the 1-skeleton of the Voronoi diagram is the set of points whose distance to the sources is realized by several shortest paths; in particular the open Voronoi cells are homeomorphic to disks). The Delaunay tessellation is generically a triangulation. If not, triangulating its faces provides the Delaunay triangulation(s).

The (global) aspect ratio of a triangulation T is the greatest side length of a triangle of T divided by the smallest height of a (possibly distinct) triangle of T. Our main result is that if the surface $\mathcal{S}(T)$ is closed, then the Delaunay tessellation of $\mathcal{S}(T)$ can be computed from T in time polynomial in the number n of triangles of T and in the *logarithm* of the aspect ratio of T:

▶ **Theorem 1.** *Let T be a triangulation, with n triangles, of aspect ratio r, whose surface $\mathcal{S}(T)$ is closed. One can compute the Delaunay tessellation of $\mathcal{S}(T)$ in $O^*(n^3 \log^4(r+1))$ time.*

Here $O^*()$ stands for for domination up to a poly-logarithmic factor. We analyze all our results in the real RAM model of computation. The algorithm of Theorem 1 remains polynomial in n and $\log(r+1)$ when r is the *local* aspect ratio of T, the maximum over the triangles t of T of the maximum side length of t divided by the smallest height of t.

3 Overview and techniques for the proof of Theorem 1

On the surface $\mathcal{S}(T)$ of a triangulation T, we consider the graph T^1 corresponding to the sides of the triangles of T. Every edge e of T^1 is a *segment* of T^1, a geodesic relatively disjoint from the singularities of the surface. Adapting the notion of happiness to our needs, we define the *segment-happiness* of e as the maximum number of times it is visited by a shortest path. The segment-happiness of T is the maximum segment-happiness of the edges of T^1. To prove Theorem 1, the crux of the matter is to transform the input triangulation into a triangulation of low segment-happiness.

For that our approach is to focus on triangulations T whose surface $S(T)$ is connected *but may have boundary*, is not simply connected (equivalently, is not homeomorphic to a sphere nor a disk), and more importantly has no positively curved point in its interior (no

singularity surrounded by an angle smaller than 2π). The *systole* of $\mathcal{S}(T)$ is the smallest length of a non-contractible geodesic closed curve in $\mathcal{S}(T)$. Our key technical result is:

▶ **Proposition 2.** *Let T be a triangulation with n triangles, of maximum edge length L. Assume that the surface $\mathcal{S}(T)$ is connected, is not simply connected, and has no positively curved point in its interior. Let $s > 0$ be at most the systole of $\mathcal{S}(T)$. One can compute in $O(n \log^2(n) \log^2(L/s + 2))$ time a triangulation of $\mathcal{S}(T)$ with $O(n \log(L/s + 2))$ triangles, and of segment-happiness $O(\log(n) \log^2(L/s + 2))$.*

The algorithm for Proposition 2 is a finely tuned combination of elementary operations such as inserting and deleting edges and vertices. While the algorithm itself is relatively simple, its analysis is more involved. For that we introduce a new parameter on the segments of a surface, the *enclosure*, possibly of independent interest.

We extend Proposition 2 to surfaces having positively curved points, essentially by cutting out caps around those points. Once we have a triangulation of low segment-happiness, we can compute shortest paths on its surface, adapting the algorithm of [2, Section 3], to construct the Voronoi diagram, and then the Delaunay tessellation. Theorem 1 follows.

References

1 Jeff Erickson. Ernie's 3D Pancakes: Shortest paths on PL surfaces. `https://3dpancakes.typepad.com/ernie/2006/03/shortest_paths_.html`, 2006.

2 Maarten Löffler, Tim Ophelders, Rodrigo I. Silveira, and Frank Staals. Shortest Paths in Portalgons. In *39th International Symposium on Computational Geometry (SoCG 2023)*, volume 258, pages 48:1–48:16, 2023. `doi:10.4230/LIPIcs.SoCG.2023.48`.

3 Nicholas Sharp, Mark Gillespie, and Keenan Crane. Geometry Processing with Intrinsic Triangulations. *SIGGRAPH'21: ACM SIGGRAPH 2021 Courses*, 2021.

Realizing Coordinates of the Second Integral Cohomology Group

Mattie Ji
University of Pennsylvania, USA

Daiyuan Li
University of Pennsylvania, USA

Abstract

It is well known that $H^2(X;\mathbb{Z})$ is isomorphic to the homotopy classes of maps $X \to K(\mathbb{Z},2) = \mathbb{C}P^\infty$. Given X a finite simplicial complex and $[\alpha] \in H^2(X;\mathbb{Z})$, we describe an algorithmic approach to constructing a map of the 3-skeleton X^3, which retains the same second cohomological information as X, into $\mathbb{S}^2 = \mathbb{C}P^1 \subseteq \mathbb{C}P^\infty$ that realizes this isomorphism between cohomology and homotopy classes. From a data-application perspective, this gives a method to represent second-cohomological data spatially. We also provide `Python` code that computes said map restricted to the 2-skeleton X^2.

