

# CG Week Young Researchers Forum 2021

## Booklet of Abstracts

2021

This volume contains the abstracts of papers at “Computational Geometry: Young Researchers Forum” (CG:YRF), part of the Computational Geometry Week (CG Week), held as online conference on June 7-11, 2021.

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

- Mikkel Abrahamsen, University of Copenhagen, Denmark
- Paz Carmi, Ben-Gurion University of the Negev, Israel
- Brittany Terese Fasy, Montana State University, USA
- Eunjin Oh, Pohang University of Science and Technology, South Korea
- Marcel Roeloffzen, Eindhoven University of Technology, Netherlands
- Don Sheehy, North Carolina State University, USA
- Haitao Wang (Chair), Utah State University, USA
- Jie Xue, University of California, Santa Barbara, USA

There were 27 papers submitted to CG:YRF. Of these, 21 were accepted with revisions after a limited refereeing process to ensure some minimal standards and to check for plausibility. The abstracts have been made public for the benefit of the community and should be considered preprints rather than a formally reviewed papers. Thus, these works are expected to appear in conferences with formal proceedings and/or in journals.

Copyrights of the articles in this booklet are maintained by their respective authors. More information about this conference and about previous and future editions is available online at

<http://www.computational-geometry.org/>

# Independent Hyperplanes in Oriented Paving Matroids

Lamar Chidiac 

Department of Discrete Mathematics and Optimization, FernUniversität in Hagen, Germany

<https://www.fernuni-hagen.de/MATHEMATIK/DMO/mitarbeiter/chidiac.html>

lamar.chidiac@fernuni-hagen.de

Winfried Hochstättler 

Department of Discrete Mathematics and Optimization, FernUniversität in Hagen, Germany

[winfried.hochstaettler@fernuni-hagen.de](mailto:winfried.hochstaettler@fernuni-hagen.de)

---

## Abstract

In 1993, Csima and Sawyer [2] proved that in a non-pencil arrangement of  $n$  pseudolines, there are at least  $\frac{6}{13}n$  simple points of intersection. Since pseudoline arrangements are the topological representations of reorientation classes of oriented matroids of rank 3, in this paper, we will use this result to prove by induction that an oriented paving matroid of rank  $r \geq 3$  on  $n$  elements, where  $n \geq 5 + r$ , has at least  $\frac{12}{13(r-1)} \binom{n}{r-2}$  independent hyperplanes, yielding a new necessary condition for a paving matroid to be orientable.

**2012 ACM Subject Classification** Theory of computation Computational geometry; Mathematics of computing Combinatoric problems

**Keywords and phrases** Oriented matroids, paving matroids, simple points, independent hyperplanes.

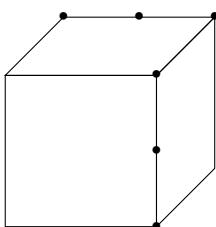
**Digital Object Identifier** 10.4230/LIPIcs.CVIT.2016.23

**Related Version** <https://arxiv.org/abs/2101.12290><sup>1</sup>

## 1 Introduction

The well-known Sylvester-Gallai theorem states that given a set of non-collinear points in the Euclidean plane, we can always find at least one line that has exactly two of the given points. A generalization of this theorem to higher dimension is not always true, i.e. given a finite set of points in a d-dimensional Euclidean space which is not contained in a hyperplane, we cannot always find a hyperplane containing exactly d of the given points, which we call an *independent hyperplane*. A counterexample was given by Hansen in [4](Figure 1).

In his counterexample the main issue lies in the 3-point lines, and we can forbid this by considering a more specific type of point configuration.



**Figure 1** An illustration of Hansen's construction [4]

---

<sup>1</sup> This is an abstract of a presentation given at CG:YRF 2021. 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 consider oriented paving matroids which we will define later. In fact, realizable simple oriented paving matroids of rank 4 are exactly the point configurations in 3-space that have no 3-point lines. By considering this type of matroids we can not only prove the existence of an independent hyperplane but also examine their number. To do so we will have to go back to the 2-dimensional space. The dualization of Sylvester-Gallai's theorem to the projective plane states: For every finite set of lines, not all going through one point (non-pencil arrangement), then among all the intersection points of these lines, at least one is incident with exactly two of the lines. We call it a *simple point*. The proofs of most of the results for line arrangements reveals that the straightening of the lines which form the arrangements plays only a very limited role. This leads naturally to the idea of investigating arrangements of more general types; arrangements of pseudolines.

► **Definition 1.** *An arrangement of pseudolines is a finite collection of simple curves (no self-intersection) in the real projective plane satisfying the following two properties: (a) any two curves intersect in exactly one point, where they cross and (b) the intersection of all curves is empty. As in line arrangements, a simple point in a pseudoline arrangement is the point formed by the intersection of exactly 2 pseudolines (curves).*

Now let's see the main connection between oriented matroids and pseudoline arrangements. We assume basic familiarity with matroid theory and oriented matroids. The standard references are [8, 1]. As mentioned earlier, the main problem with Hansen's counterexample is that the 6 point configuration does not correspond to a paving matroid since we have three points lying on the same line.

► **Definition 2.** *A paving matroid is a matroid in which every circuit has size either  $r$  or  $r + 1$ , where  $r$  is the rank of the matroid. In a point configuration, this means that no  $r - 1$  points lie on a same flat of co-dimension 2.*

Mayhew et al. [7] conjecture that almost all matroids are paving. This is mainly why we find paving matroids to be interesting specially since in rank 3, all simple matroids are paving. The following is immediate:

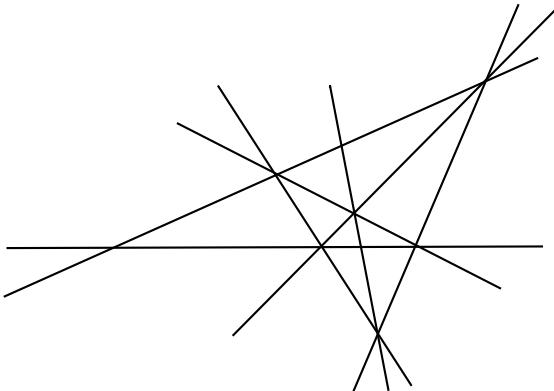
► **Proposition 3.** *The class of paving matroid is closed under minors.*

Oriented matroids and pseudoline arrangements are strongly connected by the topological representation theorem.

► **Theorem 4** (The Topological Representation Theorem [3, 5]). *Any reorientation class of a rank-3 oriented matroid has a representation as a pseudoline arrangement.*

In this representation, the pseudolines are the elements of the oriented matroid. Their intersection points are the hyperplanes of the matroid and their simple points of intersection are exactly the independent hyperplanes in the oriented matroid. Therefore, counting simple points in a pseudoline arrangement is actually counting independent hyperplanes in the oriented matroid. The latest result on the lower bound of simple points in line arrangement was in 1993, where Csima and Sawyer proved in [2] that except for the Kelly-Moser configuration (Figure 2), an arrangement of  $n$  lines has at least  $6n/13$  simple points. In the same paper, they indicated that their proof easily extends to arrangement of pseudolines. This is why we were able to use their result to prove our main result.

► **Theorem 5.** *An oriented paving matroid  $M$  of rank  $r \geq 3$ , on  $n$  elements, where  $n \geq 5 + r$ , has at least  $f(n, r) = \frac{12}{13(r-1)} \binom{n}{r-2}$  independent hyperplanes.*



■ **Figure 2** The Kelly-Moser configuration [6].

## 2 Proof of Theorem 5

We proceed by induction on  $r \geq 3$ . By the Topological Representation Theorem 4, a rank-3 oriented matroid has a representation as a pseudoline arrangement in the projective plane. By Csima and Sawyer [2], except for the Kelly-Moser configuration, a pseudoline arrangement has at least  $6n/13$  simple points. Since  $n \geq 5 + r$  the Kelly-Moser configuration is excluded. In matroid terms, this corresponds to  $6n/13$  independent hyperplanes in a rank 3 oriented matroid founding our induction. Now assume  $r > 3$ . We fix an element  $e \in E$ . By Proposition 3, the contraction  $M/e$  is paving and has at least  $5 + (r - 1)$  elements, so by induction,  $M/e$  has at least  $f(n - 1, r - 1) = \frac{12}{13(r - 2)} \binom{n-1}{r-3}$  independent hyperplanes. Now any independent hyperplane in  $M/e$  can be extended by  $e$  to an independent hyperplane in  $M$ . To prove this, we take an independent hyperplane  $H$  in  $M/e$  and prove that  $H \cup e$  is an independent hyperplane in  $M$ . We have that  $r(H) = r - 2$ , thus  $r(H \cup e) = r - 1 = |H \cup e|$ . Therefore  $H \cup e$  is independent. It is also a flat, because if it wasn't i.e., if there exists an element  $x \in cl(H \cup e) \setminus (H \cup e)$  then  $x \in cl_{M/e}(H) \setminus H$  contradicting,  $H$  being closed in  $M/e$ . Therefore  $H \cup e$  is an independent flat of rank  $r - 1$ , thus an independent hyperplane in  $M$ . Since each of the  $f(n - 1, r - 1)$  independent hyperplanes in  $M/e$  can be extended by  $e$  to obtain an independent hyperplane in  $M$  and since this is the case for any element  $e$  of the matroid, each of these independent hyperplanes will be counted  $r - 1$  times, and so  $M$  has at least  $\frac{n}{r-1} f(n - 1, r - 1) = \frac{12}{13(r - 1)} \binom{n}{r-2} = f(n, r)$  independent hyperplanes. This concludes the proof of Theorem 5.

---

### References

- 1 Anders Björner, Michel Las Vergnas, Bernd Sturmfels, Neil White, and Günter M. Ziegler. *Oriented matroids*, volume 46 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, second edition, 1999. doi:[10.1017/CBO9780511586507](https://doi.org/10.1017/CBO9780511586507).
- 2 J. Csima and E. T. Sawyer. There exist  $6n/13$  ordinary points. *Discrete Comput. Geom.*, 9(2):187–202, 1993. doi:[10.1007/BF02189318](https://doi.org/10.1007/BF02189318).
- 3 Jon Folkman and Jim Lawrence. Oriented matroids. *J. Combin. Theory Ser. B*, 25(2):199–236, 1978. doi:[10.1016/0095-8956\(78\)90039-4](https://doi.org/10.1016/0095-8956(78)90039-4).
- 4 Sten Hansen. A generalization of a theorem of Sylvester on the lines determined by a finite point set. *Math. Scand.*, 16:175–180, 1965. doi:[10.7146/math.scand.a-10758](https://doi.org/10.7146/math.scand.a-10758).

- 5 W. Hochstättler. Oriented matroids from wild spheres. In *Proceedings of the Colloquy “Discrete Optimization—Structure and Stability of Dynamical Systems” (Cologne, 2000)*, volume 7, pages 16–26, 2002. Special Issue.
- 6 L. M. Kelly and W. O. J. Moser. On the number of ordinary lines determined by  $n$  points. *Canadian J. Math.*, 10:210–219, 1958. doi:10.4153/CJM-1958-024-6.
- 7 Dillon Mayhew, Mike Newman, Dominic Welsh, and Geoff Whittle. On the asymptotic proportion of connected matroids. *European J. Combin.*, 32(6):882–890, 2011. doi:10.1016/j.ejc.2011.01.016.
- 8 James Oxley. *Matroid theory*, volume 21 of *Oxford Graduate Texts in Mathematics*. Oxford University Press, Oxford, second edition, 2011. doi:10.1093/acprof:oso/9780198566946.001.0001.

# Plane Matchings in Simple Drawings of Complete Graphs

Oswin Aichholzer 

Graz University of Technology, Austria

Alfredo García 

Departamento de Métodos Estadísticos and IUMA, Universidad de Zaragoza, Spain

Javier Tejel 

Departamento de Métodos Estadísticos and IUMA, Universidad de Zaragoza, Spain

Birgit Vogtenhuber 

Graz University of Technology, Austria

Alexandra Weinberger 

Graz University of Technology, Austria

---

## Abstract

Simple drawings are drawings of graphs in which the edges are Jordan arcs and each pair of edges shares at most one point (a proper crossing or a common endpoint). We show that every simple drawing of the complete graph with  $n$  vertices contains  $\Omega(n^{\frac{1}{2}})$  pairwise disjoint edges. This improves the currently known best lower bound  $\Omega(n^{\frac{1}{2}-\varepsilon})$  for any  $\varepsilon > 0$  by Ruiz-Vargas [8].

**2012 ACM Subject Classification** Mathematics of computing → Combinatorics; Mathematics of computing → Graph theory

**Keywords and phrases** Simple drawings, simple topological graphs, disjoint edges, plane matching

**Digital Object Identifier** 10.4230/LIPIcs.CVIT.2016.23

**Funding** *Oswin Aichholzer*: Supported by the Austrian Science Fund (FWF): W1230.

*Alfredo García*: Supported by H2020-MSCA-RISE project 734922 - CONNECT and Gobierno de Aragón project E41-17R.

*Javier Tejel*: Supported by H2020-MSCA-RISE project 734922 - CONNECT, Gobierno de Aragón project E41-17R and project PID2019-104129GB-I00 / AEI / 10.13039/501100011033 of the Spanish Ministry of Science and Innovation.

*Birgit Vogtenhuber*: Partially supported by Austrian Science Fund within the collaborative DACH project *Arrangements and Drawings* as FWF project I 3340-N35.

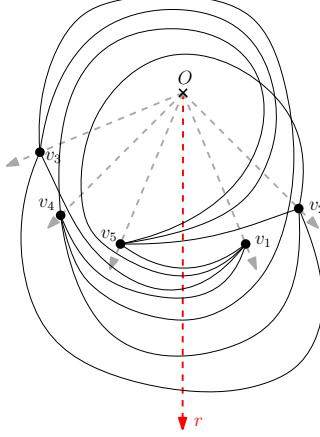
*Alexandra Weinberger*: Supported by the Austrian Science Fund (FWF): W1230.

## 1 Introduction

Simple drawings are drawings of graphs in the plane such that vertices are distinct points in the plane, edges are Jordan arcs connecting their endpoints, and edges intersect at most once either in a proper crossing or in a shared endpoint. A tantalizing open question regarding simple drawings is how many pairwise disjoint edges every simple drawing of the complete graph with  $n$  vertices,  $K_n$ , contains. In other words, we ask for the maximal size of a plane matching that can always be found in such a drawing. Ruiz-Vargas [8] showed in 2017 that every simple drawing of  $K_n$  contains  $\Omega(n^{\frac{1}{2}-\varepsilon})$  pairwise disjoint edges for any  $\varepsilon > 0$ , which

---

This is an abstract of a presentation given at CG:YRF 2021. 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.



**Figure 1** A generalized twisted drawing of  $K_5$ . All edges cross the (red) ray  $r$ .

improves several previous results ( $\Omega((\log n)^{\frac{1}{6}})$  in 2003 [5],  $\Omega(\frac{\log n}{\log \log n})$  in 2005 [6],  $\Omega((\log n)^{1+\varepsilon})$  in 2009 [1], and  $\Omega(n^{\frac{1}{3}})$  in 2013 and 2014 [2, 3, 9].) We further improve the bound by showing that every simple drawing of  $K_n$  contains  $\Omega(n^{\frac{1}{2}})$  pairwise disjoint edges.

To prove this new bound, we will use some properties of a special kind of simple drawings that we call generalized twisted drawings. (Classical twisted drawings as defined in [5] belong to the family of generalized twisted drawings.)

► **Definition 1.** *A simple drawing  $D$  is **c-monotone** (short for circularly monotone) if there is a point  $O$  such that any ray emanating from  $O$  intersects any edge of  $D$  at most once.*

*A **generalized twisted** drawing is a c-monotone drawing  $D$ , in which there exists a ray  $r$  emanating from  $O$  that intersects every edge of  $D$ .*

For simplicity, we assume that in c-monotone drawings, the vertices lie on a circle with  $O$  as the center. We label them  $v_1, \dots, v_n$  in counterclockwise order along the circle. Further, for generalized twisted drawings, we assume that the ray  $r$  intersects the circle between  $v_1$  and  $v_n$ . Figure 1 shows an example of a generalized twisted drawing of  $K_5$ .

We prove in Section 2 that generalized twisted drawings of  $K_n$  always contain  $\lfloor \frac{n}{2} \rfloor$  pairwise disjoint edges (Theorem 2). We use this result to show that all simple drawings of  $K_n$  contain  $\Omega(\sqrt{n})$  disjoint edges (Theorem 3). The proof of Theorem 3 is sketched in Section 3.

► **Theorem 2.** *Every generalized twisted drawing of  $K_n$  contains  $\lfloor \frac{n}{2} \rfloor$  pairwise disjoint edges.*

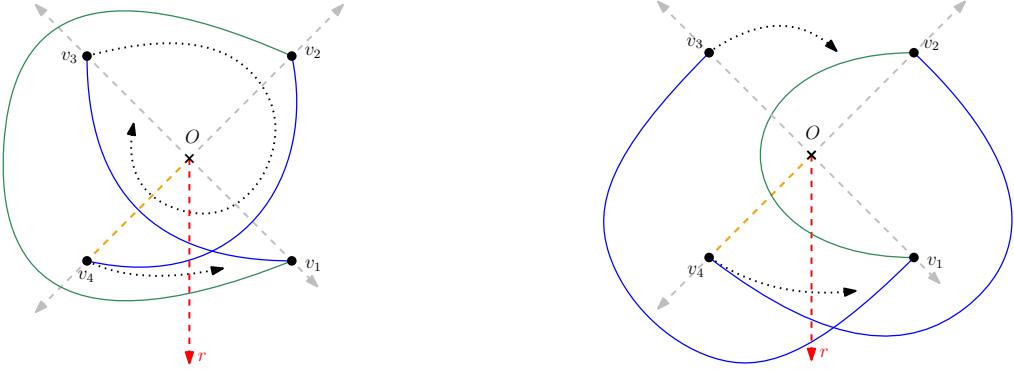
► **Theorem 3.** *Every simple drawing of  $K_n$  contains at least  $\left\lfloor \sqrt{\frac{n}{48}} \right\rfloor$  pairwise disjoint edges.*

## 2 Proof of Theorem 2

To prove Theorem 2, we will use the following lemma.

► **Lemma 4.** *Let  $D$  be a generalized twisted drawing of  $K_4$ , with vertices  $\{v_1, v_2, v_3, v_4\}$  labeled counterclockwise along the circle. Then the edges  $\overline{v_1v_3}$  and  $\overline{v_2v_4}$  do not cross.*

**Proof.** Assume, for a contradiction, that the edge  $\overline{v_1v_3}$  crosses the edge  $\overline{v_2v_4}$ . There are two possibilities to draw the crossing edges  $\overline{v_1v_3}$  and  $\overline{v_2v_4}$ , depending on whether  $\overline{v_1v_3}$  crosses the (straight-line) segment from  $O$  to  $v_4$  or not; cf. Figure 2. In both cases, there is only one way to draw  $\overline{v_1v_2}$  such that the drawing stays generalized twisted, yielding two regions



**Figure 2** The two possibilities to draw  $\overline{v_1v_3}$  and  $\overline{v_2v_4}$  crossing and generalized twisted.

bounded by all drawn edges. The vertices  $v_3$  and  $v_4$  lie in the same region. Since a simple drawing of  $K_4$  has at most one crossing, the edge  $\overline{v_3v_4}$  cannot leave this region. However, it is impossible to draw  $\overline{v_3v_4}$  without leaving the region such that it is c-monotone and crosses the ray  $r$  (see the dotted arrows in Figure 2 for necessary emanating directions of  $\overline{v_3v_4}$ ).  $\blacktriangleleft$

With Lemma 4, we can find  $\lfloor \frac{n}{2} \rfloor$  pairwise disjoint edges in generalized twisted drawings.

**Proof of Theorem 2.** Let  $D$  be a generalized twisted drawing of  $K_n$ . Let  $S$  be the set of edges  $\overline{v_1v_{\lceil \frac{n}{2} \rceil+1}}, \overline{v_2v_{\lceil \frac{n}{2} \rceil+2}}, \dots, \overline{v_{\lfloor \frac{n}{2} \rfloor}v_n}$ . Every pair of edges in  $S$  has the form  $\overline{v_iv_{\lceil \frac{n}{2}+i \rceil}}, \overline{v_jv_{\lceil \frac{n}{2}+j \rceil}}$ , where  $0 < i, j < \lfloor \frac{n}{2} \rfloor$ . Thus, all edges in  $S$  are pairwise disjoint by Lemma 4.  $\blacktriangleleft$

### 3 Proof Sketch of Theorem 3

The complete proof of Theorem 3 will appear in a future full version of this paper. We sketch the proof here. We will use the following two theorems.

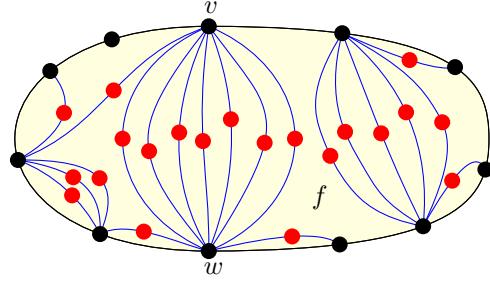
► **Theorem 5 ([4]).** *Any maximal plane subgraph of a simple drawing of  $K_n$  is biconnected.*

► **Theorem 6 ([7]).** *Let  $D$  be a simple drawing of  $K_n$  with  $n \geq 3$ . Let  $H$  be a connected plane subdrawing of  $D$  containing at least two vertices, and let  $v$  be a vertex in  $D \setminus H$ . Then  $D$  contains two edges incident to  $v$  that connect  $v$  with  $H$  and do not cross any edges of  $H$ .*

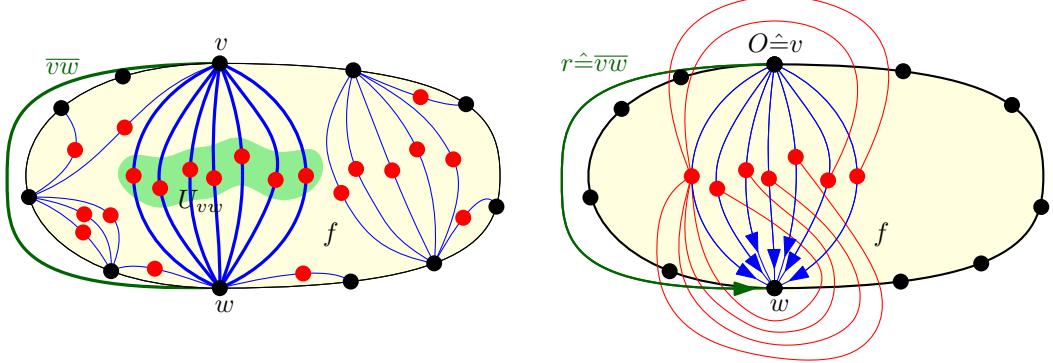
Let  $D$  be a simple drawing of  $K_n$ , and let  $M$  be a maximal plane matching of  $D$ . If  $|M| \geq \sqrt{\frac{n}{48}}$ , then Theorem 3 holds. So assume that  $|M| < \sqrt{\frac{n}{48}}$ . We show how to find another plane matching, whose size is at least  $\sqrt{\frac{n}{48}}$ , which is a contradiction.

We consider a maximal plane subdrawing  $H$  of  $D$  that contains  $M$ . It is biconnected by Theorem 5, and thus partitions the plane into faces. By counting arguments and planarity, we can show that there exists a face  $f$  in  $H$  such that the number of (unmatched) vertices of  $D$  inside  $f$  is at least  $\frac{\sqrt{48n}}{12}|f|$ , where  $|f|$  denotes the number of (matched) vertices on the boundary of  $f$ .

From  $f$  and the set  $U(f)$  of vertices strictly inside  $f$ , we construct a plane subdrawing  $H'$  as follows; see Figure 3. We add the vertices and edges on the boundary of  $f$ , and for every vertex in  $U(f)$ , we add the vertex and two edges of  $D$  incident to the vertex such that the resulting drawing stays plane in every step. These two edges exist by Theorem 6. Since  $M$  is maximal, no edge in  $H'$  can connect two vertices of  $U(f)$  (as they are unmatched).



**Figure 3** The plane subgraph  $H'$  contains two (blue) edges for every (red) vertex in  $U(f)$ .



**Figure 4** The vertices in  $U_{vw}$  induce a subdrawing that is weakly isomorphic to a generalized twisted drawing. (On the right the edges in  $D'$  incident to the leftmost vertex of  $U_{vw}$  are depicted).

By counting arguments and planarity, we can show that there are two vertices,  $v$  and  $w$ , on the boundary of  $f$ , such that the cardinality of the set of vertices in  $U(f)$  that are connected to both  $v$  and  $w$  in  $H'$  is at least  $\sqrt{\frac{n}{12}}$ . We call this set  $U_{vw}$ ; see Figure 4.

Finally, we can prove that the subdrawing  $D'$  of  $D$  induced by the vertices in  $U_{vw}$  is weakly isomorphic to a generalized twisted drawing, because every edge in  $D'$  must cross the edge  $\overline{vw}$ , see Figure 4 (right). Therefore, by Theorem 2, there are  $\left\lfloor \sqrt{\frac{n}{48}} \right\rfloor$  pairwise disjoint edges in  $D'$  and thus in  $D$ .

---

## References

---

- 1 Jacob Fox and Benny Sudakov. Density theorems for bipartite graphs and related ramsey-type results. *Combinatorica*, 29(2):153–196, 2009. doi:10.1007/s00493-009-2475-5.
- 2 Radoslav Fulek. Estimating the number of disjoint edges in simple topological graphs via cylindrical drawings. *SIAM Journal on Discrete Mathematics*, 28(1):116–121, 2014. doi:10.1137/130925554.
- 3 Radoslav Fulek and Andres J. Ruiz-Vargas. Topological graphs: empty triangles and disjoint matchings. In *Proceedings of the 29th Annual Symposium on Computational Geometry (SoCG’13)*, pages 259–266, 2013. doi:10.1145/2462356.2462394.
- 4 Alfredo García, Alexander Pilz, and Javier Tejel. On plane subgraphs of complete topological drawings. *ARS MATHEMATICA CONTEMPORANEA*, 2020. doi:10.26493/1855-3974.2226.e93.
- 5 János Pach, József Solymosi, and Gézak Tóth. Unavoidable configurations in complete topological graphs. *Discrete Comput Geometry*, 30:311–320, 2003. doi:10.1007/s00454-003-0012-9.