2012 ACM Subject Classification Mathematics of computing → Algebraic topology; Theory of computation → Computational geometry

Keywords and phrases Computational topology, Eilenberg-Maclane spaces, persistent cohomology

Acknowledgements We want to thank Tanishq Bhatia, Jonathan Block, Richard Schwartz, Kartik Tandon, and Elena Xinyi Wang for helpful discussions and conversations.

1 Introduction

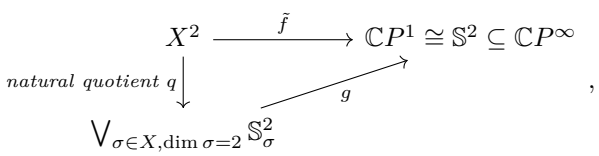
Topological methods, such as persistent homology [4], have been prominent in machine learning [9, 5]. By Brown representability [1], persistent cohomology [2, 8] has the advantage of producing coordinate representations of data with applications in nonlinear dimensionality reductions. However, the spaces associated to higher cohomologies are often difficult to parse.

Let X denote a finite simplicial complex and $\alpha \in C^2(X;\mathbb{Z})$ a 2-cocycle. Brown representability implies that $H^2(X;\mathbb{Z}) \cong [X;\mathbb{C}P^\infty]$. The long exact sequence of cohomology for the pair (X, X^3) shows that the inclusion map $i : X^3 \to X$ is an isomorphism on $H^2(\bullet;\mathbb{Z})$. Thus, we will operate with the heuristic that keeping the 3-skeleton is enough to retain the 2-dimensional topological information of X. In this work, we give an algorithmic outline of how to produce the correspondent map $X^3 \to \mathbb{C}P^1 \subseteq \mathbb{C}P^\infty$. We remark that the methods discussed can be adapted to produce a map $X^{n+1} \to \mathbb{S}^n \subseteq K(\mathbb{Z},n)$ from $\alpha \in C^n(X;\mathbb{Z})$.

2 Defining the Map on the 2-Skeleton

We know the cohomology class $[\alpha]$ corresponds to a map $f : X \to \mathbb{C}P^\infty$. Explicitly, the map f may be realized as follows.

▶ **Proposition 1.** *Suppose $\tilde{f} : X^2 \to \mathbb{C}P^\infty$ such that (i) $\tilde{f}(X^1) = *$, (ii) $\tilde{f}(X^2) \subseteq \mathbb{C}P^1$, and (iii) $\tilde{f}$ factors as maps $g \circ q$ in the diagram:*



42nd Conference on Very Important Topics (CVIT 2016).
Editors: John Q. Open and Joan R. Access; Article No. ; pp. :1– :4
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

and the induced map $g_* : H_2(\bigvee_{\sigma\in X,\dim\sigma=2} \mathbb{S}^2_\sigma;\mathbb{Z}) \to H_2(\mathbb{C}P^\infty;\mathbb{Z}) = \mathbb{Z}$ *sends* $[\mathbb{S}^2_\alpha]$ *to* $\alpha(\sigma) \in \mathbb{Z}$. *Then* $\tilde{f}$ *can be extended to a function* $f : X \to \mathbb{C}P^\infty$ *such that* $f(X^3) \subseteq \mathbb{C}P^1$ *and corresponds to the class* $[\alpha]$.

Such $\tilde{f}$ can always be explicitly constructed by defining the map g, which amounts to explicitly constructing a map $g_\sigma : \mathbb{S}^2_\sigma \to \mathbb{S}^2$ of degree $\alpha(\sigma)$ for each 2-simplex $\sigma \in X$. Since $\tilde{f}$ is defined on X^2 rather than the quotient, we will instead construct maps $\sigma \to \mathbb{S}^2$ that is constant on $\partial\sigma$ and induces the desired map $g_\sigma : \mathbb{S}^2_\sigma \to \mathbb{S}^2$.

One might consider using smooth maps, but there are difficulties in extending this to 3-cells. We use simplicial maps instead to make our analysis more tractable. We fix the codomain to be $\mathbb{S}^2_{\text{min}}$, the four-point minimal triangulation of $\mathbb{S}^2$ (see Figure 1), and subdivide the domain σ to produce a simplicial map into $\mathbb{S}^2_{\text{min}}$. Instead of barycentric subdivision, we use a special case of the edgewise subdivision (see [3]) known as the **midpoint division**.

▶ **Definition 2.** *Let* σ *be a standard 2-simplex of* X*, the* ***m-th subdivision*** $\text{div}(\sigma, m)$ *of* σ *as follows. By convention,* $\text{div}(\sigma, 0) = \sigma$*. We define the* $\text{div}(\sigma, 1)$ *as the subdivision of* σ *by joining the 3 midpoints of the edges of* σ *together (see left of Figure 1). For* $m > 1$*, we define* $\text{div}(\sigma, m)$ *recursively by replacing each 2-simplex of* $\text{div}(\sigma, m-1)$ *with a copy of* $\text{div}(\sigma, 1)$.

Observe that $\text{div}(\sigma, m)$ may be decomposed into four copies of $\text{div}(\sigma, m-1)$, with one copy in the center.

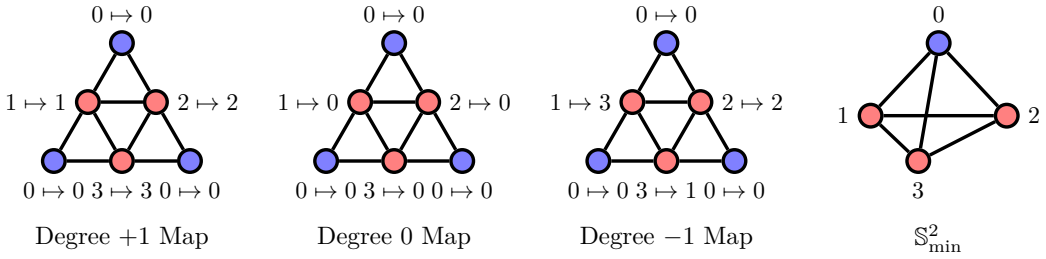


Figure 1 Example realizing degree $-1, 0, +1$ self-maps of $\mathbb{S}^2_{\text{min}}$.

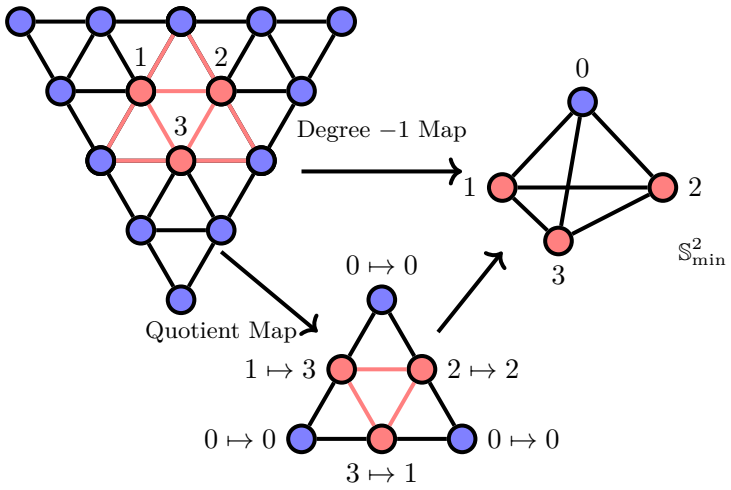


Figure 2 Example realizing a degree -1 map. Every blue-vertex is mapped to $0 \in \mathbb{S}^2_{\text{min}}$.