- 6 János Pach and Géza Tóth. Disjoint edges in topological graphs. In *Proceedings of the 2003 Indonesia-Japan Joint Conference on Combinatorial Geometry and Graph Theory (IJC-CGGT'03)*, volume 3330, pages 133–140, 2005. doi:[10.1007/978-3-540-30540-8\\_15](https://doi.org/10.1007/978-3-540-30540-8_15).
- 7 Andres J. Ruiz-Vargas. Empty triangles in complete topological graphs. In *Discrete Computational Geometry*, volume 53, pages 703–712, 2015. doi:[10.1007/s00454-015-9671-4](https://doi.org/10.1007/s00454-015-9671-4).
- 8 Andres J. Ruiz-Vargas. Many disjoint edges in topological graphs. *Computational Geometry*, 62:1–13, 2017. doi:[10.1016/j.comgeo.2016.11.003](https://doi.org/10.1016/j.comgeo.2016.11.003).
- 9 Andrew Suk. Disjoint edges in complete topological graphs. *Discrete & Computational Geometry*, 49(2):280–286, 2013. doi:[10.1007/s00454-012-9481-x](https://doi.org/10.1007/s00454-012-9481-x).

# Connecting 3-manifold triangulations with semi-monotonic sequences of bistellar flips\*

Benjamin A. Burton

The University of Queensland, Brisbane, Australia

<http://www.maths.uq.edu.au/~bab/>

bab@maths.uq.edu.au

Alexander He 

The University of Queensland, Brisbane, Australia

<https://sites.google.com/view/alex-he>

a.he@uqconnect.edu.au

---

## Abstract

---

A key result in computational 3-manifold topology is that any two triangulations of the same 3-manifold are connected by a finite sequence of bistellar flips, also known as Pachner moves. We strengthen this result by showing that there must always be a sequence that satisfies a rigid property that we call “semi-monotonicity”.

**2012 ACM Subject Classification** Mathematics of computing → Geometric topology

**Keywords and phrases** Computational topology, 3-manifolds, triangulations, special spines, bistellar flips, Pachner moves

**Related Version** A full version of the paper is available at <https://arxiv.org/abs/2012.02398>.

**Supplementary Material** The source code for the experiments is available at  
<https://people.smp.uq.edu.au/BenjaminBurton/code/>

**Acknowledgements** We would like to thank the anonymous reviewers for their helpful comments.

For computational purposes, simplicial complexes are often prohibitively large; thus, it is useful to work with a more flexible (and typically smaller) structure called a **one-vertex triangulation**. Every 3-manifold<sup>1</sup> admits a one-vertex triangulation [3].

► **Notation 1.** In the statements of Theorem 2, Conjecture 3 and Theorem 4, let  $\mathcal{T}$  and  $\mathcal{U}$  be two one-vertex triangulations of the same 3-manifold, each with at least two tetrahedra.

How can we tell when two one-vertex triangulations represent the same 3-manifold? It is clear that applying a 2-3 or 3-2 move (see Figure 1) preserves the underlying 3-manifold.<sup>2</sup> Matveev [4, p. 29] and Piergallini [5] independently proved the following theorem.

► **Theorem 2 (Matveev and Piergallini).**  $\mathcal{T}$  can always be transformed into  $\mathcal{U}$  by a finite sequence of 2-3 and 3-2 moves.

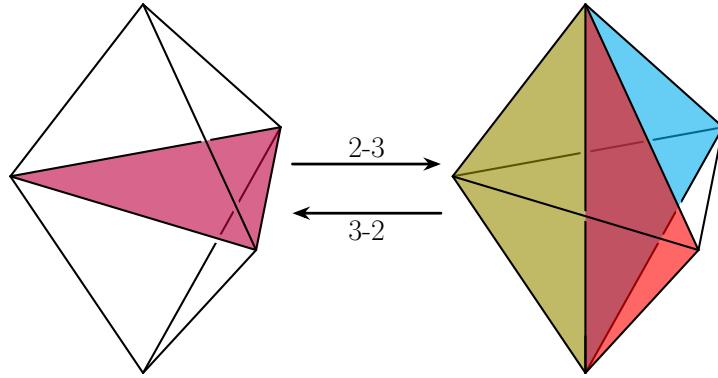
Unfortunately, the Matveev-Piergallini result says nothing about the structure of the sequence of moves. Being able to guarantee some structure could be useful for theoretical applications, such as proving new topological invariants for 3-manifolds. A natural way to impose structure is to require our sequences to break up into two parts: first, a **(monotonic) ascent**, where the moves only increase the number of tetrahedra; and second, a **(monotonic)**

---

\* This is an abstract of a presentation given at CG:YRF 2021. 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.

<sup>1</sup> For an introduction to 3-manifold topology that is written for a wide audience, see [6].

<sup>2</sup> The 2-3 and 3-2 moves are two of the four moves known collectively as *bistellar flips* or *Pachner moves*.

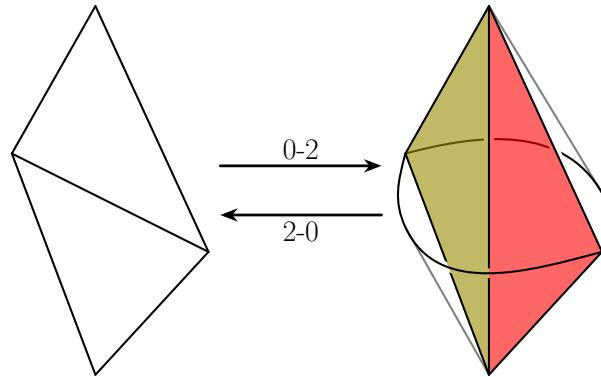


■ **Figure 1** The 2-3 and 3-2 moves. The shaded triangles are the internal faces.

**descent**, where the moves only decrease the number of tetrahedra. Based on detailed experiments, it seems reasonable to believe the following claim.

► **Conjecture 3.**  $\mathcal{T}$  can always be transformed into  $\mathcal{U}$  by a sequence consisting of: an ascent via zero or more 2-3 moves, and then a descent via zero or more 3-2 moves.

We prove a variant of this conjecture where we allow the descent to begin with some 2-0 moves; Figure 2 illustrates the 2-0 move (as well as its inverse, the 0-2 move). It is important to note that certain conditions are required for a 2-0 move to preserve the underlying 3-manifold of a triangulation; see [1, pp. 8–9] for a list of such conditions.



■ **Figure 2** The 0-2 and 2-0 moves. The shaded triangles are the internal faces.

We call a sequence **semi-monotonic** if it consists of: an ascent via zero or more 2-3 moves, and then a descent via zero or more 2-0 moves followed by zero or more 3-2 moves.

► **Theorem 4.**  $\mathcal{T}$  can always be transformed into  $\mathcal{U}$  by a semi-monotonic sequence.

**Proof outline.** Given any sequence  $\Sigma$  of 2-3, 3-2 and 2-0 moves, let  $\delta_\Sigma$  denote the last 2-3 move in  $\Sigma$ . We say that  $\Sigma$  is *benign* if all the 2-0 moves occur consecutively in a (possibly empty) sequence appearing immediately after  $\delta_\Sigma$ . Observe that a benign sequence  $\Sigma$  is semi-monotonic if and only if all the 3-2 moves occur after the 2-3 move  $\delta_\Sigma$ . In other words,

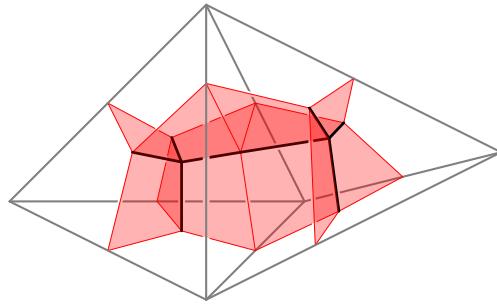
we can regard any 3-2 move that occurs before  $\delta_\Sigma$  as a *bad 3-2 move*, and hence say that a benign sequence is semi-monotonic if and only if it has no bad 3-2 moves.

With this in mind, our strategy is to start with a benign sequence that transforms  $\mathcal{T}$  into  $\mathcal{U}$ , and to then reduce the number of bad 3-2 moves until we get a semi-monotonic sequence. The Matveev-Piergallini theorem (Theorem 2) tells us that  $\mathcal{T}$  can be transformed into  $\mathcal{U}$  by a sequence that uses only 2-3 and 3-2 moves; we use this as our benign starting sequence.

To reduce the number of bad 3-2 moves in a benign sequence  $\Sigma$ , let  $\gamma$  denote the last such move in  $\Sigma$ . Consider the subsequence of moves starting with  $\gamma$  and ending with  $\delta_\Sigma$ ; this subsequence consists of: a single 3-2 move (namely  $\gamma$ ), and then a series of 2-3 moves ending with  $\delta_\Sigma$ . We claim that this subsequence can be replaced by a new subsequence consisting of: a series of 2-3 moves, and then a series of 2-0 moves. Observe that this “replacement step” (if it is possible) yields a new benign sequence  $\Sigma'$  that has one less bad 3-2 move than  $\Sigma$ .

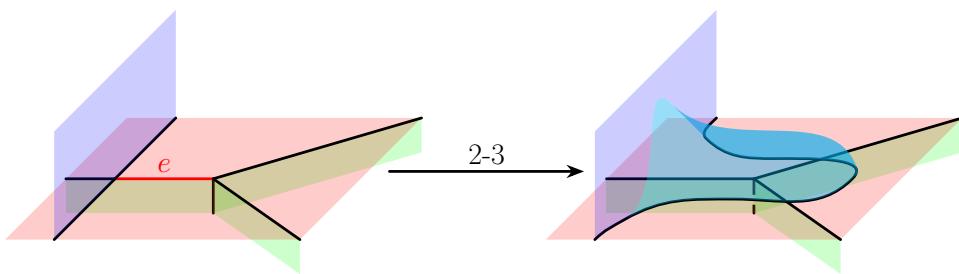
It only remains to show that the “replacement step” is indeed always possible. The full details can be found in [2]; here, we settle with giving a rough sketch.

Our argument is more easily described by converting the triangulations into their dual special spines. In essence, the special spine dual to a triangulation is the 2-dimensional complex  $P$  obtained by replacing each tetrahedron with a six-winged “butterfly”, as shown in Figure 3; see [4, pp. 12–13] for a more rigorous description of this duality.



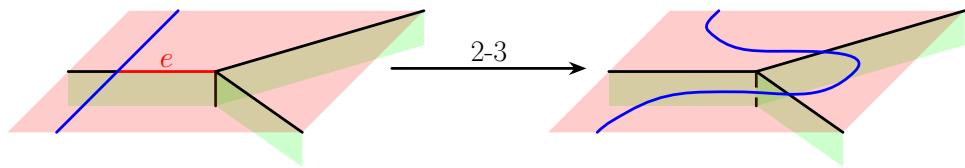
**Figure 3** The dual butterflies inside two adjacent tetrahedra.

We can translate the 2-3 and 0-2 moves (as well as their inverses, the 3-2 and 2-0 moves) into the setting of special spines; see Figures 4 to 7.

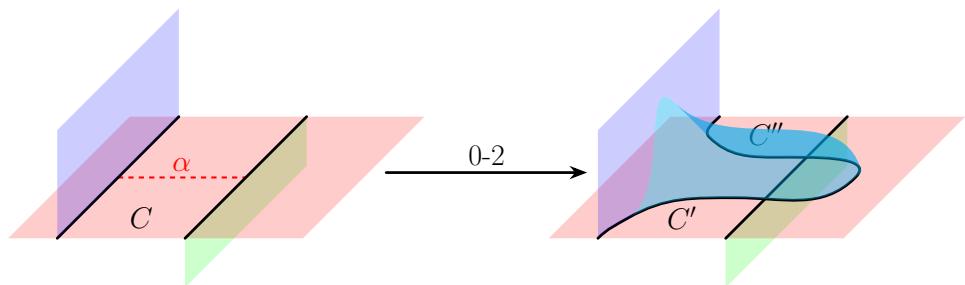


**Figure 4** The 2-3 move along the red edge  $e$ .

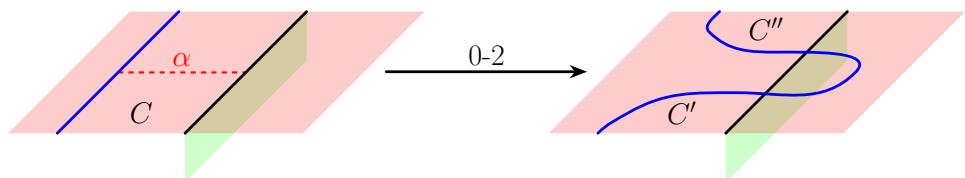
A useful trick when working with special spines is using a 0-2 move to create an “arch-with-membrane”, or “arch” for short; this is illustrated in Figures 8 and 9. At the very end, we use this idea in reverse: we destroy an arch using a 2-0 move.



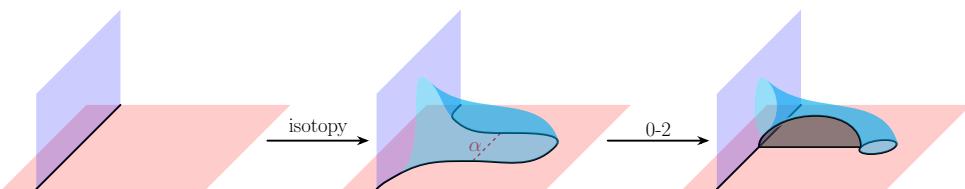
**Figure 5** A simplified drawing of the 2-3 move.



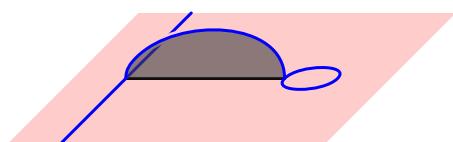
**Figure 6** The 0-2 move along the red dashed curve  $\alpha$ .



**Figure 7** A simplified drawing of the 0-2 move.

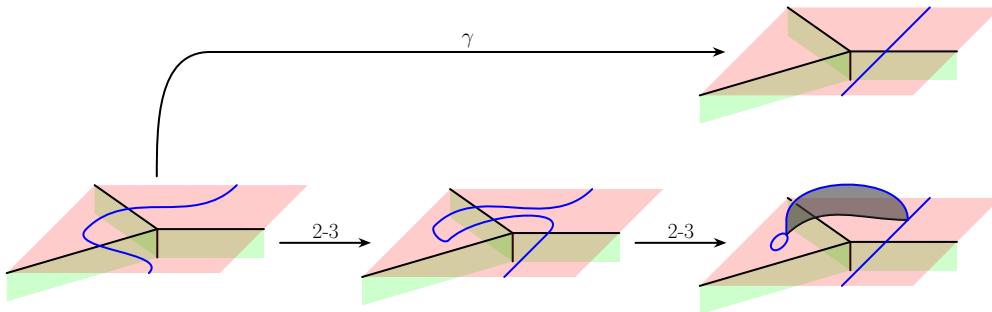


**Figure 8** We can create an arch by performing a 0-2 move along the curve  $\alpha$ .

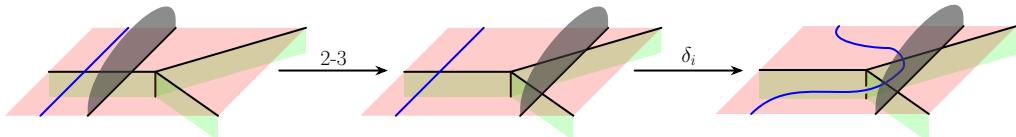


**Figure 9** A simplified drawing of an arch.

With this in mind, let  $P$  and  $Q$  be two special spines, and suppose we have a sequence  $\gamma, \delta_1, \dots, \delta_k$  transforming  $P$  into  $Q$ , such that  $\gamma$  is a 3-2 move and each  $\delta_i$  is a 2-3 move. Our “replacement step” begins by replacing the 3-2 move  $\gamma$  with two 2-3 moves, as shown in Figure 10; this creates an arch in the special spine. We then perform each 2-3 move  $\delta_i$  in order, using additional 2-3 moves to shift the arch out of the way whenever necessary; this idea is illustrated in Figure 11. To recover the special spine  $Q$ , we need to remove the arch that we created; as shown in Figure 12, we can do this using a series of 2-3 moves followed by a series of 2-0 moves. This completes the “replacement step”.  $\blacktriangleleft$



**Figure 10** We replace the 3-2 move  $\gamma$  with two 2-3 moves, at the cost of creating an arch.

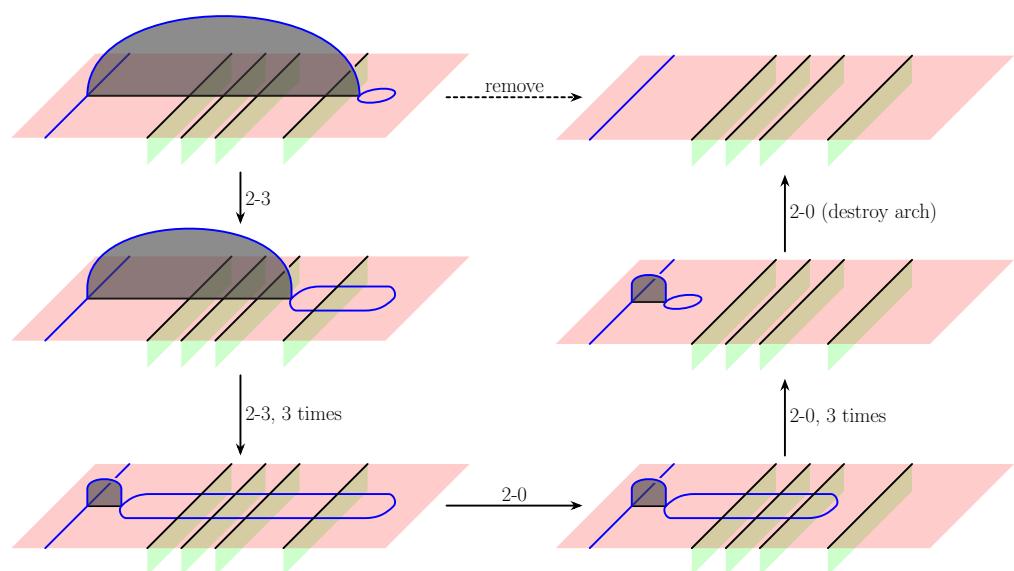


**Figure 11** To perform the move  $\delta_i$ , we first need to shift the grey arch fragment out of the way.

---

## References

- 1 Benjamin A. Burton. Computational topology with Regina: Algorithms, heuristics and implementations. In *Geometry and Topology Down Under*, volume 597 of *Contemporary Mathematics*, pages 195–224. American Mathematical Society, 2013.
- 2 Benjamin A. Burton and Alexander He. Connecting 3-manifold triangulations with monotonic sequences of bistellar flips. [arXiv:2012.02398](https://arxiv.org/abs/2012.02398), 2020.
- 3 William Jaco and J. Hyam Rubinstein. 0-efficient triangulations of 3-manifolds. *J. Differential Geom.*, 65(1):61–168, 2003.
- 4 Sergei V. Matveev. *Algorithmic Topology and Classification of 3-Manifolds*, volume 9 of *Algorithms and Computation in Mathematics*. Springer-Verlag, second edition, 2007.
- 5 Riccardo Piergallini. Standard moves for standard polyhedra and spines. In *III Convegno Nazionale Di Topologia: Trieste, 9-12 Giugno 1986 : Atti*, number 18 in *Rend. Circ. Mat. Palermo (2) Suppl.*, pages 391–414, 1988.
- 6 Jeffrey R. Weeks. *The Shape of Space*. CRC Press, second edition, 2002.



■ **Figure 12** Removing an arch using 2-3 and 2-0 moves.

# Large Perimeter Objects Surrounded by 1.5D Terrains\*

Vahideh Keikha  

The Czech Academy of Sciences, Institute of Computer Science, Pod Vodárenskou věží 2, 182 07 Prague, Czech Republic

---

## Abstract

Given is a 1.5D terrain  $\mathcal{T}$ , i.e., an  $x$ -monotone polygonal chain in  $\mathbb{R}^2$ . Our objective is to approximate a largest perimeter convex polygon with at most  $k$  vertices inside  $\mathcal{T}$ . We introduce a general algorithm that efficiently approximates such polygons with a constant  $k > 0$  within a factor  $(1 - \epsilon)$ .

**2012 ACM Subject Classification** Design and analysis of algorithms

**Keywords and phrases** 1.5D Terrain, Largest Perimeter  $k$ -gon

**Digital Object Identifier** 10.4230/LIPIcs...

**Funding** *Vahideh Keikha*: Is supported by the Czech Science Foundation, grant number GJ19-06792Y, and by institutional support RVO: 67985807

## 1 Introduction

Computing the largest-area triangle inside 1.5D terrains was studied in [2], by designing a quadratic exact algorithm and sub-quadratic approximation algorithms. We study the following problem: Let  $\mathcal{T}$  be a 1.5D terrain of  $n$  vertices, and let  $k > 0$  be a constant integer; compute the largest-perimeter convex polygon with at most  $k$  vertices in  $\mathcal{T}$ . See Figure 1(a) for an illustration for  $k = 3$ . Table 1 gives a summary of the new and known results. We refer the reader to [1] for recent results on polygon inclusion in simple polygons.

## 2 Largest Perimeter

Let  $\mathcal{T}$  be a 1.5D terrain. We first define the *diameter* of  $\mathcal{T}$  as the longest line segment within  $\mathcal{T}$ . Let  $l^*$  denote the diameter of  $\mathcal{T}$ .

► **Lemma 1.**  *$l^*$  is either (1) supported by two convex vertices, (2) supported by a reflex vertex and a convex vertex, or (3) supported by two reflex vertices of the terrain.*

See Figure 1(b). For computing  $l^*$ , the existence of an  $O(n^2 \log n)$  time algorithm by considering any pair of vertices and ray shooting is obvious. However, it can be slightly improved by careful analysis. Any arbitrary triangle which has the diameter of the terrain as a side has at least  $\frac{1}{3}$  of the perimeter of the optimal solution. This gives a simple  $O(n^2 \log n)$  time approximation algorithm which we use to design our FPTAS.

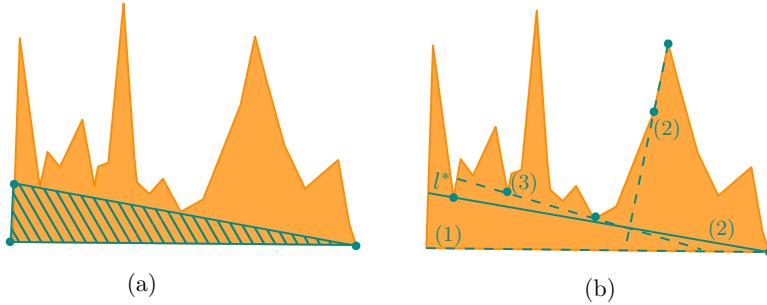
Let  $P^*$  and  $P$  denote the largest perimeter triangle and a  $\frac{1}{3}$ -approximation of the largest perimeter triangle, respectively. Also let  $|P|$  denote the perimeter of  $P$ . Consider a grid of big cells of side length  $6|P|$ . Let the bottom left corner of a big cell in the grid lies at the leftmost vertex of  $\mathcal{T}$  (let  $(0, 0)$  denote its coordinates). Consider three copies of this big cell with the bottom left corners at coordinates  $(3|P|, 0)$ ,  $(0, 3|P|)$  and  $(3|P|, 3|P|)$ , respectively.

---

\* This is an abstract of a presentation given at CG:YRF 2021. 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.

| Measure                                  | Time                                | Apprx.         | In Object      | Ref.     |
|--|-------------------------------------|----------------|----------------|----------|
| Max $\mathcal{A}$ Triangle               | $O(n \log n)$                       | $1/4$          | Simple Polygon | [4]      |
| Max $\mathcal{A}$ $\delta$ -Fat Triangle | $O(n)$                              | $1 - \epsilon$ | Simple Polygon | [4]      |
| Max $\mathcal{A}$ Triangle               | $O(n^2)$                            | exact          | Terrain        | [2]      |
| Max $\mathcal{A}$ Triangle               | $O(n \log n)$                       | $1/2$          | Terrain        | [2]      |
| Max $\mathcal{A}$ Triangle               | $O(\epsilon^{-1}n \log^2 n)$        | $1 - \epsilon$ | Terrain        | [2]      |
| Max $\mathcal{P}$ Triangle               | $O(\epsilon^{-6}n \log^2 n)$        | $1 - \epsilon$ | Terrain        | Thm. 3   |
| Max $\mathcal{P}$ at most $k$ -gon       | $O(k^{10}\epsilon^{-2k}n \log^2 n)$ | $1 - \epsilon$ | Terrain        | Sec. 2.1 |

■ **Table 1** New and known results;  $\mathcal{A}$  stands for the area, and  $\mathcal{P}$  stands for the perimeter. In a  $\delta$ -Fat triangle all three angles are at least  $\delta$ .

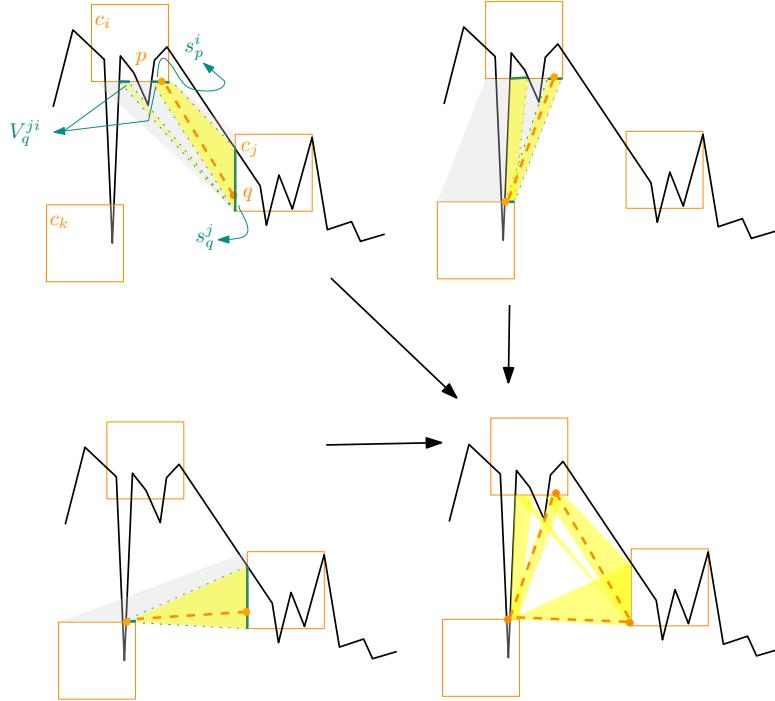


■ **Figure 1** (a) The largest perimeter triangle in  $\mathcal{T}$ . (b) Different candidates of  $l^*$ .