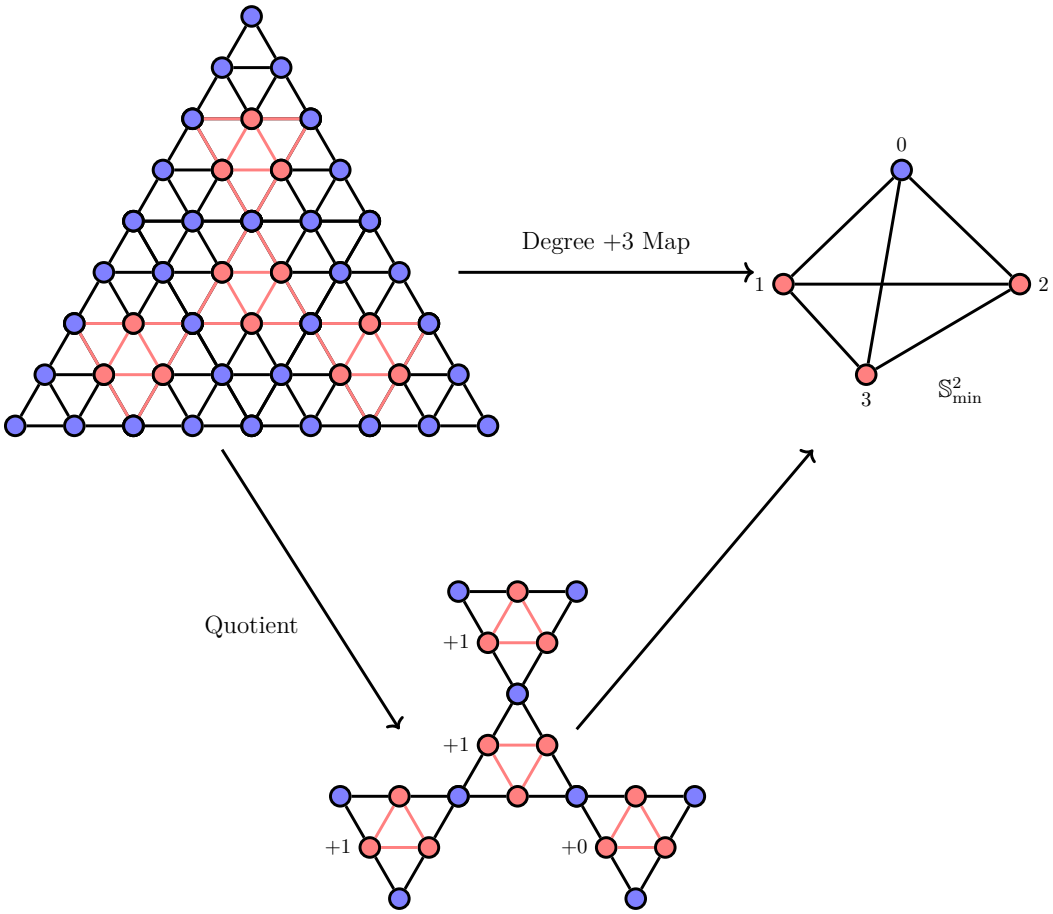


Figure 3 Example realizing a degree 3 map. Every blue-vertex is mapped to $0 \in \mathbb{S}^2_{\min}$.

▶ **Definition 3.** *Let $n(\sigma) = 0$ if $\alpha(\sigma) = 0$ and $n(\sigma) = k+2$ if $\alpha(\sigma) \neq 0$ and k is the smallest number such that $\alpha(\sigma) \leq 4^k$. We define the simplicial map $\Phi_\sigma : \mathrm{div}(\sigma, n(\sigma)) \to \mathbb{S}^2_{min}$.*

1. *If $|\alpha(\sigma)| = 0$, we send all three vertices to 0.*
2. *If $0 < |\alpha(\sigma)| \leq 1$, we sub-divide the 2-simplex σ twice. Recall $\mathrm{div}(\sigma, 2)$ decomposes into four copies of $\mathrm{div}(\sigma, 1)$ and let T_1 be the component in the center. Define the simplicial map on T_1 with degree $\alpha(\sigma)$ according to Figure 1. For every other vertex that has not been assigned a value, send it to $0 \in \mathbb{S}^2_{\min}$. See Figure 2 for an example on the degree -1.*
3. *If $1 < |\alpha(\sigma)| \leq 4$, we sub-divide $\mathrm{div}(\sigma, 2)$ to $\mathrm{div}(\sigma, 3)$. In this case, $\mathrm{div}(\sigma, 3)$ decomposes into four copies of $\mathrm{div}(\sigma, 2)$, that each has their own center T_1, T_2, T_3, T_4 respectively. Define a simplicial map on T_i of degree $d_i \in \{-1, 0, +1\}$ according to Figure 1 such that $\alpha(\sigma) = d_1 + d_2 + d_3 + d_4$. For every other vertex that has not been assigned a value, send it to $0 \in \mathbb{S}^2_{\min}$. See Figure 3 for an example on the degree 3.*
4. *In general, if $4^{k-1} < |\alpha(\sigma)| \leq 4^k$, sub-divide until $\mathrm{div}(\sigma, k+2)$. $\mathrm{div}(\sigma, k+2)$ splits into 4 copies of $\mathrm{div}(\sigma, k+1)$, say S_1, S_2, S_3, S_4. By induction, for any $0 \leq d_i \leq 4^{k-1}$, we can define $\Phi_\sigma|_{S_i}$ such that $\Phi_\sigma|_{\partial S_i} \equiv 1 \in \mathbb{S}^2_{\min}$ and has degree d_i. Choosing $d_1, ..., d_4$ appropriately, the four maps glue together to give a map of degree $\alpha(\sigma) = d_1 + d_2 + d_3 + d_4$.*

▶ **Theorem 4.** *The map $\Phi_\sigma : \mathrm{div}(\sigma, n(\sigma)) \to \mathbb{S}^2_{min}$ induces a map of degree $\alpha(\sigma)$. (Here the induced map is given by quotienting the boundary)*

3 Extending to 3-Simplicies

Let $\tau \in X$ be a 3-simplex, by definition it bounds 2-simplices $F_1, ..., F_4$ of various degrees. Without loss, we can subdivide all four faces $R \geq \max_{i\in\{1,2,3,4\}} n(F_i)$ times and still obtain a simplicial map $\phi : F_1 \cup ... \cup F_4 \to \mathbb{S}^2_{\min}$ of appropriate degrees on the 4 faces. Furthermore, the successive midpoint subdivisions induce a subdivision τ_R of τ (see [3]).

▶ **Theorem 5.** *The map ϕ extends to a simplicial map $\tau_R \to \mathbb{S}^2_{\min}$ if and only if the 1-skeleton $(\tau_R)^1$ has a solution to the following graph coloring question: Consider the vertices of $\mathbb{S}^2_{\min}$ as a coloring set. Does there exist a coloring of $(\tau_R)^1$, extending the coloring of the boundary by ϕ, such that no K_4-subgraphs of $(\tau_R)^1$ have un-repeated colors?*

Since ϕ is by construction null-homotopic (it is constructed out of the values of a 2-cocycle), we know that for R sufficiently large, a solution would exist by the virtue of simplicial approximation theorem (see Chapter 2.C of [6]), which also gives a suitable bound.

4 Implementation

A `Python` implentation of the approach discussed in Section 2 is available at the source code [7]. The graph coloring problem in Theorem 5 is also an interesting direction we plan to investigate in further work.

References

1 Edgar H. Brown. Cohomology theories. *Annals of Mathematics*, 75(3):467–484, 1962.
2 Vin de Silva, Dmitriy Morozov, and Mikael Vejdemo-Johansson. Persistent cohomology and circular coordinates. *Discrete & Computational Geometry*, 45(4):737–759, Jun 2011.
3 H. Edelsbrunner and D. R. Grayson. Edgewise subdivision of a simplex. *Discrete & Computational Geometry*, 24(4):707–719, Jan 2000. doi:10.1007/s004540010063.
4 Herbert Edelsbrunner and John Harer. Persistent homology—a survey. *Contemporary Mathematics*, page 257–282, 2008. doi:10.1090/conm/453/08802.
5 Yam Eitan, Yoav Gelberg, Guy Bar-Shalom, Fabrizio Frasca, Michael Bronstein, and Haggai Maron. Topological blind spots: Understanding and extending topological deep learning through the lens of expressivity, 2024. URL: https://arxiv.org/abs/2408.05486, arXiv:2408.05486.
6 Allen Hatcher. *Algebraic topology*. New York : Cambridge University Press, 2002.
7 Mattie Ji and Daiyuan Li. Source Code for "Realizing Coordinates of the Second Integral Cohomology Group". URL: https://colab.research.google.com/drive/1tLwxYiS14tTwDL7gaISiLKOj0-JjFwjR?usp=sharing.
8 Luis Polanco and Jose A. Perea. Coordinatizing data with lens spaces and persistent cohomology, 2019. URL: https://arxiv.org/abs/1905.00350, arXiv:1905.00350.
9 Chi Seng Pun, Si Xian Lee, and Kelin Xia. Persistent-homology-based machine learning: a survey and a comparative study. *Artificial Intelligence Review*, 55(7):5169–5213, Oct 2022. doi:10.1007/s10462-022-10146-z.

Big polytopes or rich hyperplanes

Koki Furukawa
Department of Mathematics, Kyoto University, Japan

Abstract

We extend the Erdös-Szekeres convex polygon problem to arbitrary point sets in $\mathbb{R}^d$. For $n, l > d \geq 2$, let $\mathrm{ES}_d(l, n)$ be the smallest integer N such that any set of at least N points in $\mathbb{R}^d$ contains either l points contained in a common $(d-1)$-dimensional hyperplane or n points in convex position. We prove that there is a constant $c = c_d > 1$ such that for each $l, n > d \geq 3$,

$$\frac{l-d+2\lfloor \frac{d}{2} \rfloor - 1}{2\lfloor \frac{d}{2} \rfloor} \cdot 2^{c_d n^{\frac{1}{d-1}}} \leq \mathrm{ES}_d(l, n) \leq \frac{l-d}{d!} \cdot 2^{O(\frac{dn}{\log\log\log\log n})}.$$

2012 ACM Subject Classification Theory of computation → Computational geometry

Keywords and phrases Erdös-Szekeres theorem, Convex polytopes, Coplanarity

Related Version A full version of the paper is available at arXiv:2501.03645.

1 Introduction

1.1 Erdös-Szekeres theorem

For $d \geq 2$, we say that a finite point set P in $\mathbb{R}^d$ with $|P| \geq d+1$ is in *general position* if no $(d+1)$ points lie on a common $(d-1)$-dimensional hyperplane. We say that P is in *convex position* if every point of P is a vertex of $conv(P)$ and they are in general position. For $n > d \geq 2$, let $\mathrm{ES}_d(n)$ be the smallest integer N such that any set of at least N points in $\mathbb{R}^d$ in general position contains n points in convex position. In 1935, Erdös and Szekeres [3] proved $\mathrm{ES}_2(n) \leq \binom{2n-4}{n-2} + 1$. In 1960, they [4] showed $\mathrm{ES}_2(n) \geq 2^{n-2} + 1$ and this bound is believed to be optimal. In 2017, Suk [8] gave a great improvement which is $\mathrm{ES}_2(n) \leq 2^{n+O(n^{\frac{2}{3}}\log n)}$. Shortly after, Holmsen et al. [5] showed that $\mathrm{ES}_2(n) \leq 2^{n+O(\sqrt{n\log n})}$ by optimizing Suk's argument. See [1] for a detailed proof. Currently, this is the best known upper bound.