Since  $\mathcal{T}$  is monotone with a horizontal base, the union of these 4 squares covers the terrain entirely, and since a triangle has at most one obtuse angle,  $P^*$  is contained in one of the big cells entirely. From the construction, each edge of  $\mathcal{T}$  intersects at most 4 big cells. We then decompose each big cell by  $O(\frac{1}{\epsilon})$  finer cells of side length  $\epsilon|P|$ , for a given  $\epsilon > 0$ . Let  $X$  denote the set of finer cells. Observe that the total complexity of the intersection of  $X$  and  $\mathcal{T}$  is in  $O(n)$  since each edge of  $\mathcal{T}$  is intersecting with at most 4 big cells. So computing the set  $X' \subseteq X$  of finer cells that are intersecting with  $\mathcal{T}$  takes  $O(n)$  time. We consider any of the four big cells independently. Let  $c_i, c_j$  and  $c_k$  be any triple of finer cells in  $X'$ , all in a specific big cell  $C$ . For two line segments  $s_p^i$  and  $s_q^j$ , we consider the intersection of the visibility polygons by assuming an edge guard at  $s_p^i$  and another one at  $s_q^j$ . However, we do not need to compute the visibility polygons explicitly. Let  $V_q^{ji}$  denote the ranges on the corresponding side of  $c_i$  that are visible to  $s_q^j$ . We are interested in determining whether there are three line segments in  $\mathcal{T} \cap C$  whose endpoints lie at the intersection of the pairwise visibility ranges of the segments  $s_p^i \subset c_i$ ,  $s_q^j \subset c_j$  and  $s_r^k \subset c_k$ . This happens when the intersection of the pairwise visibility ranges of  $s_p^i, s_q^j$  and  $s_r^k$  have a non-empty intersection, i.e.,  $V_p^{ij} \cap V_q^{kj} \neq \emptyset$ ,  $V_q^{ji} \cap V_r^{ki} \neq \emptyset$  and  $V_p^{ik} \cap V_r^{jk} \neq \emptyset$ , as illustrated in Figure 2.

Let  $s_1^i, \dots, s_{n_i}^i$  and  $s_1^j, \dots, s_{n_j}^j$  denote the sequence of the intervals on  $c_i$  and  $c_j$ , respectively, that lies within  $\mathcal{T}$ . For each pair of  $s_p^i$  and  $s_q^j$ , to efficiently determine whether there is any segment which lies within  $\mathcal{T}$  entirely, and with endpoints at these intervals, we use a combination of two data structures.

For two points  $p \in s_p^i$  and  $q \in s_q^j$ , the line segment  $\overline{pq}$  must lie in the visibility range of both  $s_p^i$  and  $s_q^j$ . The visibility range of any of  $s_p^i$  and  $s_q^j$  can be computed in  $O(\log n)$  time by performing shortest path queries on preprocessed connected components achieved in  $O(n)$  time [3], and checking whether they can see each other or not can be done at the same time



**Figure 2** Computing the visibility ranges on a triple of finer cells that are introducing a triangle. The visibility ranges are shown in green fat segments. This triple of segments have a non-empty intersection for the pairwise visibility ranges of the segments.

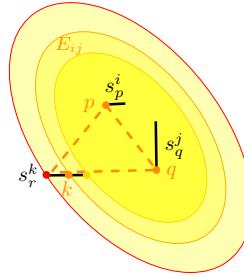
by simple visibility queries inside simple polygons. We first make a 1D range tree at the indices of the intervals in  $c_i$  in  $O(n_i \log n_i)$  time, and of the height  $O(\log n_i)$ . Then we make an interval tree at each node of the constructed range tree. Thus our data structure has a size  $O(n_i \log n_i)$  and takes  $O(n_i \log^2 n_i)$  time (spending  $O(n_i \log n_i)$  time at each level of the range tree). The queries are is there any interval  $s_p^i$  in  $c_i$  which is visible from an interval  $s_q^j$  in  $c_j$ . This can be answered in  $O(\log n_i)$  time. Performing  $n_j$  queries like this gives us the running time  $O((n_i + n_j) \log^2 n_i)$  for each pair of cells in a big cell. There are  $O(\epsilon^{-2})$  cells, and we consider any triple of cells. For any triple of intervals that the intersection of the pairwise visibility regions is non-empty, we compute the largest perimeter triangle.

**Solution on a triple of segments** Any triple  $s_p^i, s_q^j, s_r^k$  of segments that may contain the vertices of  $P^*$  have a horizontal or vertical direction. Suppose the locations of two points on their segments, say  $s_p^i, s_q^j$  are fixed. The function that describes the longest perimeter changes as a symmetric hyperbolic function with a unique minimum. Thus there will always be at least one direction in which a point can be moved such that the size does not decrease.

► **Lemma 2.** *For any triple  $s_p^i, s_q^j$  and  $s_r^k$  of segments, the largest perimeter triangle  $P^*$  with one vertex at each segment has its vertices at the endpoints of the segments.*

See Figure 3. Consequently, one can compute a  $(1 - \epsilon)$ -approximation of the largest perimeter triangle in a terrain in  $O(n^2 \log n + \epsilon^{-6} n \log^2 n)$  time. We note that there exists a  $(1 - \epsilon)$ -approximation with running time  $O(\epsilon^{-4} n \log^2 n)$  for the longest line segment inside a simple polygon [4]. Using this algorithm for computing the diameter of the terrain, we have:

► **Theorem 3.** *One can compute a  $(1 - \epsilon)$ -approximation of the largest perimeter triangle in a terrain  $\mathcal{T}$  of  $n$  vertices in  $O(\epsilon^{-6} n \log^2 n)$  time.*



**Figure 3**  $E_{ij}$  is the constructed ellipse at  $P^* = \triangle pqk$ . If there is at least one line segment  $s_r^k$  that does not contribute an endpoint to  $P^*$ , at least one of the two ellipses which are constructed at  $p, q$  and one of the endpoints of  $s_r^k$  contains  $E_{ij}$  (the two ellipses coincide if  $s_r^k$  is parallel to  $\overline{pq}$ ).

## 2.1 Extension to polygons of $k > 3$ vertices

Extension of the triangle algorithm to convex polygons of  $k > 3$  vertices and to the area measure is straightforward. We consider any set of  $k$  intervals, as before, since we need to compute the pairwise visibility conditions of the visibility ranges for all the selections of  $k$  intervals. Note that the non-empty intersection condition is necessary again to find a largest perimeter polygon of  $k$  vertices that lie inside  $\mathcal{T}$  entirely, however, in the algorithm, we may report a convex polygon of less than  $k$  vertices if it has a larger area/perimeter than any convex  $k$ -gon with vertices at the selected  $k$  intervals.

---

### References

---

- 1 Sergio Cabello, Josef Cibulka, Jan Kyncl, Maria Saumell, and Pavel Valtr. Peeling potatoes near-optimally in near-linear time. *SIAM Journal on Computing*, 46(5):1574–1602, 2017.
- 2 Arun Kumar Das, Sandip Das, and Joydeep Mukherjee. Largest triangle inside a terrain. *Theoretical Computer Science*, 858:90–99, 2021.
- 3 Leonidas J Guibas and John Hershberger. Optimal shortest path queries in a simple polygon. *Journal of Computer and System Sciences*, 39(2):126–152, 1989.
- 4 Olaf Hall-Holt, Matthew J Katz, Piyush Kumar, Joseph SB Mitchell, and Arik Sityon. Finding large sticks and potatoes in polygons. In *SODA*, volume 6, pages 474–483, 2006.

# An exact optimal algorithm for the discrete median line segment problem in the plane?\*

Ovidiu Daescu  

Department of Computer Science, University of Texas at Dallas, Richardson, TX, USA

Ka Yaw Teo<sup>1</sup>  

Department of Computer Science, University of Texas at Dallas, Richardson, TX, USA

---

## Abstract

---

We present an exact algorithm for computing the *discrete median line segment* of a set  $P$  of  $n$  points in the plane – that is, finding a line segment with both of its endpoints belonging to  $P$  such that the sum of the distances from  $P$  to the line segment is minimized. Our algorithm runs in  $O(n^2)$  time.

**2012 ACM Subject Classification** Theory of computation → Computational geometry; Theory of computation → Facility location and clustering

**Keywords and phrases** Facility location, line segment, median, minsum

## 1 Introduction

Given a set  $P$  of  $n$  points in  $\mathbb{R}^2$ , the objective of the *discrete median line segment (DMLS)* problem is to locate a line segment bounded by two points of  $P$  such that the sum of the Euclidean distances from  $P$  to the line segment is minimized.

The proposed problem is closely related to a class of “discrete” problems in facility location theory, where the goal is to select one point (or several) from a given set of points  $P$  so as to minimize an objective function that is distance-dependent with respect to  $P$ .

There are two types of problems in facility location theory depending on the objective function used – i) center (minimax) and ii) median (minsum). The discrete center problem asks to locate a point in  $P$  that minimizes the maximum of the distances between the points of  $P$  and the located point. The discrete center problem can be solved in  $O(n \log n)$  time using the farthest-neighbor Voronoi diagram of  $P$  [2, Chapter 7]. The discrete median, which is commonly known as the medoid, is a point in  $P$  that has the minimal sum of distances to  $P$ . One can find the medoid of  $P$  by simply computing all  $O(n^2)$  pairwise distances. It has been argued that no exact algorithm exists for solving the medoid problem in  $o(n^2)$  time [5].

## 2 Solution approaches

One can find the DMLS in  $O(n^3)$  time by enumerating all  $O(n^2)$  candidate line segments and computing the corresponding sum of  $O(n)$  distances for each candidate.

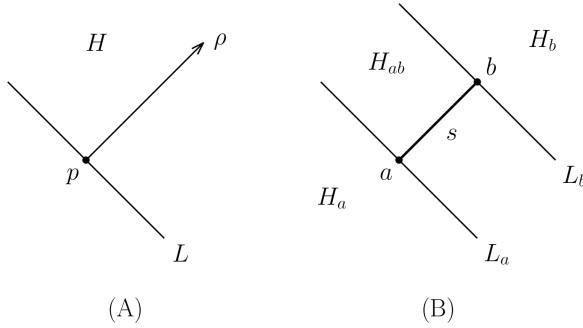
**An improved algorithm.** We now describe an  $O(n^2 \log n)$ -time algorithm for the DMLS problem. The idea is to preprocess  $P$  into some data structures of logarithmic query times for use in computing the sum of distances for each candidate line segment. The required data structures are derived from solving the following two subproblems (refer to Figure 1A).

---

\* This is an abstract of a presentation given at CG:YRF 2021. 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.

<sup>1</sup> Corresponding author

## An exact optimal algorithm for the discrete median line segment problem in the plane?



**Figure 1** (A) Illustration for Subproblems 1 and 2. (B) Computing the distances from  $P$  to  $s$ .

► **Subproblem 1.** Given a set  $P$  of  $n$  points in the plane, let  $H$  be a query half-plane bounded by a line  $L$  containing a point  $p \in P$ . Preprocess  $P$  so that, for a point  $p \in P$  and a half-plane  $H$  given at query time, one can efficiently report the sum of the distances from  $P \cap H$  to  $p$ .

► **Subproblem 2.** Given a set  $P$  of  $n$  points in the plane, let  $H$  be a query half-plane bounded by a line  $L$  containing a point  $p \in P$ . Let  $\rho$  be the ray emanating from  $p$ , perpendicular to  $L$ , and contained in  $H$ . Preprocess  $P$  so that, for a point  $p \in P$  and a half-plane  $H$  given at query time, one can efficiently report the sum of the orthogonal distances from  $P \cap H$  to  $\rho$ .

Here we give a brief description of the preprocessing procedure for solving the subproblems. For each point  $p \in P$ , we i) sort the points of  $P \setminus \{p\}$  around  $p$  in  $O(n \log n)$  time, ii) define a sequence of  $O(n)$  intervals in the sorted order such that  $P \cap H$  remains constant within each interval, iii) enumerate the intervals in the sorted order so that it takes  $O(1)$  time to evaluate the sum of distances in each interval, and iv) store the distance sums computed for the intervals in an  $O(\log n)$ -query time data structure. The results are summarized below.

► **Lemma 1.** In Subproblem 1 (resp. 2), a set  $P$  of  $n$  points can be preprocessed in  $O(n^2 \log n)$  time into an  $O(n^2)$ -size data structure so that, given a query point  $p \in P$  and a query half-plane  $H$ , one can report the distance sum from  $P \cap H$  to  $p$  (resp.  $\rho$ ) in  $O(\log n)$  time.

Let  $a$  and  $b$  denote the two endpoints of a candidate line segment  $s$ , where  $a, b \in P$  (Figure 1B). Let  $L_a$  (resp.  $L_b$ ) be the line passing through  $a$  (resp.  $b$ ) and perpendicular to  $s$ . Let  $H_a$  (resp.  $H_b$ ) be the half-plane bounded by  $L_a$  (resp.  $L_b$ ) and not containing  $s$ . Define  $H_{ab} = \mathbb{R}^2 \setminus (H_a \cup H_b)$ . Let  $\mathcal{D}_1$  and  $\mathcal{D}_2$  denote the query data structures derived from Subproblems 1 and 2, respectively. We can compute the sum of the distances from  $P \setminus \{a, b\}$  to  $s$  using  $\mathcal{D}_1$  and  $\mathcal{D}_2$  as follows.

We denote by  $\Sigma_\alpha$  the sum of the distances from  $P \cap H_\alpha$  to  $s$ , where  $\alpha \in \{a, b, ab\}$ . We can determine  $\Sigma_a$  by querying  $\mathcal{D}_1$  using  $a$  and  $H_a$  as the query inputs. Similarly,  $\Sigma_b$  can be found using  $\mathcal{D}_1$  with  $b$  and  $H_b$  as inputs for the query. To calculate  $\Sigma_{ab}$ , we perform the following queries. Define  $H_a' = \mathbb{R}^2 \setminus H_a$ . Let  $\Upsilon_a$  be the distance sum reported by querying  $\mathcal{D}_2$  using  $a$  and  $H_a'$  as the query inputs. Likewise, let  $\Upsilon_b$  be the distance sum reported from a query of  $\mathcal{D}_2$  with  $b$  and  $H_b$  as the query inputs. Then,  $\Sigma_{ab} = \Upsilon_a - \Upsilon_b$ . Finally, the distance sum from  $P \setminus \{a, b\}$  to  $s$  is given by  $\Sigma_a + \Sigma_b + \Sigma_{ab}$ . Overall, we perform four  $O(\log n)$ -time queries for each of  $O(n^2)$  candidate line segments. We hence reach the following conclusion.

► **Theorem 2.** The DMLS problem can be solved in  $O(n^2 \log n)$  time using  $O(n^2)$  space.

**A further improvement.** We can reduce the running time of the algorithm above by an  $O(\log n)$  factor as follows. Using the point-line duality transform, point set  $P$  can be mapped

into a set of  $n$  lines, whose arrangement can be constructed in  $O(n^2)$  time using  $O(n^2)$  space [1, 3]. For any point  $p \in P$ , the point set  $P \setminus \{p\}$  can be obtained in sorted order around  $p$  by simply traversing the vertices along the dual line of point  $p$  in  $O(n)$  time. These sorted points correspond to the limit points of the sorted intervals in Subproblems 1 and 2.

In addition, we note that each query half-plane in our problem is associated with a line containing a pair of points in  $P$ . Thus, we can compute in advance the set of all  $O(n)$  possible query half-planes for each point  $p \in P$ . We can then index each of these half-planes, and use the indices to create hash tables for look-up in place of the current logarithmic-time query data structures  $\mathcal{D}_1$  and  $\mathcal{D}_2$ . That is, in Subproblems 1 and 2, for each point  $p \in P$ , we create a linear-size (perfect) hash table that maps each of the  $O(n)$  pre-computed query half-planes to its corresponding sum of distances. As a result, we can perform each query in  $O(1)$  time.

► **Theorem 3.** *The DMLS problem can be solved in  $O(n^2)$  time using  $O(n^2)$  space.*

Our algorithm matches in time complexity the lower bound of the medoid problem, as well as the fastest known algorithm for finding the median line in  $\mathbb{R}^2$  (i.e., the line having the minimal sum of distances from  $P$ ) [4]. Hence, we conjecture that our algorithm is optimal. Notice that, by allowing each point of  $P$  to be associated with a positive weight, we can generalize our problem to one of minimizing the sum of weighted distances. The algorithms proposed herein can be directly extended to solve the weighted problem with the same time and space bound. Finally, it is worth noting, for the sake of completeness, that we have also derived an  $O(n^2)$ -time algorithm for solving the discrete center line segment problem.

---

## References

---

- 1 Bernard Chazelle, Leo J Guibas, and Der Tsai Lee. The power of geometric duality. *BIT Numerical Mathematics*, 25(1):76–90, 1985.
- 2 Mark de Berg, Otfried Cheong, Marc van Kreveld, and Mark Overmars. *Computational Geometry: Algorithms and Applications*. Springer Berlin Heidelberg, 2008.
- 3 Herbert Edelsbrunner, Joseph O'Rourke, and Raimund Seidel. Constructing arrangements of lines and hyperplanes with applications. *SIAM Journal on Computing*, 15(2):341–363, 1986.
- 4 Nickolay M Korneenko and Horst Martini. Approximating finite weighted point sets by hyperplanes. In *Scandinavian Workshop on Algorithm Theory*, pages 276–286. Springer, 1990.
- 5 James Newling and François Fleuret. A sub-quadratic exact medoid algorithm. In *Artificial Intelligence and Statistics*, pages 185–193. PMLR, 2017.

# Perfect Matchings with Crossings

Oswin Aichholzer  

Institute of Software Technology, Graz University of Technology, Austria

Ruy Fabila-Monroy  

Departamento de Matemáticas, Cinvestav, Ciudad de México, México

Philipp Kindermann  

Fachbereich IV – Informatikwissenschaften, Universität Trier, Germany

Irene Parada  

Department of Mathematics and Computer Science, TU Eindhoven, The Netherlands

Rosna Paul  

Institute of Software Technology, Graz University of Technology, Austria

Daniel Perz  

Institute of Software Technology, Graz University of Technology, Austria

Patrick Schnider   

Department of Mathematical Sciences, University of Copenhagen, Denmark

Birgit Vogtenhuber  

Institute of Software Technology, Graz University of Technology, Austria

---

## Abstract

In this paper, we analyze the number of straight-line perfect matchings with  $k$  crossings on point sets of size  $n = 2m$  in general position. We show that for every  $k \leq 5n/8 - \Theta(1)$ , every  $n$ -point set admits a perfect matching with exactly  $k$  crossings and that there exist  $n$ -point sets where every perfect matching has fewer than  $5n^2/72$  crossings. We also study the number of perfect matchings with at most  $k$  crossings. Finally we show that convex point sets maximize the number of perfect matchings with  $\binom{n/2}{2}$  crossings and  $\binom{n/2}{2} - 1$  crossings.

**2012 ACM Subject Classification** Mathematics of computing → Combinatorics; Mathematics of computing → Graph theory

**Keywords and phrases** Perfect matchings, Crossings, Combinatorial geometry, Order types

**Digital Object Identifier** 10.4230/LIPIcs.CVIT.2016.23

**Funding** *Oswin Aichholzer*: Partially supported by the Austrian Science Fund (FWF): W1230 and the European Union H2020-MSCA-RISE project 73499 - CONNECT.

*Ruy Fabila-Monroy*: Partially supported by CONACYT (Mexico), grant 253261

*Irene Parada*: Partially supported by Austrian Science Fund within the collaborative DACH project *Arrangements and Drawings* as FWF project I 3340-N35.

*Rosna Paul*: Supported by the Austrian Science Fund (FWF): W1230.

*Daniel Perz*: Supported by the Austrian Science Fund within the collaborative DACH project *Arrangements and Drawings* as FWF project I 3340-N35.

*Patrick Schnider*: Has received funding from the European Research Council under the European Unions Seventh Framework Programme ERC Grant agreement ERC StG 716424 - CASE.

*Birgit Vogtenhuber*: Partially supported by Austrian Science Fund within the collaborative DACH project *Arrangements and Drawings* as FWF project I 3340-N35.

---

This is an abstract of a presentation given at CG:YRF 2021. 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.

**Acknowledgements** Research on this work has been initiated at the 16th European Research Week on Geometric Graphs which was held from November 18 to 22, 2019, near Strobl (Austria). We thank all participants for the good atmosphere as well as for discussions on the topic. Further, we thank Clemens Huemer for bringing this problem to our attention in the course of a meeting of the H2020-MSCA-RISE project 73499 - CONNECT.

## 1 Introduction

The question of how many different plane (crossing-free) straight-line perfect matchings can be drawn on a point set  $P$  in general position has been extensively studied [4, 5, 7, 11, 12]. It is known that point sets in convex position (for short, convex point sets) with  $n = 2m$  points admit exactly  $C_m$  plane perfect matchings, where  $C_m = \frac{1}{m+1} \binom{2m}{m} \in 2^{\Theta(n)}$  is the  $m$ th *Catalan number*. Further, for general  $n$ -point sets,  $C_m$  is a lower bound. If we allow crossings, then we can draw every possible perfect matching, and on  $n$  vertices there exist  $(n-1)!! \in 2^{\Theta(n \log n)}$  of them. Additionally, any such matching has at most  $\binom{n/2}{2} \in O(n^2)$  crossings. However, not much is known about the number or existence of straight-line perfect matchings with  $k$  crossings. For convex point sets, there are several results on the distribution of crossings over all perfect matchings [6, 9, 10]. Considering general point sets, Pach and Solymosi [8] characterize which point sets admit perfect matchings with  $\binom{n/2}{2}$  crossings.

In this work, we analyze the number of straight-line perfect matchings with exactly or at most  $k$  crossings that a point set can admit. All point sets are in general position and have an even number of points. Further,  *$k$ -crossing matchings* and  $\leq k$ -crossing matchings refer to perfect matchings with exactly  $k$  and at most  $k$  crossings, respectively. We denote by  $\text{pm}_k(P)$  the number of  $k$ -crossing matchings on a point set  $P$ , by  $\text{pm}_k^{\max}(n)$  the maximum of  $\text{pm}_k(P)$ , taken over all  $n$ -point sets  $P$ , and by  $\text{pm}_k^{\min}(n)$  the minimum of  $\text{pm}_k(P)$ , also taken over all  $n$ -point sets  $P$ . Similarly, we denote with  $\text{pm}_{\leq k}(P)$  the number of  $\leq k$ -crossing matchings on a point set  $P$  and let  $\text{pm}_{\leq k}^{\max}(n)$  and  $\text{pm}_{\leq k}^{\min}(n)$  be defined analogously as before. Finally,  $\text{pm}_k^{\text{conv}}(n)$  is the number of  $k$ -crossing matchings on a convex  $n$ -point set.

The complete proofs for all statements will appear in a future full version of this paper.

## 2 Exactly $k$ crossings

Our first two theorems imply that for every  $n$ -point set  $P$  and every  $k \in \{0, \dots, \frac{5n}{8} - \Theta(1)\}$ ,  $P$  admits a  $k$ -crossing matching, while this is not the case for  $k \geq \frac{5n^2}{72}$ ; see also Figure 1.

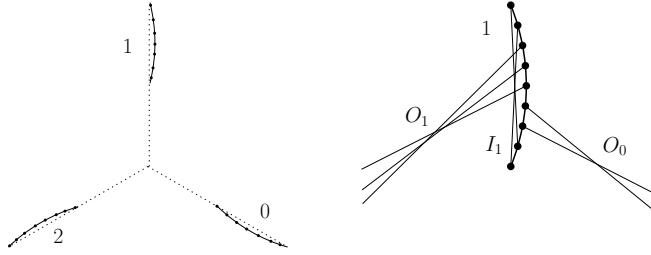
► **Theorem 1.** *For  $0 \leq k \leq \frac{5n}{8} - \Theta(1)$  crossings, it holds that  $\text{pm}_k^{\min}(n) \geq 1$ .*

**Proof sketch.** The following statements can be shown by exhaustive computations:

- (S1) For any  $k \leq 2$  every 8-point set admits a  $k$ -crossing matching.
- (S2) For any  $k \leq 3$ , every 10-point set admits a  $k$ -crossing matching.
- (S3) Every 15-point set admits a matching with 4 edges and 6 crossings [3].

To prove Theorem 1, we process  $P$  from left to right in groups of 32 points. Using (S1)–(S3), we find a perfect matching for each group with at least 20 crossings. We iterate this step, as long as the total number of crossings is at most  $k$ , and need at most  $\frac{32}{20}k = \frac{8}{5}k < n$  points. The remaining points are then used to fine tune the number of crossings. ◀

► **Theorem 2.** *For  $k \geq \frac{5n^2}{72}$  crossings, it holds that  $\text{pm}_k^{\min}(n) = 0$ .*



**Figure 1** Illustration for the proof of Theorem 2: A point set  $P$  for which every perfect matching has  $< \frac{5n^2}{72}$  crossings (left). Interior ( $I_1$ ) and outgoing ( $O_0, O_1$ ) matching edges for wing 1 of  $P$  (right).

### 3 At most $k$ crossings

We next show that if  $k$  is super-linear in  $n$ , then the number of  $\leq k$ -crossing matchings is super-exponential for every  $n$ -point set.

► **Theorem 3.** *For  $k \in \omega(n)$  crossings, it holds that  $\text{pm}_{\leq k}^{\min}(n) \in 2^{\Omega(n \log(\frac{k}{n}))}$ .*

**Proof sketch.** We split the  $n$  points to  $\frac{n^2}{k}$  disjoint groups of size  $\frac{k}{n}$ . Any combination of perfect matchings on those groups has at most  $\frac{n^2}{k} \cdot \frac{k^2}{n^2} = k$  crossings. The bound then follows from a lower bound on the number of perfect matchings on  $\frac{k}{n}$  points. ◀

Note that for a  $\leq k$ -crossing matching, at most  $4k$  points can be incident to crossing edges. Hence, the next theorem implies the upper bound on  $\text{pm}_{\leq k}^{\max}(n)$  stated in Corollary 5.