Erdös and Szekeres also noted in their 1935 paper [3], that the number $\mathrm{ES}_d(n)$ is finite for all $n > d \geq 3$. Recently, Pohoata and Zakharov [7] showed that $\mathrm{ES}_d(n) = 2^{o(n)}$ for $d \geq 3$. Their $o(n)$ takes the form $\frac{n}{\log_{(5)} n}$ where $\log_{(k)} n$ is the k-th iterated logarithm function.

On the other direction, Károlyi and Valtr [6] showed that there is a constant $c = c(d) > 1$ such that $\mathrm{ES}_d(n) \geq 2^{cn^{\frac{1}{d-1}}}$ for every $d \geq 2$. Füredi conjectured that $\mathrm{ES}_d(n) = 2^{\Theta(n^{\frac{1}{d-1}})}$.

1.2 Erdös-Szekeres theorem for arbitrary point sets

For $n, l > d \geq 2$, let $\mathrm{ES}_d(l, n)$ be the smallest integer N such that any set of at least N points in $\mathbb{R}^d$ contains either l points contained in a common $(d-1)$-dimensional hyperplane or n points in convex position. In 2024, Conlon et al. [2] gave the upper and lower bounds for the case $d = 2$. that is there exists a constant $C > 0$ such that for each $l, n \geq 3$,

$$(3l-1) \cdot 2^{n-5} < \mathrm{ES}_2(l, n) < l^2 \cdot 2^{n+C\sqrt{n\log n}}.$$

[0] This is an abstract of a presentation given at CG:YRF 2025. It has been made public for the benefit of the community and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear in a conference with formal proceedings and/or in a journal.

We give the upper and lower bounds for $\mathrm{ES}_d(l,n)$. Omitted proofs can be found in the full version on arXiv:2501.03645.

2 Upper bound

Valtr [9] showed that the inequality

$$\mathrm{ES}_d(n) \le \mathrm{ES}_{d-1}(n) \le \cdots \le \mathrm{ES}_2(n) (\le 2^{n+O(\sqrt{n \log n})})$$

holds. The idea of the proof uses a simple projection technique. Consider any set X of at least $\mathrm{ES}_{d-1}(n)$ points in general position in $\mathbb{R}^d$ and its projection Y onto a generic $(d-1)$-dimensional hyperplane. Since $|Y| \ge \mathrm{ES}_{d-1}(n)$, there is a subset Z of size n in convex position. He observed that the set obtained by lifting Z back to former space $\mathbb{R}^d$ is also in convex position. This implies the above inequality. One might think that a similar idea shows $\mathrm{ES}_d(l,n) \le \mathrm{ES}_{d-1}(l,n)$ for $d \ge 2$. However, this is not always true. This is because when projecting a d-dimensional point configuration onto $(d-1)$-dimensional space and then trying to reconstruct the d-dimensional configuration, n-points in convex position in $(d-1)$-dimensions are not always in general position when projected back to d-dimensions. It should be noted once again that, in order for a set of points to be in convex position, it must be in general position. We did not find any proof using projection techniques to establish this type of inequality. Our approach is as follows: Consider a point set X in $\mathbb{R}^d$ which contains no l points on the same $(d-1)$-dimensional hyperplane and evaluate the maximum size of subsets of X in general position. For $d \ge 3$, we obtain the following upper bound for $\mathrm{ES}_d(l,n)$ by using this method.

▶ **Theorem 1.** *For $d \ge 3$,*

$$\mathrm{ES}_d(l,n) \le \frac{l-d}{d!} \cdot 2^{O(\frac{dn}{\log\log\log\log n})}.$$

Using a similar approach for $d = 2$, we obtain the following upper bound:

$$\mathrm{ES}_2(l,n) \le (l-1) \cdot \binom{\mathrm{ES}_2(n)-1}{2} + \mathrm{ES}_2(n) \lesssim l \cdot 4^{n+O(\sqrt{n \log n})}.$$

This bound gives a worse estimate than the upper bound by Conlon et al. [2], unless l is sufficiently large compared to n.

3 Lower bound

As we mentioned above, Kalolyi and Valtr [6] proved $\mathrm{ES}_d(n) \ge 2^{c_d n^{\frac{1}{d-1}}}$ for some $c_d > 1$. Their construction is as follows: Start with one point set X_0, and X_{i+1} is obtained from X_i by replacing each point $x \in X_i$ with two points $x - v(x)$ and $x + v(x)$ where $v(x) = (v^1(x), \ldots, v^d(x))$ is a vector which satisfies $0 < v^1(x) < v^2(x) < \cdots < v^d(x) < \varepsilon_i$ and $v^f(x) < \varepsilon_i v^{f+1}(x)$ for every $1 \le f \le d-1$ and $\varepsilon_i > 0$ be sufficiently small. Then, apply a small perturbation (i.e. taking ε_i sufficiently small) to X_{i+1} to be in general position. By construction, we have $|X_i| = 2^i$. See Figure 1. The key lemma is that the inequality

$$mc(X_{i+1}) \le mc(X_i) + mc(\pi_{d-1}(X_i)) \tag{1}$$

holds where $mc(X) := \max\{|S| : S(\subseteq X) \text{ is in convex position}\}$ and $\pi_{d-1} : \mathbb{R}^d \to \mathbb{R}^{d-1}$ is the projection to the $(d-1)$-dimensional hyperplane $\{x_d = 0\}$. We obtain the following lower bound for $\mathrm{ES}_d(l,n)$ by generalizing their construction.

▶ **Theorem 2.** *There is a constant $c = c_d > 1$ such that*

$$\mathrm{ES}_d(l,n) \geq \frac{l-d+2\lfloor \frac{d}{2} \rfloor - 1}{2\lfloor \frac{d}{2} \rfloor} \cdot 2^{c_d n^{\frac{1}{d-1}}}.$$

Here the order of the lower bound is roughly $\mathrm{ES}_d(l,n) \gtrsim \frac{l-1}{d} \cdot 2^{c_d n^{\frac{1}{d-1}}}$.

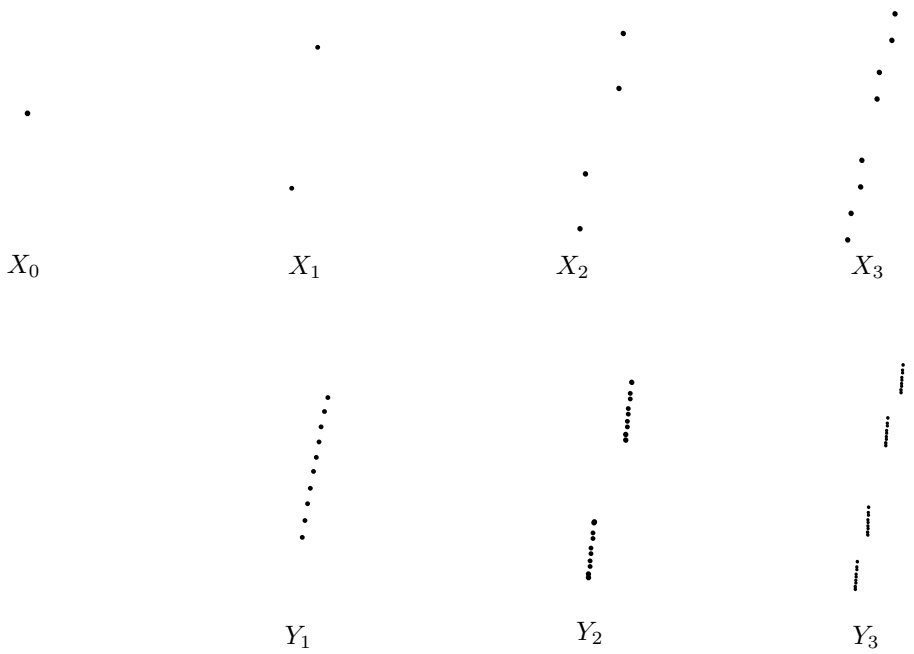


Figure 1

To prove Theorem 2, we consider the set constructed by replacing many points on the line segment $conv(x - v(x), x + v(x))$ for each $i \geq 1$ and each $x \in X_i$. More precisely, for a point $x \in X$, we define the set

$$P_x(\varepsilon, d, l) = \begin{cases} \{x \pm \frac{1}{k} v(x) : k = 1, \ldots, \frac{1}{2}\lfloor \frac{2(l-1)}{d} \rfloor\} & (d : \text{even}, l : \text{odd}) \\ \{x \pm \frac{1}{k} v(x) : k = 1, \ldots, \frac{1}{2}\lfloor \frac{2(l-1)}{d} \rfloor - \frac{1}{2}\} \cup \{x\} & (d : \text{even}, l : \text{even}) \\ \{x \pm \frac{1}{k} v(x) : k = 1, \ldots, \frac{1}{2}\lfloor \frac{2(l-2)}{d-1} \rfloor\} & (d : \text{odd}, l : \text{even}) \\ \{x \pm \frac{1}{k} v(x) : k = 1, \ldots, \frac{1}{2}\lfloor \frac{2(l-3)}{d-1} \rfloor - \frac{1}{2}\} \cup \{x\} & (d : \text{odd}, l : \text{odd}). \end{cases}$$

and define Y_{i+1} as the set obtained by replacing each point x in X_i with $P_x(\varepsilon_i, d, l)$. The size of $P_x(\varepsilon_i, d, l)$ is $\lfloor \frac{2(l-1)}{d} \rfloor$ when d is even, and $\lfloor \frac{2(l-2)}{d-1} \rfloor$ when d is odd.

▶ **Proposition 3.** *For each $i \geq 1$, the set Y_i contains no l points on the same $(d-1)$-dimensional hyperplane.*

Note that if the size of $P_x(\varepsilon_i, d, l)$ were any larger, Proposition 3 would not hold. See Figure 2.