► **Theorem 4.** *For an  $n$ -point set  $P$  and  $0 \leq x \leq n$ , let  $\text{pm}^x(n)$  be the number of perfect matchings whose crossing edges are incident to at most  $x$  points. Then  $\text{pm}^x(n) \in 2^{O(n+x \log x)}$ .*

**Proof sketch.** The bound is obtained by combining the upper bound on the number of perfect matchings on  $x$  points, with an upper bound on the number of plane perfect matchings on  $n - x$  points and with the number of possible subsets of  $P$  of size  $x$ . ◀

► **Corollary 5.**  $\text{pm}_{\leq k}^{\max}(n) \in 2^{O(n+k \log k)}$ .

For  $k \in \Omega(n)$ , this bound is worse than the trivial upper bound from the number of all perfect matchings. For  $k \in O(\frac{n}{\log n})$  we get a bound of  $2^{O(n)}$ , which is asymptotically tight.

### 4 Convex position

We have  $\text{pm}_k^{\min}(n) \leq \text{pm}_k^{\text{conv}}(n) \leq \text{pm}_k^{\max}(n)$ . It is well known that the convex sets minimize the number of plane perfect matchings; see for example [2, 7]. Hence, we have  $\text{pm}_0^{\min}(n) = \text{pm}_0^{\text{conv}}(n)$ . On the other hand, considering the maximum number  $\mu := \binom{n/2}{2}$  of crossings, we can show that for  $k \in \{\mu, \mu-1\}$ , convex sets maximize the number of different  $k$ -crossing matchings, and that all  $n$ -point sets achieving these maximum numbers have exactly  $\frac{n}{2}$  halving edges (edges that have  $\frac{n-2}{2}$  points of the set on each side of their supporting line). The result for  $k = \mu$  is a direct consequence of Theorems 1 and 2 in [8].

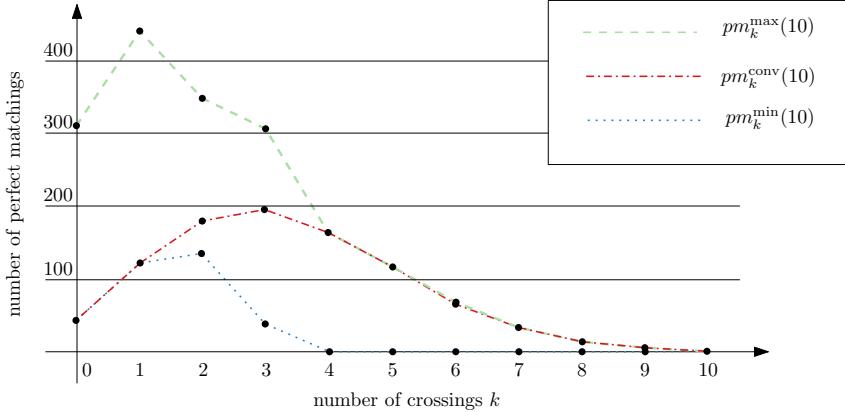
► **Proposition 6.** *For  $\mu = \binom{n/2}{2}$  crossings, it holds that  $\text{pm}_\mu^{\text{conv}}(n) = \text{pm}_\mu^{\max}(n) = 1$ .*

► **Theorem 7.** *For  $n \geq 6$  and  $\mu-1 = \binom{n/2}{2} - 1$  crossings, it holds that*

- (1)  $\text{pm}_{\mu-1}^{\text{conv}}(n) = \text{pm}_{\mu-1}^{\max}(n) = \frac{n}{2}$ , obtained by any  $n$ -point set with exactly  $\frac{n}{2}$  halving edges.
- (2) Any  $n$ -point set  $P$  with more than  $\frac{n}{2}$  halving edges has  $\text{pm}_{\mu-1}(P) \leq 2$ .

**Proof sketch.** Any matching with  $\mu - 1$  crossings has exactly one pair of non-crossing edges. The endpoints of these edges can be either in convex position (Case 1) or not (Case 2). It can be shown that Cases 1 and 2 match statements (1) and (2) of the theorem, respectively.  $\blacktriangleleft$

In ongoing work, we investigate for which values of  $k$  and  $n$  it holds that  $\text{pm}_k^{\text{conv}}(n) \in \{\text{pm}_k^{\min}(n), \text{pm}_k^{\max}(n)\}$ ; exhaustive computations for all point sets of small size indicate that this might be true for more than just  $k \in \{0, \mu - 1, \mu\}$ ; Figure 2 shows the results for  $n = 10$ .



**Figure 2** Comparison of  $\text{pm}_k^{\min}(n)$ ,  $\text{pm}_k^{\max}(n)$ , and  $\text{pm}_k^{\text{conv}}(n)$  for  $n = 10$  and  $0 \leq k \leq 10$ .

---

## References

---

- 1 Oswin Aichholzer, Franz Aurenhammer, and Hannes Krasser. Enumerating order types for small point sets with applications. *Order*, 19:265–281, 2002.
- 2 Oswin Aichholzer, Thomas Hackl, Clemens Huemer, Ferran Hurtado, Hannes Krasser, and Birgit Vogtenhuber. On the number of plane geometric graphs. *Graphs and Combinatorics*, 23(1):67–84, 2007.
- 3 Oswin Aichholzer, Jan Kynčl, Manfred Scheucher, and Birgit Vogtenhuber. On 4-crossing-families in point sets and an asymptotic upper bound. In *Proceedings of the 37th European Workshop on Computational Geometry (EuroCG 2021)*, 2021. To appear.
- 4 Andrei Asinowski. The number of non-crossing perfect plane matchings is minimized (almost) only by point sets in convex position. *arXiv preprint arXiv:1502.05332*, 2015.
- 5 Andrei Asinowski and Günter Rote. Point sets with many non-crossing perfect matchings. *Computational Geometry*, 68:7–33, 2018.
- 6 Philippe Flajolet and Marc Noy. Analytic combinatorics of chord diagrams. In *Formal Power Series and Algebraic Combinatorics*, pages 191–201. Springer, 2000.
- 7 Alfredo García, Marc Noy, and Javier Tejel. Lower bounds on the number of crossing-free subgraphs of  $K_n$ . *Computational Geometry*, 16(4):211–221, 2000.
- 8 János Pach and József Solymosi. Halving lines and perfect cross-matchings. *Advances in Discrete and Computational Geometry*, 223:245–249, 1999.
- 9 Vincent Pilaud and Juanjo Rue. Analytic combinatorics of chord and hyperchord diagrams with  $k$  crossings. *Advances in Applied Mathematics*, 57:60–100, 2014.
- 10 John Riordan. The distribution of crossings of chords joining pairs of  $2n$  points on a circle. *Mathematics of Computation*, 29(129):215–222, 1975.
- 11 Micha Sharir and Emo Welzl. On the number of crossing-free matchings, cycles, and partitions. *SIAM Journal on Computing*, 36(3):695–720, 2006.
- 12 Chenchao You. Improving Sharir and Welzl’s bound on crossing-free matchings through solving a stronger recurrence. *arXiv preprint arXiv:1701.05909*, 2017.

# Triangulations of Exotic 4-Manifolds

Rhuaidi A. Burke

School of Mathematics and Physics, University of Queensland, Australia  
rhuaidi.burke@uq.edu.au

Benjamin A. Burton

School of Mathematics and Physics, University of Queensland, Australia  
bab@maths.uq.edu.au

---

## Abstract

---

Dimension 4 is the first dimension in which exotic smooth manifold pairs appear — manifolds which are topologically the same but for which there is no smooth deformation of one into the other. Despite there being numerous examples of exotic 4-manifolds in the smooth category, explicit triangulations of these manifolds have been nearly non-existent. We present an implementation of an algorithm to systematically generate triangulations of a particular class of 4-manifolds. Using this implementation, we present new triangulations of four different exotic 4-manifold pairs.

**2012 ACM Subject Classification** Mathematics of computing → Continuous mathematics → Topology → Geometric topology

**Keywords and phrases** Exotic 4-manifolds, triangulations, handle decomposition, Kirby diagram, edge-coloured graph

## 1 Introduction

A pair of smooth manifolds  $X_1, X_2$  are *exotic* if they are homeomorphic but not diffeomorphic.

One of the great remaining open problems of classical topology is the smooth 4-dimensional Poincaré conjecture, which asks whether or not there exist exotic 4-spheres.

Every smooth manifold can be triangulated (i.e. represented as a PL-manifold) [17, 4], and the converse holds in dimensions  $\leq 6$  [12]. Consequently, there is hope that one can study smooth structures on 4-manifolds via combinatorial and computational techniques.

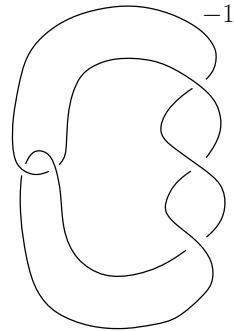
For  $0 \leq i \leq 4$ , a *4-dimensional  $i$ -handle* is a copy of  $D^i \times D^{4-i}$ , attached to a smooth 4-manifold  $W$  via an embedding  $\varphi : \mathbb{S}^{i-1} \times D^{4-i} \rightarrow \partial W$ . A *2-handlebody* here refers to a 4-manifold obtained by attaching 2-handles to  $D^4$ . The attaching map for a single 2-handle in a 2-handlebody is of the form  $\varphi : \mathbb{S}^1 \times D^2 \rightarrow \mathbb{S}^3$ . The image of  $\varphi(\mathbb{S}^1 \times \{0\})$  can be understood as a knot  $K$  in  $\mathbb{S}^3$ . We then “thicken” this cell to be 4-dimensional by specifying a *framing* of  $K$  (i.e. a trivialisation of the normal disk bundle  $\nu K$  in  $\mathbb{S}^3$ ) which describes how the  $D^2$  factor of  $\mathbb{S}^1 \times D^2$  “twists” around  $K$ . The set of all possible framings on  $K$  is in bijection with  $\pi_1(SO(2)) \cong \mathbb{Z}$  [2, 11]. If there are multiple 2-handles, we can visualise a 2-handlebody as a link  $L$  in  $\mathbb{S}^3$ , with an integer attached to each component of  $L$  specifying the framing.

► **Theorem 1.** *The manifolds in Figures 1, 2, 3, and 4, are each respectively exotic pairs.*

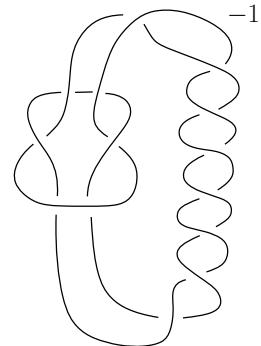
The manifolds which appear in the aforementioned figures are respectively exotic pairs of: homotopy  $\overline{\mathbb{CP}}_0^2 = \mathbb{CP}^2 - \text{int}(D^4)$ , originally due to Akbulut [1]; homotopy  $\mathbb{S}^2 \times D^2 - D^4$ , originally due to Yasui [18]; the simplest example of the so-called “nuclei of elliptic surfaces” due to Gompf [10]; and finally, a pair of homotopy  $\mathbb{CP}^2 \# 2\overline{\mathbb{CP}}^2 = \text{int}(D^4)$  due to Naoe [13]. By a “homotopy  $X$ ”, we mean a manifold which is homotopy equivalent to  $X$ .

This is an abstract of a presentation given at CG:YRF 2021. 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.

## Exotic 4-Manifold Triangulations

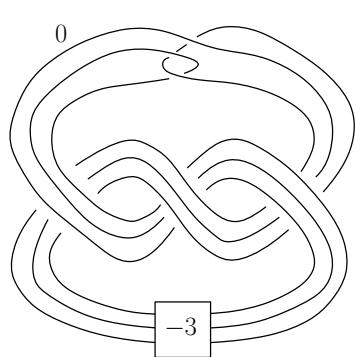


$A_1$

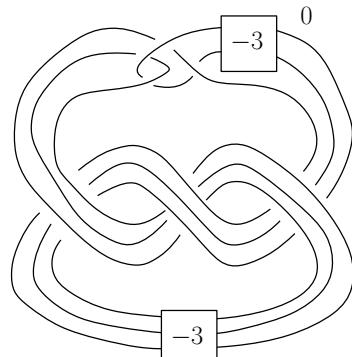


$A_2$

■ **Figure 1** Akbulut Exotic Homotopy  $\overline{\mathbb{CP}}_0^2$ .

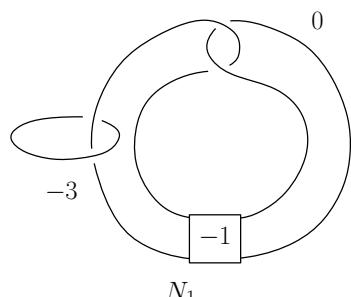


$Y_1$

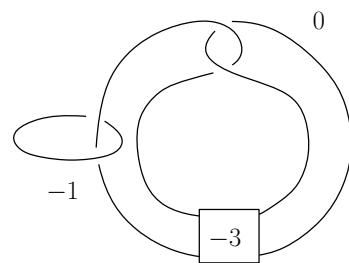


$Y_2$

■ **Figure 2** Yasui Exotic Homotopy  $\mathbb{S}^2 \times D^2 - D^4$ .

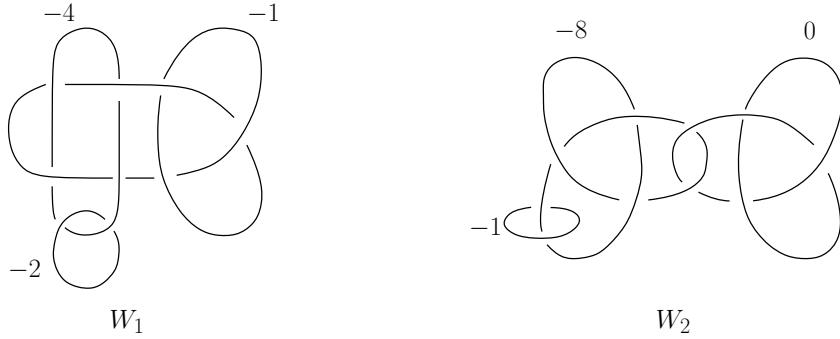


$N_1$



$N_2$

■ **Figure 3** The simplest Gompf Exotic Nuclei.



■ **Figure 4** Naoe Exotic Homotopy  $\mathbb{CP}^2 \# 2\overline{\mathbb{CP}}^2 - \text{int}(D^4)$ .

## 2 Gems & Crystallisations

In what follows, the term *graph* is used to refer to finite multigraphs without loops. An *edge colouring* of a graph  $\Gamma = (V, E)$  is a surjective map  $\gamma : E \rightarrow \Delta_n$ , where  $\Delta_n = \{0, \dots, n\}$ , such that  $\gamma(e) \neq \gamma(f)$  whenever  $e$  and  $f$  share a common vertex.

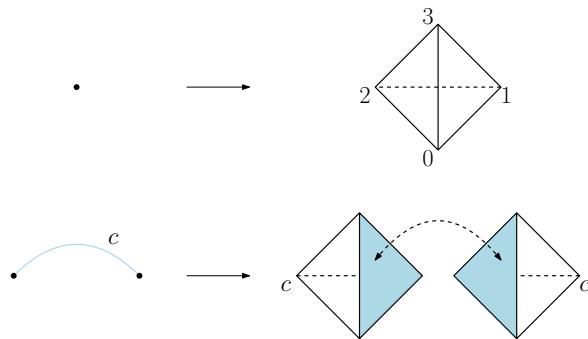
An  $(n+1)$ -coloured graph  $(\Gamma, \gamma)$  can be used to visualise an  $n$ -dimensional pseudocomplex  $K(\Gamma)$ , which is constructed according to the following procedure [9]:

1. For each vertex  $v \in V$ , take an  $n$ -simplex  $\sigma(v)$ , with its vertices labelled by  $0, 1 \dots, n$ .
2. For each  $c$ -coloured edge between  $v$  and  $w$  ( $v, w \in V$ ), identify the  $(n-1)$ -faces of  $\sigma(v)$  and  $\sigma(w)$  opposite from the vertices labelled by  $c$ , so that equally labelled vertices of  $\sigma(v)$  and  $\sigma(w)$  are identified.

This process is depicted for the three-dimensional case in Figure 5. If  $K(\Gamma)$  triangulates a PL  $n$ -manifold  $M$ , then  $(\Gamma, \gamma)$  is called a *gem* (graph encoded manifold) representing  $M$ . Figure 6 depicts a gem of the standard  $S^4$ , and how the resulting triangulation is obtained by identification of the facets of the two simplices  $x$  and  $y$ .

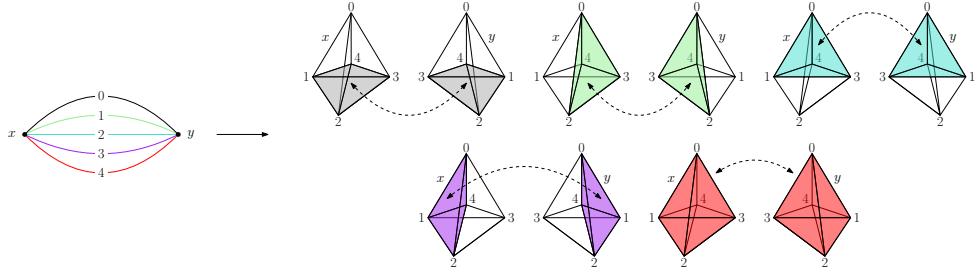
An  $(n+1)$ -coloured graph representing the  $n$ -manifold  $M$  is called a *crystallisation* of  $M$  if the subgraph  $\Gamma_j = (V(\Gamma), \gamma^{-1}(\Delta_n - \{j\}))$  is connected for each  $j \in \Delta_n$ .

► **Theorem 2.** *Every PL  $n$ -manifold admits a crystallisation representing it [14, 8].*



■ **Figure 5** Constructing a triangulation from a coloured graph.

## Exotic 4-Manifold Triangulations



**Figure 6** Left: A gem of  $S^4$ . Right: The coloured triangulation obtained from the gem.

### 3 The Algorithm

In [5, 6], Casali describes an algorithmic way to obtain a crystallisation of a 2-handlebody  $M = M_L$  directly from a framed link diagram  $L$  representing  $M$ . Though a theoretical description and proof of the validity of the construction is given, to the best of the author's knowledge, there was until now, no available software implementation. We implement Casali's original algorithm for 2-handlebodies using C++: we take as input a combinatorial encoding of a framed link diagram representing a 2-handlebody and, via the methodology of [5], return a triangulation of the associated manifold.

### 4 Results

Our implementation was able to reproduce the results presented in [5] and [6] in the form of the Poincaré Homology 3-sphere, obtained by +1-framed Dehn surgery along the trefoil knot and of  $S^2 \times S^2 - D^4$ . Our software also implements a *capping off* procedure [7], through which we were able to also reproduce (after simplifications in *Regina* [3]) specific pre-existing triangulations of  $\mathbb{CP}^2$  and  $S^2 \times S^2$  (cf. p. 16 of [15]).

Using our implementation of Casali's algorithm and *Regina*, we obtain triangulations of the manifold pairs  $(A_1, A_2)$ ,  $(Y_1, Y_2)$ ,  $(N_1, N_2)$ , and  $(W_1, W_2)$ , as depicted in Figures 1, 2, 3, and 4 respectively. Table 1 shows the combinatorial size information of the obtained triangulations. The isomorphism signatures are available at <https://raburke.github.io/>.

| Manifold | Vertices | Edges | Triangles | Tetrahedra | Pentachora |
|----------|----------|-------|-----------|------------|------------|
| $A_1$    | 5        | 72    | 206       | 207        | 70         |
| $A_2$    | 5        | 156   | 458       | 459        | 154        |
| $Y_1$    | 5        | 440   | 1310      | 1311       | 438        |
| $Y_2$    | 5        | 508   | 1514      | 1515       | 506        |
| $N_1$    | 7        | 88    | 254       | 258        | 88         |
| $N_2$    | 7        | 144   | 422       | 426        | 144        |
| $W_1$    | 7        | 62    | 228       | 280        | 112        |
| $W_2$    | 7        | 64    | 236       | 290        | 116        |

**Table 1** Size of (unsimplified) triangulations.

---

### References

---

- 1 Selman Akbulut. An exotic 4-manifold. *Journal of Differential Geometry*, 33(2):357–361, 1991.
- 2 Selman Akbulut. *4-manifolds*, volume 25. Oxford University Press, 2016.

- 3 Benjamin A. Burton, Ryan Budney, William Pettersson, et al. Regina: Software for low-dimensional topology. <http://regina-normal.github.io/>, 1999–2019.
- 4 Stewart S. Cairns. A simple triangulation method for smooth manifolds. *Bull. Amer. Math. Soc.*, 67:389–390, 1961.
- 5 Maria Rita Casali. From framed links to crystallizations of bounded 4-manifolds. *J. Knot Theory Ramifications*, 9(4):443–458, 2000.
- 6 Maria Rita Casali. Dotted links, Heegaard diagrams, and colored graphs for PL 4-manifolds. *Rev. Mat. Complut.*, 17(2):435–457, 2004.
- 7 Maria Rita Casali and Paola Cristofori. Kirby diagrams and 5-colored graphs representing compact 4-manifolds, 2021.
- 8 Alberto Cavicchioli and Carlo Gagliardi. Crystallizations of pl-manifolds with connected boundary. *Bollettino della Unione Matematica Italiana. Series V. B*, 01 1980.
- 9 M. Ferri, C. Gagliardi, and L. Grasselli. A graph-theoretical representation of PL-manifolds—a survey on crystallizations. *Aequationes Math.*, 31(2-3):121–141, 1986.
- 10 Robert E. Gompf. Nuclei of elliptic surfaces. *Topology*, 30(3):479–511, November 1991.
- 11 Robert E. Gompf and András I. Stipsicz. *4-manifolds and Kirby calculus*, volume 20 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, 1999.
- 12 Morris W. Hirsch and Barry Mazur. *Smoothings of piecewise linear manifolds*. Princeton University Press, Princeton, N. J.; University of Tokyo Press, Tokyo, 1974. Annals of Mathematics Studies, No. 80.
- 13 Hironobu Naoe. Corks with large shadow-complexity and exotic four-manifolds. *Experimental Mathematics*, 0(0):1–15, 2019.
- 14 Mario Pezzana. Sulla struttura topologica delle varietà compatte. *Atti Sem. Mat. Fis. Univ. Modena*, 23(1):269–277, 1974.
- 15 Jonathan Spreer and Stephan Tillmann. Determining the trisection genus of orientable and non-orientable pl 4-manifolds through triangulations. *Experimental Mathematics*, 0(0):1–11, 2020.
- 16 The Sage Developers. *SageMath, the Sage Mathematics Software System (Version 8.6)*, 2019. <https://www.sagemath.org>.
- 17 J. H. C. Whitehead. On  $C^1$ -complexes. *Ann. of Math. (2)*, 41:809–824, 1940.
- 18 Kouichi Yasui. Corks, exotic 4-manifolds and knot concordance. *arXiv preprint arXiv:1505.02551*, 2015.

# Homotopical decompositions of simplicial and Vietoris-Rips complexes

Wojciech Chacholski 

Mathematics, KTH, S-10044, Stockholm, Sweden

Alvin Jin 

Mathematics, KTH, S-10044, Stockholm, Sweden

Martina Scolamiero 

Mathematics, KTH, S-10044, Stockholm, Sweden

Francesca Tombari 

Mathematics, KTH, S-10044, Stockholm, Sweden

---

## Abstract

Motivated by applications in topological data analysis, we consider decompositions of simplicial complexes based on coverings of their vertices. We aim to study under what circumstances the global homotopy type of a simplicial complex can be inferred from the local homotopy type of its components. After addressing this problem in general, we specialise it to Vietoris-Rips complexes.

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

**Keywords and phrases** Vietoris-Rips complexes, Metric gluings, Closed classes, Homotopy push-outs

**Digital Object Identifier** 10.4230/LIPIcs.CVIT.2016.23

**Related Version** Full version hosted on arXiv

*Full Version:* <https://arxiv.org/abs/2002.03409>

**Funding** Wojciech Chacholski: Partially supported by VR and WASP

Alvin Jin: Supported by WASP

Martina Scolamiero: Supported by WASP

Francesca Tombari: Supported by WASP

**Acknowledgements** This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

## 1 Introduction

This note is based on the results presented in [6]. In topological data analysis the input is commonly given by a pseudometric space. This spacial information is then converted into geometrical information by constructing a filtered simplicial complex depending on the underlying pseudometric. The geometrical information that they summarise leads to the definition of global homology-based invariants describing the dataset, such as persistent homology [4, 8, 9, 11, 12], barcodes [5], stable rank [7, 15] and persistence landscape [3]. Their global nature prohibits the computation of such invariants for large datasets. However, decomposing the dataset might facilitate the step from local to global analysis, which would allow parallelisation of computations [14] (see also [10, 13] for related work). The issue is that this step is not always possible, since in general the homotopy type of a simplicial complex

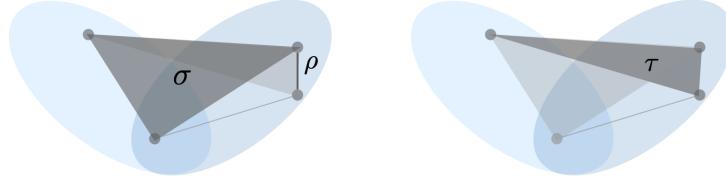
---

This is an abstract of a presentation given at CG:YRF 2021. 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.

and of its decomposition is different. Our work aims to study this difference and under what circumstances it is possible to retrieve the homology of a simplicial complex from what we called its data-driven decomposition. Although our techniques hold for generic simplicial complexes, we also provide some results about Vietoris-Rips complexes arising from distance spaces, pseudometric spaces and, in particular, metric gluings.

## 2 General case

Let  $K$  be a simplicial complex (also seen as a poset with the inclusion of simplices) and let  $K_0$  be the collection of its vertices. If  $K_0 = X \cup Y$ , with  $A = X \cap Y$ , then we can restrict  $K$  to  $X$  and  $Y$ , obtaining the two subcomplexes  $K_X$  and  $K_Y$ . Their union  $K_X \cup K_Y$  is still a subcomplex of  $K$ , that we refer to as its data-driven decomposition. Furthermore, for every simplex  $\sigma$  in  $K$  we define a collection  $St(\sigma, A)$  of non-empty subsets  $\mu$  of  $A$  such that  $\mu \cup \sigma$  forms a simplex in  $K$ . These  $St(\sigma, A)$  are simplicial complexes called obstruction complexes (Figure 2 shows an example). In order to study the map  $K_X \cup K_Y \hookrightarrow K$  we introduce two subposets of  $K$ : the collection of simplices  $\sigma$  in  $K$  such that  $\sigma \subset X$  or  $\sigma \subset Y$  or  $\sigma \cap A \neq \emptyset$ , denoted by  $P$ , and its complement  $K \setminus P$ , which is the collection of simplices  $\sigma$  in  $K$  such that  $\sigma \cap X \neq \emptyset$  and  $\sigma \cap Y \neq \emptyset$  and  $\sigma \cap A = \emptyset$  (see Figure 1). It is possible to show that  $K_X \cup K_Y \hookrightarrow P \hookrightarrow K$ , where the first poset functor is always a weak equivalence and the second has fibers having the same homotopy type of certain obstruction complexes.



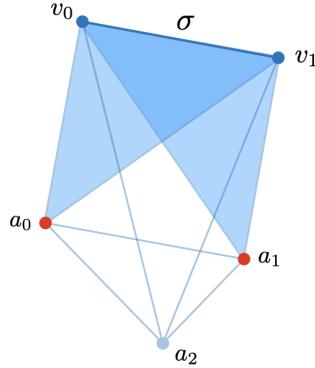
**Figure 1** A representation of a data-driven decomposition is given here. The simplices on the left,  $\sigma$  and  $\rho$ , are examples of elements in  $P$ . The simplex on the right,  $\tau$ , lies in  $K \setminus P$ .

We can now state our main result and an immediate corollary:

► **Theorem 1.** *Let  $\mathcal{C}$  be a closed collection of simplicial sets. Assume that, for every  $\sigma$  in  $K \setminus P$ , the obstruction complex  $St(\sigma, A)$  is in  $\mathcal{C}$ . Then the homotopy fibers of the inclusion  $K_X \cup K_Y \subset K$  are also in  $\mathcal{C}$ .*

In other words, our understanding of the inclusion map of a data-driven decomposition of  $K$  into  $K$  itself depends only on certain properties of the obstruction complexes associated to simplices in  $K \setminus P$ . The following are some particular cases of the above theorem specialised to different closed collections of simplicial sets.

1. If, for every  $\sigma$  in  $K \setminus P$ ,  $St(\sigma, A)$  is contractible, then  $K_X \cup K_Y \subset K$  is a weak equivalence.
2. If, for every  $\sigma$  in  $K \setminus P$ ,  $St(\sigma, A)$  is  $n$ -connected, then the homotopy fibers of  $K_X \cup K_Y \subset K$  are  $n$ -connected and this map induces an isomorphism on homotopy groups in degrees  $0, \dots, n$  and a surjection in degree  $n + 1$ .
3. If, for every  $\sigma$  in  $K \setminus P$ ,  $St(\sigma, A)$  is connected and has  $p$ -torsion reduced integral homology in degrees not exceeding  $n$ , for some  $p$  prime, then the homotopy fibers of  $K_X \cup K_Y \subset K$  are connected and have  $p$ -torsion reduced integral homology in degrees not exceeding  $n$ . Thus in this case, for prime  $q \neq p$ ,  $K_X \cup K_Y \subset K$  induces an isomorphism on  $H_*(-, \mathbf{Z}/q)$  for  $* \leq n$  and a surjection on  $H_{n+1}(-, \mathbf{Z}/q)$ .



**Figure 2** The simplicial complex  $K$  has five vertices in total. Consider  $X = \{v_0, a_0, a_1, a_2\}$  and  $Y = \{v_1, a_0, a_1, a_2\}$  subsets of  $K_0$ . Then the set  $A = X \cap Y$  consists of three points  $\{a_0, a_1, a_2\}$ . The obstruction complex  $\text{St}(\sigma, A)$  contains only vertices  $a_0$  and  $a_1$ , but not the edge  $\{a_0, a_1\}$ .

### 3 Vietoris-Rips complexes

Given a distance space  $(Z, d)$  ( $d(x, x) = 0$  and  $d(x, y) = d(y, x)$ ) and a non-negative real number  $r$ , we can define the Vietoris-Rips complex at scale  $r$ , denoted by  $VR_r(Z)$ , as the simplicial complex whose simplices are subsets  $\sigma$  of  $Z$ , such that  $\text{diam}(\sigma) \leq r$  (i.e. for every  $x$  and  $y$  in  $Z$ ,  $d(x, y) \leq r$ ). In this case  $VR_r(Z)_X = VR_r(X)$ , where  $X$  is a subspace of  $Z$ . Hence, the inclusion seen in Section 2 assumes the form  $VR_r(X) \cup VR_r(Y) \hookrightarrow VR_r(Z)$ , where  $Z = X \cup Y$  and  $A = X \cap Y$ . The following result contains two assumptions that together guarantee contractibility and non-emptiness of obstruction complexes.

► **Proposition 2.** *Assume that  $A$  is non-empty and for every  $x$  in  $X \setminus A$ ,  $y$  in  $Y \setminus A$ , and  $v$  in  $A$ , the following inequalities hold:*

- $d(x, y) \geq d(x, v)$  and  $d(x, y) \geq d(y, v)$ ,
- $d(x, y) \geq \text{diam}(A)$ .

*Then  $VR_r(X) \cup VR_r(Y) \hookrightarrow VR_r(Z)$  is a weak equivalence for all  $r$  in  $[0, \infty)$ .*

Consider now a pseudometric space  $(Z, d_Z)$  (distance space with  $d_Z(x, z) \leq d_Z(x, y) + d_Z(y, z)$ ) obtained by gluing together two intersecting pseudometric spaces  $(X, d_X)$  and  $(Y, d_Y)$ . Here the pseudometric is given by

$$d_Z(z, z') = \begin{cases} d_X(z, z') & \text{if } z, z' \in X \\ d_Y(z, z') & \text{if } z, z' \in Y \\ \inf\{d(z, a) + d(z', a) \mid a \in A\} & \text{if } z \in X \setminus A \text{ and } z' \in Y \setminus A \end{cases}$$

A space obtained in this way is called a metric gluing (see [1, 2]). For this kind of spaces we can state the following result, where the assumption ensures contractibility of all obstruction complexes associated to simplices  $\sigma$  in  $VR_r(Z) \setminus P$  such that either  $|\sigma \cap X| = 1$  or  $|\sigma \cap Y| = 1$ .

► **Proposition 3.** *Let  $r$  be in  $[0, \infty)$ . For any vertex  $v$  in an edge  $\sigma$  in  $VR_r(Z) \setminus P$ , if  $a$  and  $b$  are elements in  $A$  such that  $d(a, v) \leq r$  and  $d(v, b) \leq r$ , then  $2d(a, b) \leq d(a, v) + d(v, b)$ . Then the homotopy fibers of the inclusion  $VR_r(X) \cup VR_r(Y) \hookrightarrow VR_r(Z)$  are simply connected and this map induces an isomorphism on  $\pi_0$  and  $\pi_1$  and a surjection on  $\pi_2$ .*

---

**References**

---

- 1 Michał Adamaszek, Henry Adams, Ellen Gasparovic, Maria Gommel, Emilie Purvine, Radmila Sazdanovic, Bei Wang, Yusu Wang, and Lori Ziegelmeier. Vietoris-Rips and Čech complexes of metric gluings. In *34th International Symposium on Computational Geometry*, volume 99 of *LIPICS. Leibniz Int. Proc. Inform.*, pages Art. No. 3, 15.
- 2 Michał Adamaszek, Henry Adams, Ellen Gasparovic, Maria Gommel, Emilie Purvine, Radmila Sazdanovic, Bei Wang, Yusu Wang, and Lori Ziegelmeier. On homotopy types of Vietoris-Rips complexes of metric gluings. *J. Appl. Comput. Topol.*, 4(3):425–454, 2020. doi:10.1007/s41468-020-00054-y.
- 3 Peter Bubenik. Statistical topological data analysis using persistence landscapes. *J. Mach. Learn. Res.*, 16:77–102, 2015.
- 4 Francesca Cagliari, Massimo Ferri, and Paola Pozzi. Size functions from a categorical viewpoint. *Acta Appl. Math.*, 67(3):225–235, 2001.
- 5 Gunnar Carlsson, Afra Zomorodian, Anne Collins, and Leonidas Guibas. Persistence barcodes for shapes. *International Journal of Shape Modeling*, 11(2):149–187, 2005.
- 6 Wojciech Chacholski, Alvin Jin, Martina Scolamiero, and Francesca Tombari. Homotopical decompositions of simplicial and Vietoris Rips complexes. *To appear in Journal of Applied and Computational Topology*, 2021.
- 7 Wojciech Chacholski and Henri Riihimäki. Metrics and Stabilization in One Parameter Persistence. *SIAM J. Appl. Algebra Geom.*, 4(1):69–98, 2020.
- 8 Cecil Jose A. Delfinado and Herbert Edelsbrunner. An incremental algorithm for betti numbers of simplicial complexes on the 3-sphere. *Comput. Aided Geom. Design*, 12:771–784, 1995.
- 9 Herbert Edelsbrunner and John Harer. Persistent homology—a survey. In *Surveys on discrete and computational geometry*, volume 453 of *Contemp. Math.*, pages 257–282.
- 10 Herbert Edelsbrunner and Michael Kerber. Alexander duality for functions: the persistent behavior of land and water and shore. In Tamal K. Dey and Sue Whitesides, editors, *Symposium on Computational Geometry 2012, SoCG '12, Chapel Hill, NC, USA, June 17-20, 2012*, pages 249–258. ACM, 2012. doi:10.1145/2261250.2261287.
- 11 Massimo Ferri. Size functions: a new topological approach to shape analysis. In *Geometry and topology of submanifolds, VII (Leuven, 1994/Brussels, 1994)*, pages 299–304. World Sci. Publ., River Edge, NJ, 1995.
- 12 Patrizio Frosini and Claudia Landi. Size theory as a topological tool for computer vision. *Pattern Recognition and Image Analysis*, 9:596–603, 1999.
- 13 Michael Kerber and Hannah Schreiber. Barcodes of towers and a streaming algorithm for persistent homology. *Discret. Comput. Geom.*, 61(4):852–879, 2019. doi:10.1007/s00454-018-0030-0.
- 14 Ryan Lewis and Dmitriy Morozov. Parallel computation of persistent homology using the blowup complex. In Guy E. Blelloch and Kunal Agrawal, editors, *Proceedings of the 27th ACM on Symposium on Parallelism in Algorithms and Architectures, SPAA 2015, Portland, OR, USA, June 13-15, 2015*, pages 323–331. ACM, 2015. doi:10.1145/2755573.2755587.
- 15 Martina Scolamiero, Wojciech Chacholski, Anders Lundman, Ryan Ramanujam, and Sebastian Öberg. Multidimensional persistence and noise. *Foundations of Computational Mathematics*, 17(6):1367–1406, 2017. doi:10.1007/s10208-016-9323-y.

# Algorithms For Max Cut on Unit Interval and Laminar Interval Graphs

Joshi Utkarsh 

Department of Computer Science and Automation, Indian Institute of Science, India

Saladi Rahul 

Department of Computer Science and Automation, Indian Institute of Science, India

Josson Joe Thoppil 

Department of Information Technology, National Institute of Technology, Karnataka, Surathkal, India

---

## Abstract

In this project, we built algorithms for maximum cut on unit interval graphs and laminar interval graphs. We obtained an exact polynomial time algorithm for laminar interval graphs and a near-linear time 0.66 approximation algorithm for unit interval graphs.

**2012 ACM Subject Classification** Theory of computation → Computational geometry

**Keywords and phrases** Maximum cut, unit interval graphs, laminar interval graphs, approximation algorithms

## 1 Introduction

In the *maximum cut* (a.k.a., max cut) problem, the input is an undirected graph  $G = (V, E)$  where  $V$  is the set of vertices and  $E$  is the set of edges, and the goal is to partition  $V$  into two disjoint sets such that the number of edges having their endpoints in different sets is maximized. Max cut is an NP-hard problem. The celebrated Goemans and Williamson's SDP based approximation algorithm for the max cut problem gives an approximation ratio of  $\approx 0.878$  [1].

In this project, we are interested in studying the max cut problem for *interval graphs*. Formally, let  $I = \{I_1, \dots, I_n\}$  be a set of  $n$  intervals on the real line. The interval graph corresponding to  $I$  has a vertex corresponding to each interval in  $I$  and an edge is placed between two vertices if and only if the corresponding intervals intersect.

At SoCG'21, Adhikary *et al.* [2] will be presenting a result which shows that max cut is NP-complete *even* for interval graphs. Unit interval graphs are a special case of interval graphs where each interval has unit length. An interesting open problem has been to resolve the complexity status of max cut on unit interval graphs. Two previous results [3, 4] claimed a polynomial time algorithm for unit interval graphs, but they were reported as flawed later [5, 6].

Since we are dealing with interval graphs which have more structure than general graphs, the following questions are of interest:

1. Are there special cases of interval graphs which can be solved exactly in polynomial time?
2. A simple linear-time randomized algorithm exists to solve max cut on general graphs with an expected approximation factor of 0.5 (randomly throw each vertex in  $V$  into one of the two sets). For interval graphs or its special cases, is it possible to design near-linear time algorithms (in terms of  $n$ ) with an approximation factor strictly greater than 0.5?

---

This is an abstract of a presentation given at CG:YRF 2021. 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.

3. Can an approximation factor better than 0.878 be obtained for interval graphs?

In our on going project, we have made progress on the first two questions posed above:

1. For laminar intervals (which are defined later in Section 3) we obtain an exact solution in polynomial time.
2. For unit intervals, we obtain an  $O(n \log n)$  time algorithm with an approximation factor of 0.66.

## 2 Unit interval graphs

In a *unit interval graph* all the intervals on the real-line are of unit length. We obtain an  $O(n \log n)$  time algorithm with an approximation factor of 0.66. Let  $I_1, I_2, \dots, I_n$  be the  $n$  intervals in our input sorted in increasing order based on the coordinate value of the left endpoints. We partition  $I$  into *groups*  $\{S_1, S_2, \dots, S_m\}$  as follows:

- Let the intervals intersecting  $I_1$  be  $I_2, \dots, I_t$ . Then  $S_1$  consists of  $\{I_1, I_2, \dots, I_t\}$ .
- Recursively construct the remaining groups based on  $I_{t+1}, \dots, I_n$ ; stop when all the intervals have been assigned to some group.

Our grouping has the following two interesting properties: (i) the intervals belonging to the same group form a clique, and (ii) for all  $1 \leq i \leq m$ , since our inputs are unit intervals, any interval in  $S_i$  can only intersect with intervals within its group and its adjacent groups i.e.,  $S_{i-1}$  and  $S_{i+1}$  (if they exist).

Based on this grouping we will partition the intersections among intervals in  $I$  into *intra group intersections* and *inter group intersections*. For each intersection between two intervals in  $I$ , if the two intervals belong to same group (resp., different groups), then the intersection belongs to intra (resp., inter) group intersections.

In the interval graphs setting, max cut problem can be re-stated as follows: color each interval in  $I$  either red or green, so that the number of intersections among intervals having different colors is maximized. Our final algorithm will make use of one of the following two subroutines to color the intervals:

1. *Alternate interval coloring*: For any  $1 \leq i \leq m$ , let the intervals in group  $S_i$  be sorted in increasing order based on the coordinate value of their left endpoints. The intervals in this sorted sequence are colored alternatively: either (red, green, red, green, ...) or (green, red, green, red, ...).

As a base case,  $S_1$  can be colored using either of the two choices. Once group  $S_i$  is colored, to color  $S_{i+1}$  there are two choices. The choice which leads to more intersections between intervals from  $S_i$  and  $S_{i+1}$  with different colors is chosen.

2. *Alternate group coloring*: As the name suggests, all the intervals in an odd-indexed group (such as  $S_1, S_3, \dots$ ) are colored red, while the intervals in an even-indexed group (such as  $S_2, S_4, \dots$ ) are colored green.

The alternate interval coloring subroutine works well if the number of intra group intersections are dominant, whereas the alternate group coloring subroutine works well if the number of inter group intersections are dominant. Our final algorithm does the following: if the number of inter group intersections is at most  $2(\sum_{i=1}^m \lfloor |S_i|/2 \rfloor \times \lceil |S_i|/2 \rceil)$ , then run the alternate interval coloring subroutine; otherwise, run the alternate group coloring subroutine. Unfortunately, due to lack of space, we are omitting the analysis.

### 3 Laminar interval graphs

A *laminar interval graph* is a subclass of interval graphs in which if two intervals intersect, then one is completely contained in the other. We built an algorithm based on dynamic programming to solve the problem exactly in  $O(n^4)$  time. Currently, we are working on building an  $O(n \log n)$  time greedy algorithm to solve the problem exactly.

### 4 Future work

Immediate open problems which we are working on are the following: (a) Is there a polynomial time algorithm which gives an exact answer for unit interval graphs?, (b) Is there an  $O(n \log n)$  time algorithm to compute an exact answer for laminar interval graphs? and (c) Can an approximation algorithm better than 0.878 be obtained for (general) interval graphs? Broadly speaking, maximum cut for geometric intersection graphs looks a fertile field to combine modern techniques from approximation algorithms (such as SDPs) with geometric approaches.

---

### References

---

- 1 M.X. Goemans, D.P. Williamson. *Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming*. Journal of the ACM, 42 (1995), pp. 1115–1145.
- 2 Ranendu Adhikary, Kaustav Bose, Satwik Mukherjee, Bodhayan Roy. *Complexity of maximum cut on interval graphs*. Available as preprint arXiv:2006.00061v2, 2020.
- 3 Hans L Bodlaender, Ton Kloks, and Rolf Niedermeier. *Simple max-cut for unit interval graphs and graphs with few p4s*. Electronic Notes in Discrete Mathematics, 3:19–26, 1999.
- 4 Arman Boyacı, Tinaz Ekim, and Mordechai Shalom. *A polynomial-time algorithm for the maximum cardinality cut problem in proper interval graphs*. Information Processing Letters, 121:29–33, 2017. doi:10.1016/j.ipl.2017.01.007.
- 5 Hans L Bodlaender, Celina MH De Figueiredo, Marisa Gutierrez, Ton Kloks, and Rolf Niedermeier. *Simple max-cut for split-indifference graphs and graphs with few p4's*. In International Workshop on Experimental and Efficient Algorithms, pages 87–99, 2004.
- 6 Jan Kratochvíl, Tomáš Masařík, and Jana Novotná. *Bubble model for mixed unit interval graphs and its applications: The MaxCut problem revisited*. Available as preprint arXiv:2002.08311, 2020.

# Realizing Persistent Homology by Subcomplexes

Magnus Bakke Botnan

Vrije Universiteit Amsterdam, The Netherlands  
m.b.botnan@vu.nl

Pepijn E.R. Roos Hoefgeest

Vrije Universiteit Amsterdam, The Netherlands  
p.e.r.rooshoefgeest@vu.nl

---

## Abstract

---

We study the problem of realizing a feature in persistent homology by a subcomplex of minimal persistent homological complexity.

**2012 ACM Subject Classification** Mathematics of computing → Algebraic topology

**Keywords and phrases** Persistent Homology, Topological Data Analysis

## 1 Introduction

In topological data analysis, one is typically interested in stable invariants associated to diagrams of topological spaces arising from data. In our case, we will consider a continuous function  $f : X \rightarrow \mathbb{R}$  on a topological space  $X$ . For simplicity, we will assume that  $X = |K|$  is the geometric realization of some finite simplicial complex  $K$ , and that  $f$  is a piecewise linear function obtained from linearly interpolating the values of  $f$  on the vertices of  $K$ . The two diagrams we will be concerned with are the *sublevel set filtration* and the *level set zigzag diagram*. *Persistent homology* captures the evolution of the homology of the sublevel sets. More precisely, let  $H_p$  denote  $p$ -dimensional homology with coefficients in a fixed field  $k$ , then the persistent homology of  $(X, f)$  is defined as the functor  $H_p \mathcal{S}_f^\uparrow : \mathbf{R} \rightarrow \mathbf{Vec}_k$ , where

$$\mathcal{S}_f^\uparrow : \mathbf{R} \rightarrow \mathbf{Top} \quad \mathcal{S}_f^\uparrow(t) = \{x \in X \mid f(x) \leq t\}$$

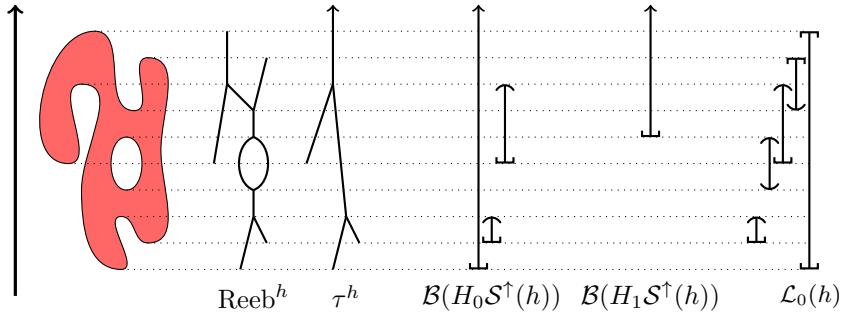
It is well-known that, for every such functor, there exists a finite collection of half-open intervals  $\mathcal{B}_f^p = \{[b_i, d_i] : b_i \leq d_i \leq \infty\}$  such that  $H_p \mathcal{S}_f^\uparrow \cong \bigoplus_{[b_i, d_i] \in \mathcal{B}_f^p} k_{[b_i, d_i]}$ , where  $k_{[b_i, d_i]}$  denotes the functor  $\mathbf{R} \rightarrow \mathbf{Vec}_k$  which is equal to  $k$  on  $[b_i, d_i]$  and which connects any two non-trivial vector spaces by the identity map. The collection  $\mathcal{B}_f^p$  is called the barcode of  $H_p \mathcal{S}_f^\uparrow$ , and completely determines  $H_p \mathcal{S}_f^\uparrow$  up to (non-unique) isomorphism. We define a *choice of basis in persistent homology* to be an isomorphism  $h_K : H_p \mathcal{S}_f^\uparrow \xrightarrow{\cong} \bigoplus_{[b_i, d_i] \in \mathcal{B}_f} k_{[b_i, d_i]}$ , and we shall refer to  $h_K^{-1}(k_{[b_i, d_i]})$  as a *basis element*. A finer invariant can be obtained by means of the  $p$ -dimensional *levelset persistence barcode*  $\mathcal{L}_f^p$  which measures the evolution of the homological features across fibers [1]; see Figure 1.

## 2 Topological Complexity of Features

The barcode  $\mathcal{B}_f^p$  provides a succinct summary of the homological evolution of the sublevel filtration of  $f$ . But what does a bar really represent? That is a problem which has received a significant amount of interest within the community. As there is no canonical choice of basis in homology, one needs to choose representatives which are optimal with respect to a

---

<sup>1</sup> This is an abstract of a presentation given at CG:YRF 2021. 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.



**Figure 1** A continuous function  $h$  together with its Reeb graph, merge tree and barcodes.

user-defined complexity measure; see e.g. [2]. For instance, finding the representative cycle of minimal length is a well-studied problem [3][6][5]. In this project we are interested in representing *persistent* homology classes by subcomplexes that are topologically as simple as possible. We measure the topological complexity by means of the persistent homology and levelset homology across subcomplexes *carrying* (see below) the feature. In the following  $L$  denotes a subcomplex of  $K$ ,  $i: L \hookrightarrow K$  the corresponding inclusion, and  $i_*$  the induced map in persistent homology  $i_*: H_p S_{f \circ i}^\dagger \rightarrow H_p S_f^\dagger$ .

► **Remark 1.** For the sake of simplicity we shall assume that no barcode has an interval appearing with multiplicity greater than 1.

► **Definition 2.**

1. *The basis element  $h_K^{-1}(k_{[b_j, d_j]})$  is carried by the subcomplex  $L$ , if  $[b_j, d_j] \in \mathcal{B}_{f \circ i}^p$  and there exists a choice of basis  $h_L$  such that  $h_K^{-1}(k_{[b_j, d_j]}) = i_*(h_L^{-1}(k_{[b_j, d_j]}))$ .*
2. *The bar  $[b_j, d_j] \in \mathcal{B}_f^p$  is carried by the subcomplex  $L$ , if  $[b_j, d_j] \in \mathcal{B}_{f \circ i}^p$ , and there exist choices of basis  $h_K$  and  $h_L$  such that  $h_K^{-1}(k_{[b_j, d_j]}) = i_*(h_L^{-1}(k_{[b_j, d_j]}))$ .*

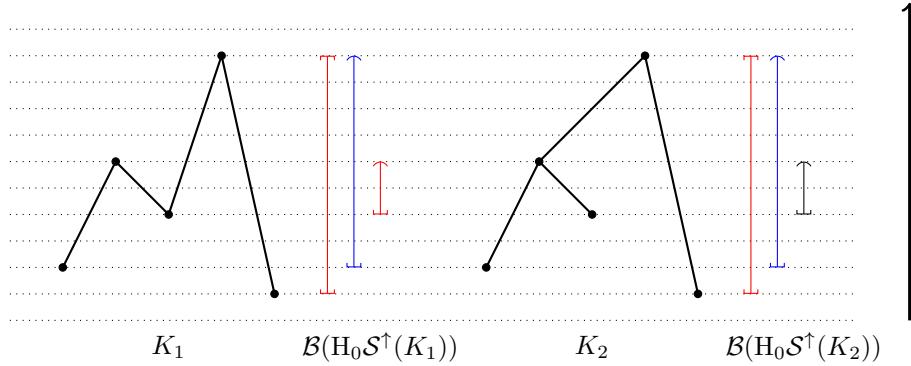
The first definition can be thought of as fixing a persistent homology class and looking for all boundaries bounding that specific class. The second definition allows for the choice of any persistent homology class, as long as it defines a bar of the prescribed length. As a means to visualizing a basis element (bar) in persistent homology, one can consider the associated Merge tree or Reeb graph of a subcomplex  $L$  carrying the basis element (bar). Such subcomplexes are decidedly non-unique and may have additional persistent homology “unrelated” to the feature of interest. One thus seeks the carrying subcomplex with the “simplest” merge tree or Reeb graph. Since meaningful geometrical properties of the two graphs can be inferred by their associated barcodes in persistent homology and level set homology in dimension 0, respectively, see Figure 1, we propose to measure ‘simplicity’ in terms of their associated barcodes in dimension 0. More generally, we introduce the following: restricting the lifetimes of the bars to the maximum value of  $f$ , the *total persistence in dimension p of f* is  $TP_p(f) = \sum_{[b_i, d_i] \in \mathcal{B}_f^p} d_i - b_i$ , and the *total level set persistence in dimension p of f* is  $TLP_p(f) = \sum_{\langle b_i, d_i \rangle \in \mathcal{L}_f^p} d_i - b_i$ .