▶ **Lemma 4.** *For each $i \geq 1$, $mc(Y_i) = mc(X_i)$.*

Thus, from (1) and Lemma 4, we obtain $mc(Y_i) \leq 2i^{d-1}$. Finally, simple counting shows that $|Y_i| \geq \frac{l-d+2\lfloor \frac{d}{2} \rfloor - 1}{2\lfloor \frac{d}{2} \rfloor} \cdot 2^i$ and this proves Theorem 2.

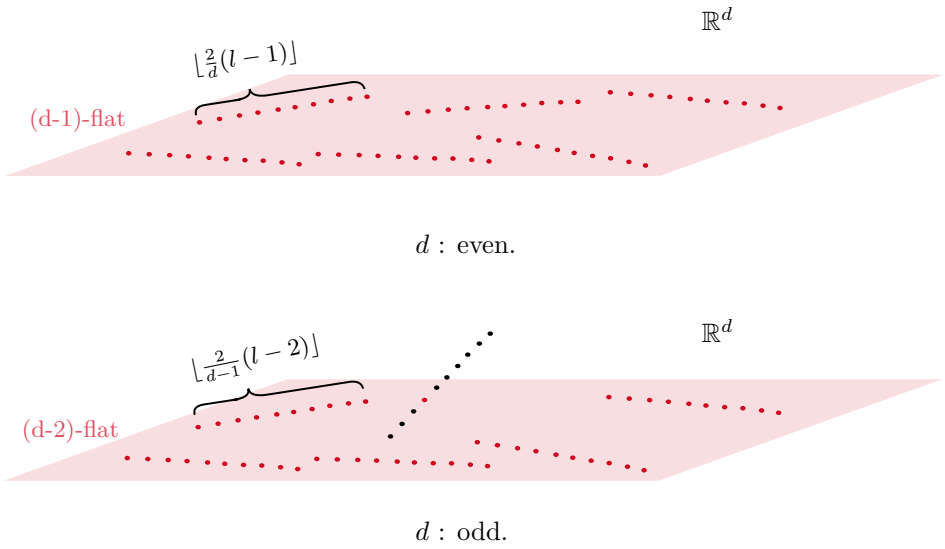


Figure 2

Remark. Károlyi and Valtr also showed that $\mathrm{mc}(X_i) \leq \frac{2}{(d-1)!} i^{d-1} + O(i^{d-2})$ holds. This gives $c_d \approx 2^{e^{-1}d} \approx 2^{0.37d}$. See Appendix in [6]. Although the lower bound in Theorem 2 also holds for $d = 2$, it gives $\mathrm{ES}_2(l, n) \gtrsim (l-1) \cdot 2^{1.67n-1}$, which is asymptotically slightly weaker than the lower bound given by Conlon et al [2].

4 Discussion

We believe that the lower bound in Theorem 2 is optimal for all $d > 2$, except for the exact value of the constant c_d. We are currently working on improving the upper bound and it might be useful to determine the following value $C_d(l, n)$: Let $C_d(l, n)$ be the minimum N such that every set of N-points in $\mathbb{R}^d$ in weakly convex position contains either l points on the same $(d-1)$-dimensional hyperplane or n in convex position where a set of points $P(\subset \mathbb{R}^d)$ is said to be in *weakly convex position* if $P \subseteq \partial(conv(P))$. We believe that $C_d(l, n) \approx \frac{nl}{d}$ and if this is true, then it immediately follows that $\mathrm{ES}_d(l, n) \lesssim \mathrm{ES}_d(\frac{nl}{d})$. However, even if this is possible, it remains asymptotically far from the lower bound.

References

1 I. Bárány, Combinatorial convexity, AMS University Lecture Series 77, (2021).

2 D. Conlon, J. Fox, X. He, D. Mubayi, A. Suk, and J. Verstraete, Big line or big convex polygon, `https://arxiv.org/abs/2405.03455` (2024).

3 P. Erdös and G. Szekeres, A combinatorial problem in geometry, *Compos. Math.* **2**, 463–470, (1935).

4 P. Erdös and G. Szekeres, On some extremum problems in elementary geometry, *Ann. Univ. Sci. Budapest, Eötvös Sect. Math.* **3-4**, 53-62, (1960).

5 A. Holmsen, H. N. Mojarrad, J. Pach, and G. Tardos, Two extensions of the Erdös–Szekeres problem, `https://arxiv.org/abs/1710.11415` (2017).

6 G. Károlyi and P. Valtr, Point configurations in d-space without large subsets in convex position. Discrete Comput. Geom. **30**, 277–286, (2003).

7 C. Pohoata and D. Zakharov, Convex polytopes from fewer points, `https://arxiv.org/abs/2208.04878` (2022).

8 A. Suk, On the Erdös-Szekeres convex polygon problem, *J.Am.Math.Soc.* **30**, 1047-1053, (2017).

9 P. Valtr, Several Results Related to the Erdös–Szekeres Theorem, Doctoral Dissertation, Charles University, Prague, (1996).