► **Definition 3.** *The persistent homological complexity in dimension q of*

- *a basis element  $h_K^{-1}(k_{[b_j, d_j]})$ ,  $[b_j, d_j] \in \mathcal{B}_f^p$ , is given by*

$$PHC_q(h_K^{-1}(k_{[b_j, d_j]})) = \min_{i: L \hookrightarrow K} (TP_q(f \circ i)),$$

- *a bar  $[b_j, d_j] \in \mathcal{B}_f^p$  is given by  $PHC_q([b_j, d_j]) = \min_{i: L \hookrightarrow K} (TP_q(f \circ i))$ ,*



**Figure 2** Two simplicial complexes filtered by height with their identical associated barcodes.

where the minima range over all  $L$  carrying the basis element and bar, respectively.

By substituting  $TP$  by  $TLP$  in the above definition one arrives at the *level set homological complexity in dimension  $q$* ,  $LHC_q$ . Note that  $p \neq q$  in general. The example in figure 2 motivates the above definition. A minimal representative for the blue bar on the left will pick up all the other red bars in its barcode, whereas a minimal representative for the blue bar on the right will only have the one other red bar in its barcode.

Using the particular structure of persistent homology in dimension 0, together with standard shortest path algorithms, we show the following.

► **Theorem 4.** Let  $[b, d] \in \mathcal{B}_f^0$ . Then  $PHC_0([b, d])$  and  $LHC_0([b, d])$  can be computed in  $O(V^2)$  where  $V$  is the total number of vertices in  $K$ .

Somewhat surprisingly, fixing a basis element turns out to be a harder problem in dimension 0: Given an instance of the Steiner tree problem in an undirected graph  $G$ , we construct from  $G$  a filtered graph  $G'$  such that finding a minimal subcomplex carrying a particular basis element  $h_K^{-1}(k_{[b, d]})$  in 0-dimensional homology amounts to finding a minimal Steiner tree in  $G$ . This proves the following:

► **Theorem 5.** Let  $[b, d] \in \mathcal{B}_f^0$ . It is in general NP-hard to compute  $PHC_0(h_K^{-1}k_{[b, d]}))$ .

Adapting the construction of [4, Section 4], we show the following:

► **Theorem 6.** Let  $[b, d] \in \mathcal{B}_f^1$ . It is in general NP-hard to compute  $LHC_1([b, d])$ .

We expect similar results to hold for different choices of  $p$  and  $q$ , for  $PHC$  as well as  $LHC$ . Future work includes working out the corresponding results for manifolds, as well as the hardness of approximation.

---

## References

---

- 1 Gunnar Carlsson, Vin De Silva, and Dmitriy Morozov. Zigzag persistent homology and real-valued functions. In *Proceedings of the twenty-fifth annual symposium on Computational geometry*, pages 247–256, 2009.
- 2 Chao Chen and Daniel Freedman. Quantifying homology classes. *arXiv preprint arXiv:0802.2865*, 2008.
- 3 Tamal K Dey, Tao Hou, and Sayan Mandal. Computing minimal persistent cycles: Polynomial and hard cases. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2587–2606. SIAM, 2020.

- 4 Nathan M Dunfield and Anil N Hirani. The least spanning area of a knot and the optimal bounding chain problem. In *Proceedings of the twenty-seventh annual symposium on Computational geometry*, pages 135–144, 2011.
- 5 Emerson Escolar and Yasuaki Hiraoka. *Optimal Cycles for Persistent Homology Via Linear Programming*, pages 79–96. 01 2016. doi:[10.1007/978-4-431-55420-2\\_5](https://doi.org/10.1007/978-4-431-55420-2_5).
- 6 Ippei Obayashi. Volume-optimal cycle: Tightest representative cycle of a generator in persistent homology. *SIAM Journal on Applied Algebra and Geometry*, 2, 12 2017. doi:[10.1137/17M1159439](https://doi.org/10.1137/17M1159439).

# Edge-unfolding nearly flat prismatoids

Manuel Radons 

Technische Universität Berlin, Chair of Discrete Mathematics/Geometry, Germany

---

## Abstract

---

A 3-Prismatoid  $P$  is the convex hull of two convex polygons  $A$  and  $B$  which lie in parallel planes  $H_A, H_B \subset \mathbb{R}^3$ . Let  $A'$  be the orthogonal projection of  $A$  onto  $H_B$ . Building on techniques introduced by O'Rourke, we show that  $P$  can be edge-unfolded if  $A'$  is properly contained in  $B$  and  $P$  is sufficiently flat.<sup>1</sup>

**2012 ACM Subject Classification** Computational Geometry

**Keywords and phrases** Duerers problem, edge unfolding polytopes, nets

**Acknowledgements** I want to thank my advisor, Michael Joswig, for pointing me at Dürer's problem. I further want to thank Joseph O'Rourke for helpful discussions on the topic

## 1 Introduction

The question whether any 3-polytope has a net, that is, whether it is possible to cut it along some spanning tree of its edge graph so that the resulting connected surface may be unfolded flat into the plane without self-overlaps, can be dated back to the ‘Painter’s Manual’ by Albrecht Dürer [5]. It is thus often referred to as Dürer’s Problem.

A polytope that has a net is called unfoldable. It was proved by Ghomi that every polytope is unfoldable after an affine stretching, which implies that every combinatorial type of polytope has an unfoldable realization [6]. O'Rourke recently established the unfoldability of nearly flat, acutely triangulated convex caps [12, 11]. A convex cap is a polytope  $C$  which has a designated facet  $F$  so that the orthogonal projection of  $C \setminus F$  to  $F$  is one-to-one. An acute triangulation is a triangulation so that any interior angle of any triangle is smaller than  $\pi/2$ . A recent negative result, which Barvinok and Ghomi distilled from a highly original but flawed preprint of Tarasov [2, 13], concerns the existence of counterexamples to a more general form of Dürer’s problem which considers cuts along so-called pseudo-edges, which are geodesics in the intrinsic metric of a polytope. Another generalized form of Dürer’s problem concerns unfoldability of non-convex polytopes which are combinatorially equivalent to a convex 3-polytope. There are several ununfoldable families of such polytopes known, cf. [7, 14, 4].

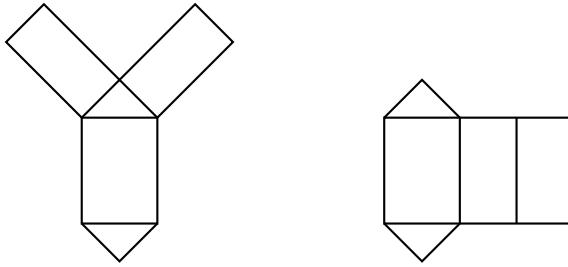
### 1.1 Unfolding Prismatoids

A prismatoid  $P$  is the convex hull of two (convex) polygons  $A$  and  $B$  that lie in parallel planes, say  $H_A$  and  $H_B$ . There are two natural ways to unfold primateoids, the *band unfolding*, and the *petal unfolding* [10], which are illustrated in Figure 1.

A band unfolding cuts one lateral edge and unrolls all lateral facets into the plane as one connected patch, while  $A$  and  $B$  are left attached to this *band* along one suitable edge each. Every prismatoid has a band that can be unfolded without self-overlap [1], but there exist prismatoids which have no band that admits a non-overlapping placement of  $A$  and  $B$  [9].

---

<sup>1</sup> This is an abstract of a presentation given at CG:YRF 2021. 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.



■ **Figure 1** Petal and band unfolding of a prism over a triangle.

In a petal unfolding either  $A$  or  $B$  is a designated facet to which all lateral facets are left attached. Assume that the designated facet is  $B$ . Then for each vertex  $v$  of  $B$  exactly one lateral edge adjacent to  $v$  is cut. The so-resulting *petals* are unfolded into the plane while  $A$  is left attached to this unit along a single suitable edge. A nonobtuse triangle is a triangle so that all its interior angles are smaller than or equal to  $\pi/2$ . It is known that a prismaticoid has a petal unfolding if all its facets, except possibly its base  $B$ , are nonobtuse triangles [10], or if it is a prismoid, that is, if all its lateral facets are trapezoids [8]. Further, so-called smooth prismaticoids, which are the convex hull of two smooth curves lying in parallel planes, can be unfolded via a similar ansatz [3].

We say a prismaticoid is in volcano constellation if the orthogonal projection of  $A$  onto  $H_B$  is properly contained in  $B$ , or vice versa. In the present work we apply a combination of the petal and the band unfolding strategies to prismaticoids in volcano constellation. That is, we leave the band of lateral facets largely intact, but not entirely. Crucial in the selection of the band-patches which are left intact is the notion of radially monotone curves, which was exploited to great effect in [12]. Just as in the latter reference, we consider nearly flat instances of the investigated polytopes.

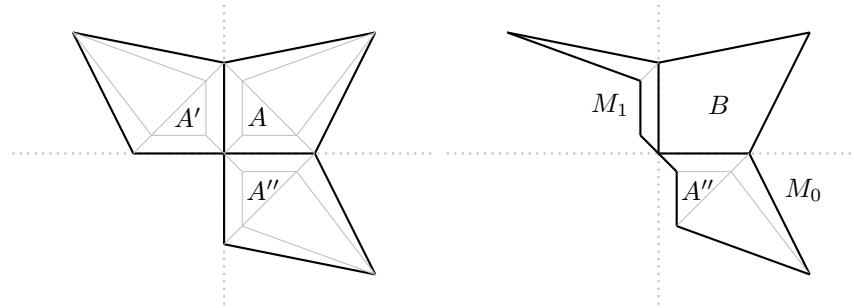
## 2 Main result

► **Theorem 1.** *Assume that  $H$  is the  $xy$ -plane embedded in  $\mathbb{R}^3$ . Let  $\alpha > 0$  and set  $H^\alpha := \mathbb{R}^2 \times \{\alpha\} \subset \mathbb{R}^3$ . Further, let  $A, B \subset H$  be two convex polygons so that  $A$  is properly contained in  $B$  or vice versa. Let  $A^\alpha$  be the orthogonal projection of  $A$  onto  $H^\alpha$ . Then there exists an  $\varepsilon > 0$  so that for all  $\alpha \in (0, \varepsilon)$  the pismatoid  $\text{conv}(A^\alpha, B)$  in volcano constellation is unfoldable.*

### 2.1 Methods of the proof

Assume that  $A = A^0$  is properly contained in  $B$ . We construct  $P^0$  as a flat 3-polytope whose lower facet is  $B$  and whose upper facets coincide with the subdivision of  $B$  induced by the orthogonal projection of  $P$  onto  $B$ . We consider two cases: In the first case  $B$  has at least one interior angle  $\leq \frac{\pi}{2}$ , say at a vertex  $v$ . In the second, it does not.

In the first case,  $P^0$  can be mirrorored twice at the edges of  $B$  incident to  $v$ , as indicated in Figure 2. Let  $\Gamma$  be a subpath of  $B$  in clockwise direction, starting at  $v$  and ending at a vertex  $v'$  which has the maximum length so that the sum of the angles spanned by the outer normal cones at the interior vertices of  $\Gamma$  is  $\leq \pi$ . Then the subpath of  $B$  starting at  $v'$  and ending at  $v$  has the same property. Cut two arbitrary edges  $e, e'$  incident to  $v$  and  $v'$ , respectively. At least one of the two so-induced subpaths of the top will also have the



■ **Figure 2** Reflections and removal of superfluous pieces

latter property. Say, this path is called  $\Gamma'$ . We call the band piece that this subpath bounds  $M_0$  and the other one  $M_1$ . Now remove the reflection of the top and the mirrored copy of  $M_0$  from one of the reflections of  $P^0$  and  $M_1$  from the other. Also, remove the top and the bands from  $P^0$  itself, and the reflections of the base from both mirrored copies.

Increasing the parameter  $\alpha$  induces a continuous deformation of the so-defined unfolding of  $P^0$ . We show that there always exists an edge of  $\Gamma'$  to which the reflection of  $A$  can be attached, so that for sufficiently small  $\alpha$  the unfolding stays one-to-one.

If all interior angles of  $B$  are  $> \frac{\pi}{2}$ , then there must exist edges  $e, e'$  of  $B$  so that the angle of the intersection of the lines through them which faces  $B$  is  $< \frac{\pi}{2}$ . Thus the argumentation devised for the first case can also be utilized in the second.

---

## References

---

- 1 Greg Aloupis. *Reconfigurations of Polygonal Structures*. PhD thesis, Montreal, Que., Canada, Canada, 2005. AAINR12794.
- 2 Nicholas Barvinok and Mohammad Ghomi. Pseudo-edge unfoldings of convex polyhedra. *Discrete and Computational Geometry*, pages 1–19, 2017.
- 3 Nadia Benbernou, Patricia Cahn, and Joseph O’Rourke. Unfolding smooth prismsatoids. *CoRR*, cs.CG/0407063, 2004. URL: <http://arxiv.org/abs/cs.CG/0407063>.
- 4 Erik D. Demaine, Martin L. Demaine, and David Eppstein. Acutely triangulated, stacked, and very ununfoldable polyhedra. In *CCCG 2020, Saskatoon, Canada, August 5–7*, 2020.
- 5 Albrecht Dürer. *The painter’s manual: A manual of measurement of lines, areas, and solids by means of compass and ruler assembled by Albrecht Dürer for the use of all lovers of art with appropriate illustrations arranged to be printed in the year MDXXV*. Abaris Books, 1977 (1525).
- 6 Mohammad Ghomi. Affine unfoldings of convex polyhedra. *Geometry and Topology*, 18:3055–3090, 2014.
- 7 Branko Grünbaum. No-net polyhedra. *Geombinatorics*, 11:111 – 114, 2002.
- 8 Joseph O’Rourke. Unfolding prismoids without overlap. *Unpublished manuscript*, 2001.
- 9 Joseph O’Rourke. Band unfoldings and prismsatoids: A counterexample. *CoRR*, abs/0710.0811, 2007. URL: <http://arxiv.org/abs/0710.0811>, arXiv:0710.0811.
- 10 Joseph O’Rourke. Unfolding prismsatoids as convex patches: Counterexamples and positive results. *CoRR*, abs/1205.2048, 2012. URL: <http://arxiv.org/abs/1205.2048>, arXiv:1205.2048.
- 11 Joseph O’Rourke. Addendum to: Edge-unfolding nearly flat convex caps. *CoRR*, abs/1709.02433, 2017. URL: <http://arxiv.org/abs/1709.02433>, arXiv:1709.02433.
- 12 Joseph O’Rourke. Edge-Unfolding Nearly Flat Convex Caps. In Bettina Speckmann and Csaba D. Tóth, editors, *34th International Symposium on Computational Geometry (SoCG)*

- 2018), volume 99 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 64:1–64:14, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. URL: <http://drops.dagstuhl.de/opus/volltexte/2018/8777>, doi:10.4230/LIPIcs.SoCG.2018.64.
- 13 Alexey S. Tarasov. Existence of a polyhedron which does not have a non-overlapping pseudo-edge unfolding. *ArXiv*, abs/0806.2360, 2008.
- 14 A.S. Tarasov. Polyhedra that do not admitnatural unfoldings. *Uspekhi Matematicheskikh.*

# Using Generalized Heegaard Splittings in Computational 3-Manifold Topology\*

Kristóf Huszár 

Inria Sophia Antipolis - Méditerranée, Sophia Antipolis, France

---

## Abstract

---

Introduced by Scharlemann and Thompson in 1992, generalized Heegaard splittings provide a powerful framework for studying 3-dimensional manifolds. Recently they have also found applications in a computational setting. Extending this line of research, here we outline how they can be used (in combination with other topological tools) to upper-bound the pathwidth of a hyperbolic 3-manifold in terms of its volume, improving upon earlier work by Maria and Purcell. To this end, we explain what generalized Heegaard splittings are, and also discuss the algorithmic implications of our result.

**2012 ACM Subject Classification** Mathematics of computing → Geometric topology; Theory of computation → Fixed parameter tractability

**Keywords and phrases** computational 3-manifold topology, fixed-parameter tractability, generalized Heegaard splittings, hyperbolic 3-manifolds, pathwidth, treewidth, triangulations, volume

**Related Version** This work is based on previously unpublished parts of the author's PhD thesis available at <https://doi.org/10.15479/AT:ISTA:8032>.

**Funding** Kristóf Huszár: This work has been supported by the French government through the 3IA Côte d'Azur Investments in the Future project managed by the National Research Agency (ANR) with the reference number ANR-19-P3IA-0002.

**Acknowledgements** I am grateful to Uli Wagner and Jonathan Spreer for their guidance and steady support during my PhD. I thank Clément Maria for many stimulating discussions, and Pascal Wild and the anonymous reviewers for proofreading this abstract.

## 1 Context and Motivation

Algorithms in computational 3-manifold topology typically take a triangulation as input and return topological information about the underlying manifold. The difficulty of extracting the desired information, however, might greatly depend on the choice of the input triangulation. In recent years, several computationally hard problems about triangulated 3-manifolds were shown to admit algorithmic solutions that are *fixed-parameter tractable* (FPT) in the *treewidth*<sup>1</sup> of the dual graph of the input triangulation [3, 4, 5, 6, 7]. These algorithms still require exponential time to terminate in the worst case. However, for triangulations with dual graph of bounded treewidth they run in polynomial time.

In the light of these algorithms, it is compelling to consider the *treewidth*  $\text{tw}(\mathcal{M})$  of a compact 3-manifold  $\mathcal{M}$ , defined as the smallest treewidth of the dual graph of any triangulation thereof. Over the last few years, the quantitative relationship between the treewidth and other properties of 3-manifolds has been studied in various settings. Together with Spreer and Wagner we showed that, for certain families of 3-manifolds the *Heegaard genus* gives a lower bound on the treewidth and it can be arbitrary large [12]. The proof of this result rests on the theory of *generalized Heegaard splittings*, a structure we now briefly review.

---

\* This is an abstract of a presentation given at CG:YRF 2021. 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.

<sup>1</sup> The treewidth is a structural graph parameter measuring the “tree-likeness” of a graph. See, e.g., [1].

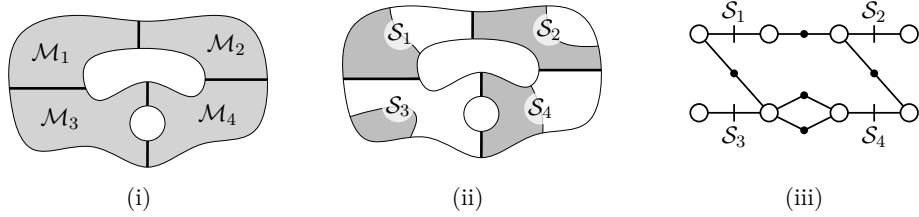
## 2 Generalized Heegaard Splittings

We begin with classical Heegaard splittings, that have been central to the study of 3-manifolds for over a century [8, 16]. Given a closed, orientable 3-manifold  $\mathcal{M}$ , a *Heegaard splitting* is a decomposition of  $\mathcal{M}$  into a union of two homeomorphic *handlebodies*<sup>2</sup> glued together along their boundary surface called the *splitting surface*. The *Heegaard genus*  $g(\mathcal{M})$ —one of the best known invariants for 3-manifolds—is the smallest genus of any splitting surface of  $\mathcal{M}$ .

This notion was refined in [18] (cf. [17]). In a *generalized Heegaard splitting* a 3-manifold is presented as a union of several pairs of *compression bodies*, building blocks that generalize handlebodies and allow one to capture the structure of 3-manifolds in more detail. To form a generalized Heegaard splitting of a compact 3-manifold  $\mathcal{M}$ , start with a decomposition

$$\mathcal{D} = \{\mathcal{M}_i : i \in I, \bigcup_{i \in I} \mathcal{M}_i = \mathcal{M}, \text{ and } \text{int}(\mathcal{M}_i) \cap \text{int}(\mathcal{M}_j) = \emptyset \text{ for } i \neq j\} \quad (1)$$

into finitely many connected 3-dimensional submanifolds with pairwise disjoint interiors, intersecting along closed surfaces, and take an appropriate Heegaard splitting<sup>3</sup> for each  $\mathcal{M}_i$ . Generalized Heegaard splittings can be manipulated in several ways, and optimal (i.e., *thin*) splittings possess desirable properties, which make them particularly useful (cf. [9, 17]).



■ **Figure 1** (i)-(ii) Schematics of a decomposition  $\mathcal{D}$  of  $\mathcal{M}$  into four submanifolds, and of a generalized Heegaard splitting arising from  $\mathcal{D}$ . (iii) The *fork complex* [17, Section 5.1] describing this splitting.

## 3 The Main Result

Maria and Purcell have recently shown that, in the realm of *hyperbolic* 3-manifolds another important invariant, the *volume*, yields an upper bound on the *treewidth* [15]. They proved the existence of a universal constant  $C > 0$ , such that, for every closed, orientable, hyperbolic 3-manifold  $\mathcal{M}$  with treewidth  $\text{tw}(\mathcal{M})$  and volume  $\text{vol}(\mathcal{M})$  the following inequality holds:

$$\text{tw}(\mathcal{M}) \leq C \cdot \text{vol}(\mathcal{M}). \quad (2)$$

Here we show that the volume provides a linear upper bound even on the *pathwidth* of  $\mathcal{M}$ —a quantity closely related to, but potentially much larger than the treewidth. We prove

► **Theorem 1** (Theorem 1.8 in [10]). *There exists a universal constant  $C' > 0$  such that, for any closed, orientable, hyperbolic 3-manifold  $\mathcal{M}$  with pathwidth  $\text{pw}(\mathcal{M})$  and volume  $\text{vol}(\mathcal{M})$ ,*

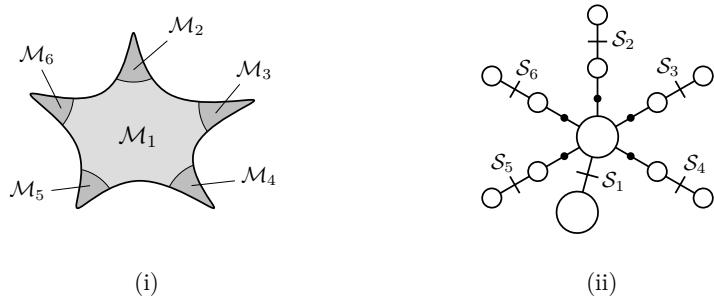
$$\text{pw}(\mathcal{M}) \leq C' \cdot \text{vol}(\mathcal{M}). \quad (3)$$

<sup>2</sup> Handlebodies are connected 3-manifolds with boundary, that can be thought of as thickened graphs.

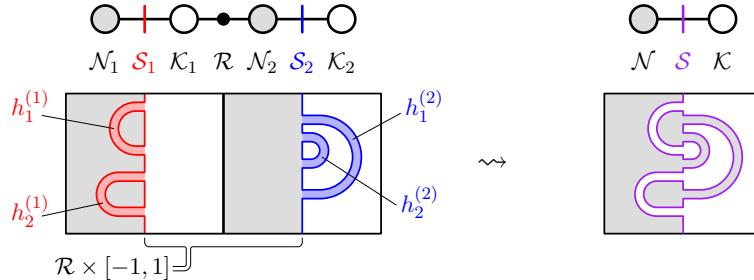
<sup>3</sup> Here ‘appropriate’ refers to a natural condition the splitting surfaces have to satisfy [10, Section 4.1.2]. Also, we might need to take Heegaard splittings of 3-manifolds with boundary, cf. [17, Theorem 2.1.11].

**Outline of the proof.** Our roadmap to establish Theorem 1 is similar to that in [15]. In particular, our construction of a triangulation of  $\mathcal{M}$  with dual graph of pathwidth bounded in terms of  $\text{vol}(\mathcal{M})$  also starts with a *thick-thin decomposition*  $\mathcal{D}$  of  $\mathcal{M}$ . The two proofs, however, diverge at this point. Maria and Purcell proceed by triangulating the thick part of  $\mathcal{D}$  using the work of Jørgensen–Thurston [20, §5.11] and Kobayashi–Rieck [14]. This partial triangulation is then simplified [2, 13] and completed into the desired triangulation of  $\mathcal{M}$ .

The novelty in our work is, that we proceed by first turning the decomposition  $\mathcal{D}$  into a generalized Heegaard splitting of  $\mathcal{M}$ , where we rely on the aforementioned results to control the genera of the splitting surfaces. We then *amalgamate* [19] this generalized Heegaard splitting into a classical one of genus  $O(\text{vol}(\mathcal{M}))$ . Finally, using our earlier work [11], we turn this Heegaard splitting into a triangulation of  $\mathcal{M}$  with dual graph of pathwidth  $O(\text{vol}(\mathcal{M}))$ .



■ **Figure 2** (i) Schematic example of a thick-thin decomposition  $\mathcal{D}$  of a hyperbolic 3-manifold  $\mathcal{M}$ .  
(ii) The fork complex of a generalized Heegaard splitting associated with  $\mathcal{D}$ .



■ **Figure 3** Example of amalgamating a generalized Heegaard splitting into a Heegaard splitting.

**Algorithmic implications.** The proof of Theorem 1 gives a template for an algorithm<sup>4</sup> to triangulate any closed hyperbolic 3-manifold  $\mathcal{M}$  in such a way, that the dual graph of the resulting triangulation has pathwidth  $O(\text{vol}(\mathcal{M}))$ . Using such triangulations—that have a dual graph not only of small treewidth, but also pathwidth—as input for FPT-algorithms may significantly reduce their running time. This is because such triangulations lend themselves to *nice tree decompositions* (the data structure underlying many algorithms FPT in the treewidth) without *join bags* (those parts of a nice tree decomposition that often account for the computational bottleneck, cf. [4]). By Theorem 1, in case of hyperbolic 3-manifolds with bounded volume working with such triangulations is (in theory) always possible.

<sup>4</sup> We refer to the discussion in [15, Section 5.1] for the description of a possible computational model.

---

**References**

---

- 1 H. L. Bodlaender. A tourist guide through treewidth. *Acta Cybern.*, 11(1-2):1–21, 1993. <urn:nbn:nl:ui:10-1874-2301>.
- 2 B. A. Burton. A new approach to crushing 3-manifold triangulations. *Discrete Comput. Geom.*, 52(1):116–139, 2014. <doi:10.1007/s00454-014-9572-y>.
- 3 B. A. Burton and R. G. Downey. Courcelle’s theorem for triangulations. *J. Comb. Theory, Ser. A*, 146:264–294, 2017. <doi:10.1016/j.jcta.2016.10.001>.
- 4 B. A. Burton, T. Lewiner, J. Paixão, and J. Spreer. Parameterized complexity of discrete Morse theory. *ACM Trans. Math. Softw.*, 42(1):6:1–6:24, 2016. <doi:10.1145/2738034>.
- 5 B. A. Burton, C. Maria, and J. Spreer. Algorithms and complexity for Turaev–Viro invariants. *J. Appl. Comput. Topol.*, 2(1–2):33–53, 2018. <doi:10.1007/s41468-018-0016-2>.
- 6 B. A. Burton and W. Pettersson. Fixed parameter tractable algorithms in combinatorial topology. In *Proc. 20th Int. Conf. Comput. Comb. (COCOON 2014)*, pages 300–311, 2014. [doi:10.1007/978-3-319-08783-2\\_26](doi:10.1007/978-3-319-08783-2_26).
- 7 B. A. Burton and J. Spreer. The complexity of detecting taut angle structures on triangulations. In *Proc. 24th Annu. ACM-SIAM Symp. Discrete Algorithms (SODA 2013)*, pages 168–183, 2013. <doi:10.1137/1.9781611973105.13>.
- 8 P. Heegaard. Sur l’"Analysis situs". *Bull. Soc. Math. France*, 44:161–242, 1916. <doi:10.24033-bsmf.968>.
- 9 H. Howards, Y. Rieck, and J. Schultens. Thin position for knots and 3-manifolds: a unified approach. In *Workshop on Heegaard Splittings*, volume 12 of *Geom. Topol. Monogr.*, pages 89–120. Geom. Topol. Publ., Coventry, 2007. <doi:10.2140/gtm.2007.12.89>.
- 10 K. Huszár. *Combinatorial width parameters for 3-dimensional manifolds*. PhD thesis, IST Austria, June 2020. <doi:10.15479/AT:ISTA:8032>.
- 11 K. Huszár and J. Spreer. 3-Manifold triangulations with small treewidth. In *35th Int. Symp. Comput. Geom. (SoCG 2019)*, volume 129 of *LIPICS. Leibniz Int. Proc. Inf.*, pages 44:1–44:20. Schloss Dagstuhl–Leibniz-Zent. Inf., 2019. <doi:10.4230/LIPIcs.SoCG.2019.44>.
- 12 K. Huszár, J. Spreer, and U. Wagner. On the treewidth of triangulated 3-manifolds. *J. Comput. Geom.*, 10(2):70–98, 2019. URL: <https://jocg.org/index.php/jocg/article/view/3088>.
- 13 W. Jaco and J. H. Rubinstein. 0-efficient triangulations of 3-manifolds. *J. Differential Geom.*, 65(1):61–168, 2003.
- 14 T. Kobayashi and Y. Rieck. A linear bound on the tetrahedral number of manifolds of bounded volume (after Jørgensen and Thurston). In *Topology and Geometry in Dimension Three*, volume 560 of *Contemp. Math.*, pages 27–42. Amer. Math. Soc., Providence, RI, 2011. <doi:10.1090/conm/560/11089>.
- 15 C. Maria and J. Purcell. Treewidth, crushing and hyperbolic volume. *Algebr. Geom. Topol.*, 19(5):2625–2652, 2019. <doi:10.2140/agt.2019.19.2625>.
- 16 M. Scharlemann. Heegaard splittings of compact 3-manifolds. In *Handbook of Geometric Topology*, pages 921–953. North-Holland, Amsterdam, 2001. <doi:10.1016/B978-044482432-5/50019-6>.
- 17 M. Scharlemann, J. Schultens, and T. Saito. *Lecture Notes on Generalized Heegaard Splittings*. World Scientific Publishing Co. Pte. Ltd., Hackensack, NJ, 2016. <doi:10.1142/10019>.
- 18 M. Scharlemann and A. Thompson. Thin position for 3-manifolds. In *Geometric topology (Haifa, 1992)*, volume 164 of *Contemp. Math.*, pages 231–238. Amer. Math. Soc., Providence, RI, 1994. <doi:10.1090/conm/164/01596>.
- 19 J. Schultens. The classification of Heegaard splittings for (compact orientable surface)  $\times S^1$ . *Proc. London Math. Soc. (3)*, 67(2):425–448, 1993. <doi:10.1112/plms/s3-67.2.425>.
- 20 W. P. Thurston. The geometry and topology of three-manifolds. Electronic version 1.1, March, 2002. URL: <http://library.msri.org/books/gt3m>.

# Untangling Almost Outerplanar Drawings\*

Sujoy Bhore  

Université libre de Bruxelles, Belgium

Guangping Li  

Algorithms and Complexity Group, TU Wien, Vienna, Austria

Martin Nöllenburg  

Algorithms and Complexity Group, TU Wien, Vienna, Austria

Ignaz Rutter  

University of Passau, Passau, Germany

Hsiang-Yun Wu  

Research Unit of Computer Graphics, TU Wien, Vienna, Austria

---

## Abstract

Given an  $n$ -vertex outerplanar graph  $G$ , let  $\delta_G$  be a straight-line drawing of  $G$ , where the vertices lie on a circle and all crossings involve a single edge. We call such a drawing an *almost outerplanar* drawing. An outerplanar drawing of  $G$  can be obtained from  $\delta_G$  by *untangling* it, i.e., moving the vertices on the circle in  $\delta_G$ . Let  $\text{fix}^\circ(\delta_G)$  denote the maximum number of vertices that can remain fixed to untangle  $\delta_G$ . We show  $\text{fix}^\circ(\delta_G) \geq \lceil (n+2)/2 \rceil$  and this bound is asymptotically tight.

**2012 ACM Subject Classification** Theory of computation → Computational geometry

**Keywords and phrases** Graph drawing, straight-line drawing, planarity, moving vertices, untangling

**Acknowledgements** Research supported by the Austrian Science Fund (FWF), grant P 31119

## 1 Introduction

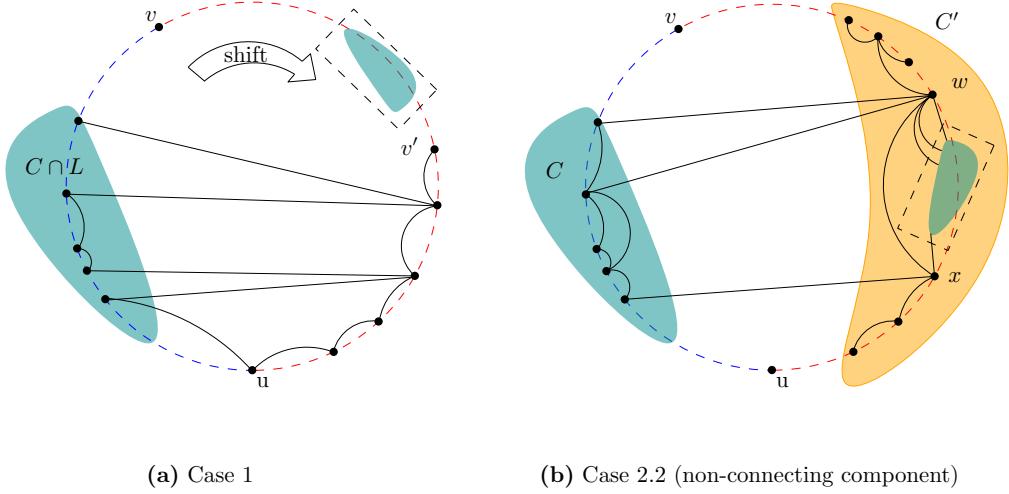
A graph is an *outerplanar* graph if it has a planar drawing in which all vertices are on the boundary of a single face, and such a drawing is known as an outerplanar drawing. Given an  $n$ -vertex outerplanar graph  $G$ , let  $\delta_G$  be a straight-line drawing of  $G$ , where the vertices lie on a circle and all crossings involve a single edge. We call  $\delta_G$  an *almost outerplanar* drawing. Since  $G$  is outerplanar, an outerplanar drawing can be obtained from  $\delta_G$  by moving the vertices on the circle. We call such a sequence of vertex moving operations an *untangling* of  $\delta_G$ . We define the *outerplanar fixing number*  $\text{fix}^\circ(\delta_G)$  of an almost outerplanar drawing  $\delta_G$  to be the maximum number of vertices that can remain fixed in an untangling of  $\delta_G$ . The notion of untangling is often used in the literature for a crossing elimination procedure that makes a non-planar drawing of a planar graph crossing-free; see [1–8]. Here, we follow an untangling procedure to obtain an outerplanar drawing from an almost outerplanar drawing.

## 2 Lower Bound for $\text{fix}^\circ(\delta_G)$

In the following let  $G = (V, E)$  be an outerplanar graph, let  $\delta_G$  be an almost outerplanar drawing of  $G$ , let  $e = uv \in E$  be the edge that contains all the crossings in  $\delta_G$ , and let  $G' = G - e$  and  $\delta_{G'} = \delta_G - e$ . The edge  $e$  partitions the vertices in  $V \setminus \{u, v\}$  into the sets  $L$  and  $R$  that lie left and right of the edge  $uv$  (in the direction from  $u$  to  $v$ ). We claim that it is possible to move vertices of  $L$  to the right side without modifying the order of  $R \cup \{u, v\}$  to

---

\* This is an abstract of a presentation given at CG:YRF 2021. 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.



**Figure 1** Moving a left component, keeping/reversing the clockwise ordering of its vertices.

obtain an outerplanar drawing. By symmetry, it is also possible to just move vertices of  $R$  to the left side. The claimed bound then follows from the fact that  $\min\{|L|, |R|\} \leq \lfloor (n-2)/2 \rfloor$ . We distinguish cases based on the connectivity of  $u$  and  $v$  in  $G'$ .

**Case 1:**  $u, v$  are not connected. Consider a connected component  $C$  of  $G'$  that contains vertices from  $L$  and from  $R$ . In this case,  $C$  contains at most one of  $u, v$ . W.l.o.g., assume  $v \notin C$ ; see Figure 1a. Let  $v'$  be the first clockwise vertex after  $v$  that lies in  $C$ . Let  $\delta'_G$  be the drawing obtained from  $\delta_G$  by moving the vertices of  $C \cap L$  clockwise just before  $v'$  without changing their clockwise ordering. Observe that this removes all crossings of  $e$  with  $C$ .

**Case 2:**  $u, v$  are connected. Now assume there exists a connected component in  $G'$  that contains both  $u$  and  $v$ . Note that if  $C'$  is a different connected component of  $G'$ , then it must lie entirely to the left or entirely to the right of  $e$ . We ignore such components as they never need to be moved. We hence assume that  $G'$  is connected.

**Case 2.1:**  $u, v$  are 2-connected. Due to the outerplanarity of  $G$ ,  $\delta_G$  is already planar.

**Case 2.2:**  $u, v$  are connected but not 2-connected.  $G'$  contains at least one cutvertex that separates  $u$  and  $v$ . Notice here, each path from  $u$  to  $v$  visits all these cutvertices between  $u$  and  $v$  in the same order. Let  $f$  and  $l$  be the first and the last cutvertex on any  $uv$ -path, respectively. Additionally, add  $u$  to the set of  $L, R$  that contains  $f$  and likewise add  $v$  to the set of  $L, R$  that contains  $l$ . Let  $X$  denote the set of edges of  $G'$  that have one endpoint in  $L$  and the other in  $R$ . Each connected component of  $G' - X$  is either a subset of  $L$  or a subset of  $R$ . we call these *left* and *right components*, respectively. We call a component of  $G' - X$  *connecting* if it either contains  $u$  or  $v$ , or removing it from  $G'$  disconnects  $u$  and  $v$ . For a left component  $C_L$  and a right component  $C_R$ , we denote by  $E(C_L, C_R)$  the edges of  $G'$  that connect a vertex from  $C_L$  to a vertex in  $C_R$ .

► **Lemma 1.** Every non-connecting component  $C$  is adjacent to exactly one component  $C'$  of  $G' - X$ . Moreover,  $C'$  is connecting, there are at most two vertices in  $C'$  that are incident to edges in  $E(C, C')$ , and if there are two such vertices  $w, x \in C'$ , then they are adjacent and removing  $wx$  disconnects  $C'$ .

► **Theorem 2.** Let  $C$  be a left (right) non-connecting component. It is always possible to obtain a new almost outerplanar drawing  $\delta'_G$  of  $G$  from  $\delta_G$  by moving only the vertices of  $C \setminus \{u, v\}$  to the right (left) side.

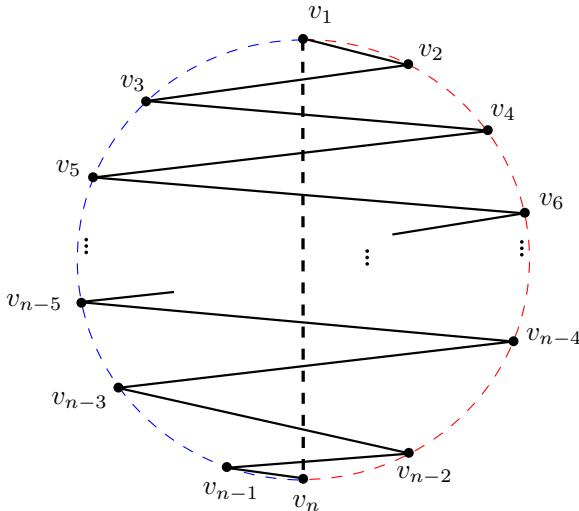
**Proof.** If  $C$  is non-connecting, then by Lemma 1, it is adjacent to at most two vertices in  $C'$  that are adjacent to  $C$ . If there are two such vertices, denote them by  $w$  and  $x$ . Note that  $w$  and  $x$  are consecutive in the drawing  $\delta_G$ , since  $G'$  is connected and  $wx$  is a bridge by Lemma 1. Otherwise let  $w$  be the only such vertex and let  $x$  be a vertex on the right side that immediately precedes or succeeds  $w$ ; see Figure 1b. We obtain  $\delta'_G$  by moving all vertices of  $C \setminus \{u, v\}$  between  $x$  and  $w$ , reversing their clockwise ordering. Observe that the choice of  $w$  and  $x$  guarantees that  $\delta'_G$  is almost outerplanar and all crossings lie on  $uv$ .  $\blacktriangleleft$

► **Lemma 3.** *The connecting component containing  $u$  or  $v$  is adjacent to at most one connecting component. Every other connecting component is adjacent to exactly two connecting components. Moreover, if  $C$  and  $C'$  are two adjacent connecting components, then there is a vertex  $w$  that is shared by all edges in  $E(C, C')$ .*

► **Theorem 4.** *Let  $C$  be a left (right) connecting component. It is always possible to obtain a new almost outerplanar drawing  $\delta'_G$  of  $G$  from  $\delta_G$  by moving only the vertices of  $C \setminus \{u, v\}$  to the right (left) side.*

### 3 The Lower Bound is Tight

Let  $n \geq 4$  be an even number and let  $G$  be the cycle on vertices  $v_1, \dots, v_n, v_1$  (in this order) and let  $\delta_G$  be a drawing with the clockwise order  $v_2, \dots, v_{2i}, \dots, v_n, v_{n-1}, \dots, v_{2i+1}, \dots, v_1$ ; see Figure 2. Clearly, the clockwise circular ordering of its vertices in a crossing-free circle drawing is either  $v_1, v_2, \dots, v_n$  or its reversal. Assume that we turn it to the clockwise ordering  $v_1, v_2, \dots, v_n$ ; the other case is symmetric. In  $\delta_G$ , the  $\frac{n}{2}$  odd-index vertices  $v_1, \dots, v_{2i+1}, \dots, v_{n-1}$  and  $v_n$  are ordered counterclockwise. To reach a clockwise ordering, at most two of these vertices can be fixed. Thus, at most  $n/2 + 1$  vertices in total can be fixed.



■ **Figure 2** The drawing  $\delta_G$  of the graph  $G$  defined in Section 3. It shows that  $\text{fix}^\circ(\delta_G) \leq \frac{n+2}{2}$ .

**Open problems for future work.** (i) The complexity of computing the outerplanar fixing number. (ii) Generalization of our result to non-outerplanar drawings of outerplanar graphs.

---

**References**

---

- 1 Prosenjit Bose, Vida Dujmovic, Ferran Hurtado, Stefan Langerman, Pat Morin, and David R. Wood. A polynomial bound for untangling geometric planar graphs. *Discret. Comput. Geom.*, 42(4):570–585, 2009. doi:[10.1007/s00454-008-9125-3](https://doi.org/10.1007/s00454-008-9125-3).
- 2 Javier Cano, Csaba D. Tóth, and Jorge Urrutia. Upper bound constructions for untangling planar geometric graphs. In Marc J. van Kreveld and Bettina Speckmann, editors, *Graph Drawing (GD'11)*, volume 7034 of *LNCS*, pages 290–295. Springer, 2011. doi:[10.1007/978-3-642-25878-7\\_28](https://doi.org/10.1007/978-3-642-25878-7_28).
- 3 Josef Cibulka. Untangling polygons and graphs. *Discret. Comput. Geom.*, 43(2):402–411, 2010. doi:[10.1007/s00454-009-9150-x](https://doi.org/10.1007/s00454-009-9150-x).
- 4 Xavier Goaoc, Jan Kratochvíl, Yoshio Okamoto, Chan-Su Shin, Andreas Spillner, and Alexander Wolff. Untangling a planar graph. *Discret. Comput. Geom.*, 42(4):542–569, 2009. doi:[10.1007/s00454-008-9130-6](https://doi.org/10.1007/s00454-008-9130-6).
- 5 Mihyun Kang, Oleg Pikhurko, Alexander Ravsky, Mathias Schacht, and Oleg Verbitsky. Untangling planar graphs from a specified vertex position - hard cases. *Discret. Appl. Math.*, 159(8):789–799, 2011. doi:[10.1016/j.dam.2011.01.011](https://doi.org/10.1016/j.dam.2011.01.011).
- 6 János Pach and Gábor Tardos. Untangling a polygon. *Discret. Comput. Geom.*, 28(4):585–592, 2002. doi:[10.1007/s00454-002-2889-y](https://doi.org/10.1007/s00454-002-2889-y).
- 7 Alexander Ravsky and Oleg Verbitsky. On collinear sets in straight-line drawings. In Petr Kolman and Jan Kratochvíl, editors, *Graph-Theoretic Concepts in Computer Science (WG'11)*, volume 6986 of *LNCS*, pages 295–306. Springer, 2011. doi:[10.1007/978-3-642-25870-1\\_27](https://doi.org/10.1007/978-3-642-25870-1_27).
- 8 Oleg Verbitsky. On the obfuscation complexity of planar graphs. *Theor. Comput. Sci.*, 396(1-3):294–300, 2008. doi:[10.1016/j.tcs.2008.02.032](https://doi.org/10.1016/j.tcs.2008.02.032).

# A Geometric Approach to Papillae Identification in 3D Meshes

Rayna Andreeva 

University of Edinburgh, UK

Anwesha Sarkar 

University of Leeds, UK

Rik Sarkar 

University of Edinburgh, UK

---

## Abstract

---

Papillae on the surface of the tongue are integral to food oral processing, sensing and transport. Fungiform papillae contain taste buds responsible for taste perception, whereas filiform papillae are crucial for friction and textural perception. We propose a novel computational pipeline to detect papillae positions and classify fungiform and filiform papillae from a 3D mesh representation of a tongue surface. The approach uses discrete differential geometry and intrinsic curvature profiles of papillae, combined with machine learning techniques. The main goal of this work is to remove the need for manual papillae identification and positioning, which are expensive and time-consuming.

**2012 ACM Subject Classification** Computing methodologies → Shape analysis

**Keywords and phrases** Papillae classification, 3D papillae detection, Curvature analysis

**Funding** *Rayna Andreeva:* Research supported by the United Kingdom Research and Innovation UKRI (grant EP/S02431X/1)

*Anwesha Sarkar:* Research supported by the Horizon 2020 European Research Council (grant 757993)

## 1 Introduction

The surface of the tongue contains numerous structures, called papillae. Figure 1 is a representation of the rough human tongue surface with various papillae structures on it. Fungiform and filiform papillae take the main role in food perception. The former are essential to food processing and sensing, while the latter create friction between food particles and the tongue.

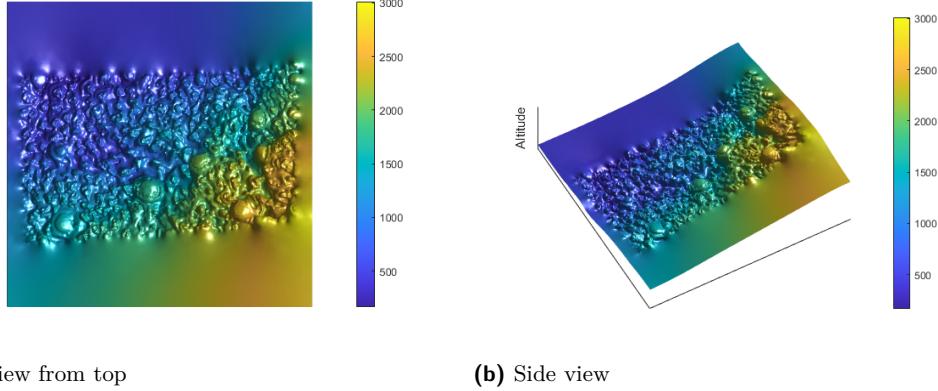
Their characteristic shapes can be used for automatic detection. In Figure 2, we can visually compare the two shapes - fungiform are generally described as round, mushroom-shaped and elevated structures [8, 9, 10], while filiform appear cylinder shaped with some spikes on top giving them a crown-like appearance.

Understanding the distribution of papillae can help us better understand oral health and food processing. The main goal of this work is to remove the need for manual papillae identification and positioning, which are expensive and time-consuming. The majority of work done in the automated papillae detection has been based on 2D digital images. However, due to the 3D nature of these structures, that information is lost in 2D representations.

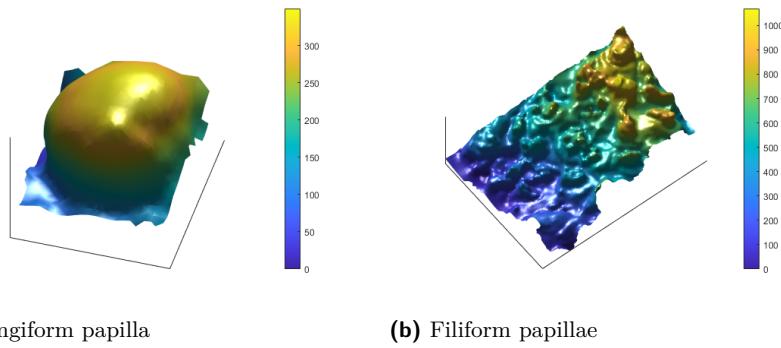
---

This is an abstract of a presentation given at CG:YRF 2021. 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.

## A Geometric Approach to Papillae Identification in 3D Meshes



**Figure 1** 3D mesh of tongue surface obtained from masks taken on real human tongue. The color bar shows the z-coordinate of the points on the surface and it represents the altitude.



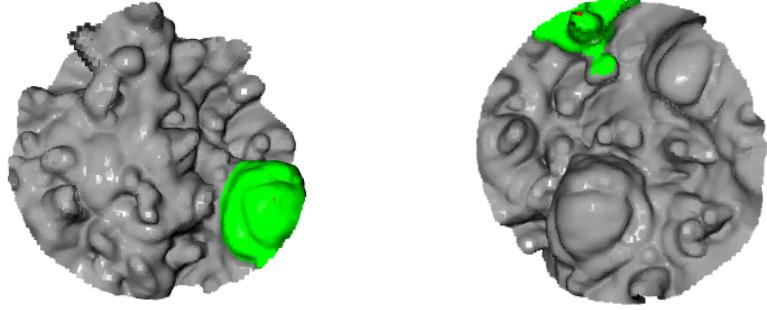
**Figure 2** Regions of the tongue with (a) Single Fungiform papilla and (b) Multiple filiform papillae. The color bar shows the z-coordinate of the points on the surface.

## 2 Algorithms

The pipeline for papillae identification works as follows. The data in the study comes from 3D scans of masks of real human tongues [1]. In the first step of the pipeline, the Screened Poisson reconstruction is used to generate the surface mesh [6]. We use Meshlab's [3] implementation of the algorithm. Next, we find segments of the mesh that are candidates for papillae. This is done using curvature and other local geometric information. In the final step of the pipeline, curvature profiles and other features are fed to a machine learning algorithm to classify filiform and fungiform papillae. Curvature computation is to be used in multiple stages of the pipeline. We thus discuss here two different approaches to curvature computation.

**Algorithm 1: Smooth Curvature.** A polynomial approximation of the discrete surface can be used to compute two curvature measures – the mean ( $H$ ) and Gaussian ( $K$ ) curvature [5, 4]. The signs of the mean and the Gaussian curvature provide information about the local behavior of the surface [4]. By performing  $HK$  classification, introduced in [2], we can describe the local shape characteristics.

**Algorithm 2: Discrete curvature.** Discrete curvature is a common tool in shape modeling, and can enable the application of many existing surface characterisation and registration techniques. However, computation of discrete curvatures is sensitive to sampling. At a high



(a) Fungiform papilla

(b) Filiform papillae

**Figure 3** Resulting cut pieces with (a) Single Fungiform papilla and (b) Single filiform papilla in green.

sampling density, such as one produced by the Poisson reconstruction, local neighborhoods of vertices are smaller, and the mesh appears almost flat in a neighborhood. We find that this effect leads to situation where the very small positive or negative curvature may produce a floating point error and snap it to zero. This effect leads to misidentification of the papillae curvature signatures.

As seen in Figure 4, typical signatures of the papillae are distribution of positive and negative curvatures. This signature can be destroyed if the curvatures snap to zero for large number of points. Further, the surface reconstruction generates a non-uniform density of vertices, resulting in irregularities of the curvature measure.

We resolve these issues by a uniform density sampling of the vertices followed by remeshing. The downsampling is done by selecting a Delone point set: In every ball of radius  $r$ , there is at least one selected vertex; and any two selections are separated by at distance at least  $r$ . The remeshing is performed by creating a Voronoi  $V$  diagram of the samples  $H$ , and deriving its dual Delaunay triangulation within the metric of the mesh  $M$ . Then, for  $x \in H$ , we compute the discrete curvature as defined by Meyer et al. [7], via the vertex's angular deficit:  $k_H(v_i) = 2\pi - \sum_{j \in N(i)} \theta_{ij}$ , where  $N(i)$  are the triangles incident on vertex  $i$  and  $\theta_{ij}$  is the angle at vertex  $i$  in triangle  $j$ .

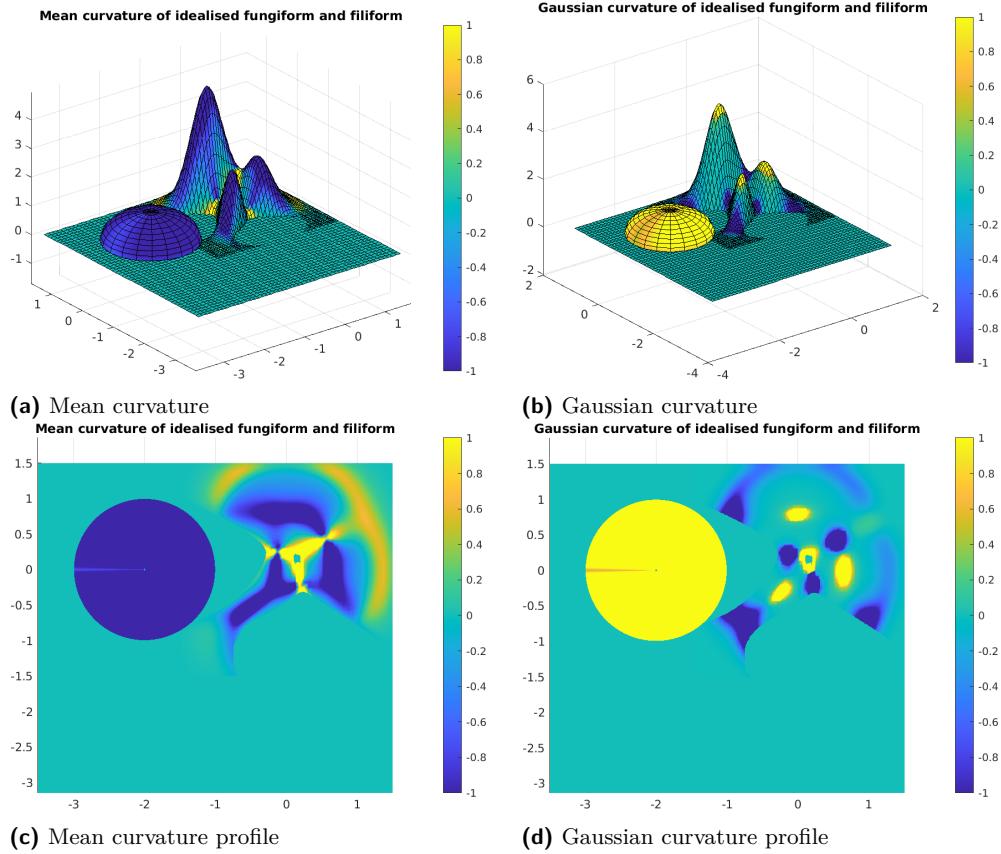
## 2.1 Candidates for papillae locations

Given the mesh, we identify candidate locations for papillae. Papillae contain local maxima and are within a certain range of width and height [1]. We thus base our identification of initial candidate locations on two features – positive curvature created by the peaks and the height of the peaks relative to surroundings, measured from a plane fitted to the neighborhood. An illustration of the latter is presented in Figure 3, where the approach successfully detects a fungiform papilla in Figure 3a and a filiform papilla in Figure 3b, coloured in green.

## 2.2 Papillae Classification

The papillae classification in our approach will be performed by identifying geometric features, and then applying machine learning to segments identified as candidates. As ground truth we will use a set of segments that have manually been labelled as *Filiform*, *Fungiform*, or

## A Geometric Approach to Papillae Identification in 3D Meshes



**Figure 4** A synthetic surface. Fungiform are dome like shapes. Filiform are sharp peaks. The color bar shows the curvature. Comparison of mean (a) vs. Gaussian curvature (b) for idealised fungiform and filiform papillae, and their corresponding profiles in (c) and (d). Peaks and the dome have maximum positive, and there are saddle points of negative curvature. Fungiform have constant positive Gaussian curvature and negative mean curvature, while filiform have positive Gaussian curvature at their peaks, and are surrounded by negative Gaussian curvature at their base.

*None.*

**Curvature profiles.** There is a substantially different curvature profile between the fungiform and the filiform papillae, and can be used for classification. The fungiform have a large region of positively curved area on top, while this is not the case for filiform as they are sharper. We demonstrate curvature profiles with synthetic surfaces in Figure 4. However, real world data is noisy and the curvature profiles will not be so well defined.

**Machine learning.** The training labels will be supplied by human experts on a dataset of several hundred segments. We would use numerical features such as height, radius, the number of local maximum points and curvature-based features (such as features based on the proportion of points with negative Gaussian and negative mean curvature (hyperbolic concave points) and positive Gaussian and negative mean curvature (elliptical concave points) as well as the proportion of flat points) to determine if we can distinguish between the 3 classes. We will use Support Vector Machine (SVM) with RBF kernel as a classification method, and additionally use decision trees to identify the most significant features. We compute the accuracy against the ground truth, and investigate the failing cases in order to improve on the suggested methods. We will further compare the results with deep learning approaches from computer vision.

---

**References**

---

- 1 Efren Andablo-Reyes, Michael Bryant, Anne Neville, Paul Hyde, Rik Sarkar, Mathew Francis, and Anwesha Sarkar. 3d biomimetic tongue-emulating surfaces for tribological applications. *ACS applied materials & interfaces*, 12(44):49371–49385, 2020.
- 2 Paul J Besl and Ramesh C Jain. Invariant surface characteristics for 3d object recognition in range images. *Computer vision, graphics, and image processing*, 33(1):33–80, 1986.
- 3 Paolo Cignoni, Marco Callieri, Massimiliano Corsini, Matteo Dellepiane, Fabio Ganovelli, and Guido Ranzuglia. Meshlab: an open-source mesh processing tool. In *Eurographics Italian chapter conference*, volume 2008, pages 129–136. Salerno, Italy, 2008.
- 4 Alessandro Colombo, Claudio Cusano, and Raimondo Schettini. 3d face detection using curvature analysis. *Pattern recognition*, 39(3):444–455, 2006.
- 5 Manfredo P Do Carmo. *Differential geometry of curves and surfaces: revised and updated second edition*. Courier Dover Publications, 2016.
- 6 Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics (ToG)*, 32(3):1–13, 2013.
- 7 Mark Meyer, Mathieu Desbrun, Peter Schröder, and Alan H Barr. Discrete differential-geometry operators for triangulated 2-manifolds. In *Visualization and mathematics III*, pages 35–57. Springer, 2003.
- 8 Inglis J Miller Jr and Frank E Reedy Jr. Quantification of fungiform papillae and taste pores in living human subjects. *Chemical Senses*, 15(3):281–294, 1990.
- 9 Carolina Segovia, Ian Hutchinson, David G Laing, and Anthony L Jinks. A quantitative study of fungiform papillae and taste pore density in adults and children. *Developmental Brain Research*, 138(2):135–146, 2002.
- 10 Maryam Shahbake, Ian Hutchinson, David G Laing, and Anthony L Jinks. Rapid quantitative assessment of fungiform papillae density in the human tongue. *Brain research*, 1052(2):196–201, 2005.

# Enumerating All Convex Polyhedra Glued from Squares in Polynomial Time\*

Stefan Langerman 

Faculté des Sciences, Université Libre de Bruxelles

Nicolas Potvin 

Faculté des Sciences, Université Libre de Bruxelles

Boris Zolotov 

Department of Mathematics and Computer Sciences, St. Petersburg State University

---

## Abstract

---

We present an algorithm that enumerates and classifies all edge-to-edge gluings of unit squares that correspond to convex polyhedra. We show that the number of such gluings of  $n$  squares is polynomial in  $n$ , and the algorithm runs in time polynomial in  $n$  (pseudopolynomial if  $n$  is considered the only input). Our technique can be applied in several similar settings, including gluings of regular hexagons and triangles.

**2012 ACM Subject Classification** #10010061 Computational geometry

**Keywords and phrases** polyhedral metrics, alexandrov theorem, squares, edge-to-edge gluings

**Related Version** <http://arxiv.org/abs/2104.06787>

**Funding** Stefan Langerman: is Directeur de recherches du F.R.S.-FNRS

Boris Zolotov: is supported in part by the Foundation for the Advancement of Theoretical Physics and Mathematics “BASIS” and in part by “Native towns”, a social investment program of PJSC “Gazprom Neft”.

**Acknowledgements** B. Z. wants to thank the international academic mobility & students exchange program of SPBU for offering him an exchange in ULB that allowed for cooperation needed to complete this project.

## 1 Introduction

Given a collection of 2D polygons, a *gluing* describes a closed surface by specifying how to glue each edge of these polygons onto another edge. Alexandrov’s uniqueness theorem [1] states that any valid gluing that is homeomorphic to a sphere and that does not yield a total facial angle greater than  $2\pi$  at any point, corresponds to the surface of a unique convex 3D polyhedron. The polygons of the gluing may be folded in order to glue the polyhedron.

There is no known exact algorithm for reconstructing the 3D polyhedron [8, 9]. Enumerating all possible valid gluings is also not an easy task, as the number of gluings can be exponential even for a single polygon [5]. Complete enumerations of gluings and the resulting polyhedra are only known for very specific cases such as the Latin cross [6], a single regular convex polygon [7], and a collection of regular pentagons [2].

The case when the polygons to be glued together are all identical regular  $k$ -gons, and the gluing is *edge-to-edge* was studied recently for  $k \geq 6$  [3]. The aim of this paper is to study the case of  $k = 4$ : namely, to *enumerate* all valid gluings of squares and *classify* them up to isomorphism.

---

\* This is an abstract of a presentation given at CG:YRF 2021. 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.



© Stefan Langerman and Nicolas Potvin and Boris Zolotov;  
licensed under Creative Commons License CC-BY 4.0



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

## Convex Polyhedra Glued from Squares

### 2 Chen—Han algorithm for gluings of squares

In [7] it is shown that polyhedra are isomorphic if the lengths of shortest geodesic paths between their vertices of nonzero curvature coincide. Thus, the problem of finding out if two gluings are isomorphic can be reduced to finding out the geodesic distances between vertices of a gluing. Algorithm we are using for this is the Chen—Han algorithm [4].

The idea of the algorithm is to project a cone of all possible paths from the source onto the surface of the gluing. For  $n$  faces, this algorithm runs in  $O(n^2)$  time. To apply it for arbitrary edge-to-edge gluings of squares, it has to be proven that the running time is preserved. To do this, we prove the following Lemma.

► **Lemma 1.** *If  $T$  is a square of the gluing and  $\pi$  is a geodesic shortest path between two vertices of the gluing then the intersection between  $\pi$  and  $T$  is of at most 5 segments.*

The lemma implies the following Theorem.

► **Theorem 2.** *The isomorphism between two edge-to-edge gluings of at most  $n$  squares can be tested in  $O(n^2)$  time.*

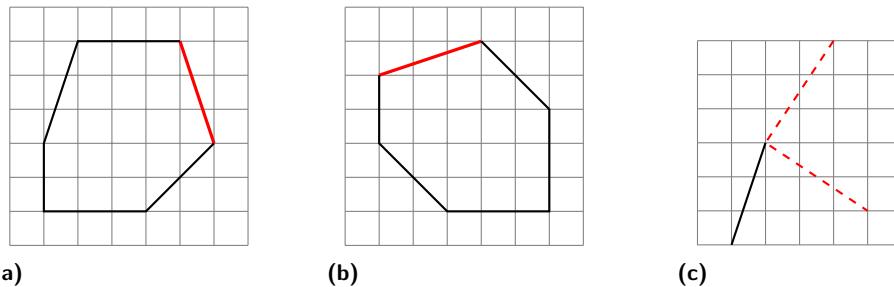
### 3 Bounds on the number of egde-to-edge gluings of squares

In this section, we prove that the number of edge-to-edge gluings of  $n$  squares is polynomial in  $n$ . This result allows to develop a polynomial algorithm to list all the gluings.

► **Theorem 3.** *There are  $O(n^{36})$  edge-to-edge gluings of at most  $n$  squares that correspond to convex polyhedra.*

**Proof.** Triangulate the polyhedron corresponding to the gluing and draw its faces on the square grid. By the Gauss-Bonnet theorem, the polyhedron has no more than 8 vertices, and thus at most 18 edges. An edge shared by two faces must have the same lengths of  $x$ - and  $y$ -projections on the drawings of these faces, see Figures 1a, 1b.

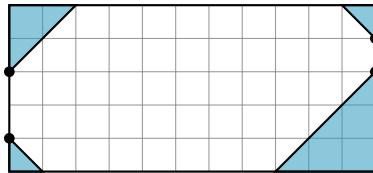
Count the number of sets of triangles satisfying this restriction and taking up at most  $n$  squares. To do so, choose the lengths of projections (that do not exceed  $n$  in length) for each of at most 18 edges. This yields the final formula. ◀



► **Figure 1** (a), (b) Highlighted edge has the same lengths of projections on the drawings of two faces. (c) Two ways to place an edge with given projections that preserve convexity of the face.

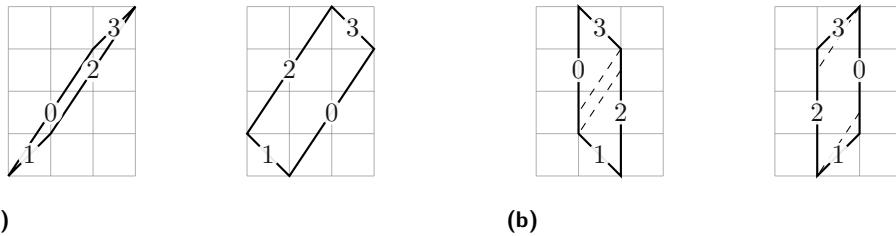
► **Theorem 4.** *There are  $\Omega(n^3)$  edge-to-edge gluings of at most  $n$  squares that correspond to convex polyhedra.*

**Proof.** To prove the theorem, we construct a series of such gluings. These gluings correspond to doubly-covered octagons, the octagons being obtained by cutting edges of a rectangle with sides no longer than  $\frac{\sqrt{n}}{2}$ , one at least twice as long as the other, see Figure 2. The bound is tight: there are  $O(n^3)$  doubly covered convex polygons that can be glued from  $n$  squares. ◀



■ **Figure 2** An example of an octagon produced by cutting angles of a rectangle

We implemented an algorithm that enumerates all the gluings of at most  $n$  squares for a given graph structure of a convex polyhedron. It showed that one gluing can admit several ways to cut itself into flat polygons, see Figure 3. Thus it can appear in the list several times.



■ **Figure 3** Doubly covered parallelogram can be cut into two flat quadrilaterals in two ways, the latter consisting of its faces

## 4 Algorithm to classify edge-to-edge gluings of squares

The algorithm consists of the following steps:

1. Generate the list of all edge-to-edge gluings of at most  $n$  squares, denote it  $L(n)$ . Due to Theorem 3, this step takes polynomial time.
2. For each gluing in  $L(n)$ , generate matrix of pairwise distances between its vertices. Due to Theorem 2, this step takes  $O(n^3)$  time per gluing.
3. Unic平化 the list of matrices up to homothety and permutation of rows and columns, leave only corresponding elements of  $L(n)$ . Since the matrices are of at most 8 rows and 8 columns, it takes polynomial time to remove duplicates from the list.

The output of this algorithm is the list of all non-isomorphic edge-to-edge gluings of at most  $n$  squares.

## 5 Discussion

The cornerstone of the technique we have been using is the possibility to draw a face of a polyhedron glued from squares on a planar grid. It allows us to estimate the number of valid gluings. The same technique can seemingly be applied for the cases of regular hexagons and triangles, since these polygons also tile the plane.

---

### References

---

- 1    Alexandr Alexandrov. *Convex Polyhedra*. Springer-Verlag, Berlin, 2005.
- 2    E. Arseneva, S. Langerman, and B. Zolotov. A complete list of all convex shapes made by gluing regular pentagons. In *XVIII Spanish Meeting on Computational Geometry*, page 1–4, Girona, Spain, 2019.
- 3    Elena Arseneva and Stefan Langerman. Which Convex Polyhedra Can Be Made by Gluing Regular Hexagons? *Graphs and Combinatorics*, page 1–7, 2019. doi:DOI:10.1007/s00373-019-02105-3.
- 4    Jindong Chen and Yijie Han. Shortest paths on a polyhedron. In *6-th annual symposium on Computational geometry*, page 360–369, Berkley, California, USA, June 1990. SCG ’90.
- 5    Erik Demaine, Martin Demaine, Anna Lubiw, and Joseph O’Rourke. Enumerating foldings and unfoldings between polygons and polytopes. *Graphs and Combinatorics*, 18(1):93–104, 2002.
- 6    Erik Demaine, Martin Demaine, Anna Lubiw, Joseph O’Rourke, and Irena Pashchenko. Metamorphosis of the cube. In *Proc. SOCG*, pages 409–410. ACM, 1999.
- 7    Erik Demaine and Joseph O’Rourke. *Geometric folding algorithms*. Cambridge University Press, 2007.
- 8    David Eppstein, Michael J Bannister, William E Devanny, and Michael T Goodrich. The Galois complexity of graph drawing: Why numerical solutions are ubiquitous for force-directed, spectral, and circle packing drawings. In *International Symposium on Graph Drawing*, pages 149–161. Springer, 2014.
- 9    Daniel M Kane, Gregory N Price, and Erik D Demaine. A Pseudopolynomial Algorithm for Alexandrov’s Theorem. In *WADS*, pages 435–446. Springer, 2009.

# The Visibility Center of a Polygon\*

Anna Lubiw 

David R. Cheriton School of Computer Science, University of Waterloo, Canada

Anurag Murty Naredla 

David R. Cheriton School of Computer Science, University of Waterloo, Canada

---

## Abstract

---

We introduce the *visibility center* of a set of points inside a simple polygon—a point  $c$  such that the maximum geodesic distance from  $c$  to see any point in the set is minimized. For a simple polygon of  $n$  vertices and a set of  $m$  points inside it, we give an  $O((m+n)\log(m+n))$  algorithm to find the visibility center. We find the visibility center of *all* points in the polygon in  $O(n\log n)$  time.

**2012 ACM Subject Classification** Theory of computation → Computational geometry

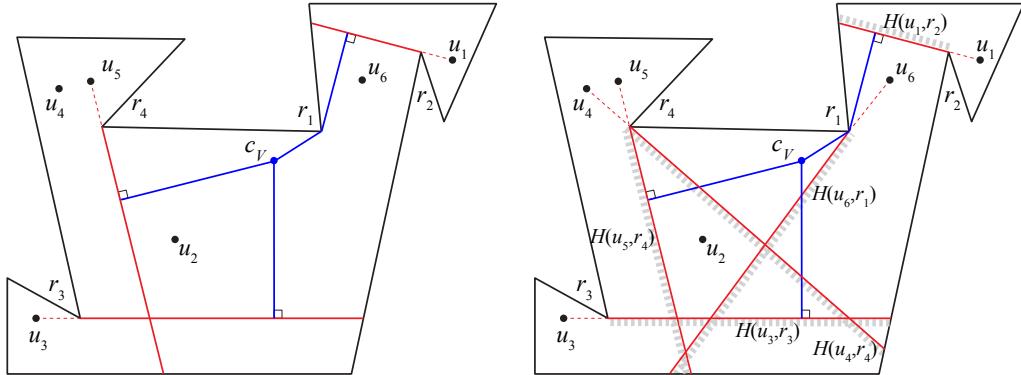
**Keywords and phrases** Visibility Algorithms, Shortest Path Algorithms

**Digital Object Identifier** 10.4230/LIPIcs.CVIT.2016.23

**Funding** Research supported by NSERC.

## 1 Introduction

Suppose you want to guard a polygon and you have many sensors but only one guard to check on the sensors. The guard must be positioned at a point  $c$  in the polygon such that when a sensor at any query point  $u$  sends an alarm, the guard travels from  $c$  on a shortest path inside the polygon to see point  $u$ ; the goal is to minimize the maximum distance the guard must travel. The optimum guard position  $c$  is called the *visibility center* of the set  $U$  of possible query points. See Figure 1.



**Figure 1** (left) Point  $c_V$  is the *visibility center* of points  $U = \{u_1, \dots, u_6\}$ . Starting from  $c_V$ , the three points we need to travel (equally) farthest to see are  $u_1, u_3$  and  $u_5$ . The shortest paths (in blue) to see these points must reach the half-polygons bounded by the chords (in red) emanating from the points. (right) Equivalently,  $c_V$  is the geodesic center of five half-polygons (each shown as a red boundary chord shaded on one side).

---

\* This is an abstract of a presentation given at CG:YRF 2021. 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 an  $O((m + n) \log(m + n))$  time algorithm to find the visibility center of a set  $U$  of size  $m$  in an  $n$ -vertex simple polygon. To find the visibility center of *all* points inside a simple polygon, we can restrict our attention to the vertices of the polygon, which yields an  $O(n \log n)$  time algorithm.

To the best of our knowledge, the idea of visibility centers is new. There are many results on the *geodesic center* of a polygon—the point  $c$  that minimizes the maximum distance inside the polygon from  $c$  to any vertex of the polygon. Pollack et al. [14] gave an  $O(n \log n)$  time divide-and-conquer algorithm. Our algorithm builds on theirs. A more recent algorithm finds the geodesic center in linear time [1]. There are efficient algorithms to construct the whole geodesic farthest point Voronoi diagram (of which the geodesic center is a vertex) for given points in a polygon [4, 5, 13, 17].

There are algorithms for the more basic “quickest visibility problem” in a polygon—to find the shortest path from point  $s$  to see point  $q$  [3, 16] or for two points to travel until they see each other [2], but we do not use these results. Also related is the watchman problem [7, 8]—to find a tour that sees the whole polygon.

**Basic Idea.** Our algorithm for computing the visibility center has two main steps. We first identify a linear sized set  $\mathcal{H}$  of “essential” half-polygons that the guard must travel to, i.e., so that the visibility center of  $U$  is the geodesic center of  $\mathcal{H}$ . Second, we find the geodesic center of any set of half-polygons by extending the divide-and-conquer approach that Pollack et al. [14] used to compute the geodesic center of the vertices of a simple polygon. These two steps are described in Sections 2 and 3.

**Notation and Definitions.** For a point  $u$  in polygon  $P$ ,  $P_V(u)$  is the *visibility polygon* of  $u$ —the set of all points in  $P$  visible from  $u$ . Any edge of  $P_V(u)$  that is not part of the boundary  $\partial P$  is a *window* that is determined by a reflex vertex  $r$  of  $P$  visible from  $u$  and is formed by extending the ray  $\overrightarrow{ur}$  from the *base*  $r$  until it hits  $\partial P$  at the *tip*. We use  $w(u, r)$  to denote this window. The subpolygon cut off by  $w(u, r)$  that contains  $u$  and  $P_V(u)$  is called the *half-polygon* associated with  $u$  and  $r$ , and is denoted  $H(u, r)$ . The complement,  $\overline{H}(u, r)$  is the *pocket* associated with  $u$  and  $r$ .

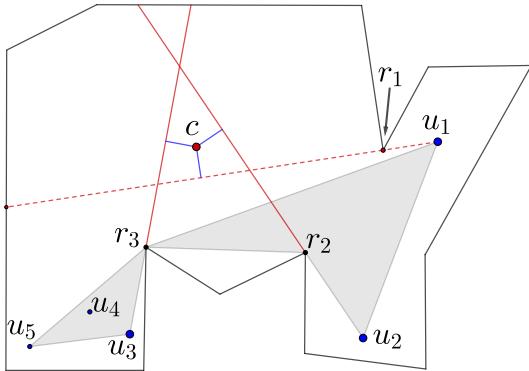
For points  $x, u \in P$ , the *distance to visibility* from  $x$  to  $u$ , denoted  $d_V(x, u)$  is the geodesic distance from  $x$  to  $P_V(u)$ . If  $x \in P_V(u)$  then  $d_V(x, u) = 0$ . Otherwise,  $x$  lies in some pocket  $\overline{H}(u, r)$  and  $d_V(x, u)$  is the geodesic distance from  $x$  to the half-polygon  $H(u, r)$ . The *visibility radius* of point  $x$  with respect to point set  $U$  is  $r_V(x, U) = \max\{d_V(x, u) : u \in U\}$ . The *visibility center* of  $U$  is  $\operatorname{argmin}\{r_V(x, U) : x \in P\}$ .

## 2 Finding a linear number of essential half-polygons

The visibility center for a given set  $U$  of points is the geodesic center of all  $O(mn)$  half polygons of the form  $H(u, r)$  where  $u \in U$  and  $r$  is a reflex vertex of the polygon and  $u$  sees  $r$ . But this set is too large. We find a set  $\mathcal{H}$  of  $O(n)$  “essential” half polygons that suffice, i.e., such that the visibility center of  $U$  is the geodesic center of the half polygons of  $\mathcal{H}$ .

We find  $\mathcal{H}$  in two steps. First make a subset  $\mathcal{H}_0$  as follows. Construct  $R$ , the geodesic convex hull of  $U$  in  $P$  in time  $O(n + m \log(m + n))$  [9, 15]. For each edge  $(u, r)$  of  $R$  where  $u \in U$  and  $r$  is a reflex vertex of  $P$ , put  $H(u, r)$  into  $\mathcal{H}_0$ . Then  $\mathcal{H}_0$  has linear size but, as shown in Figure 2,  $\mathcal{H}_0$  does not yet have the property we need.

Next, construct the geodesic center  $c_0$  of  $\mathcal{H}_0$ . Then repeat the above step for  $U \cup \{c_0\}$ , i.e., construct  $R'$ , the geodesic convex hull of  $U \cup \{c_0\}$  in  $P$  and for each edge  $(u, r)$  of  $R'$  where  $u \in U$  and  $r$  is a reflex vertex of  $P$ , add  $H(u, r)$  to  $\mathcal{H}_0$ . This defines  $\mathcal{H}$ . We apply



**Figure 2** The geodesic convex hull of  $U = \{u_1, \dots, u_5\}$  is shaded grey.  $\mathcal{H}_0$  consists of the two half-polygons  $H(u_2, r_2)$  and  $H(u_3, r_3)$  (with solid red windows). The second phase finds the third half-polygon  $H(u_1, r_1)$  that determines the visibility center  $c$ .

an  $O(n \log n)$  time ray shooting algorithm [10] to find, for each  $H(u, r) \in \mathcal{H}$ , the tip of the window  $w(u, r)$  so each half-polygon is specified by two points on  $\partial P$ .

► **Theorem 1.** *The visibility center of  $U$  is the geodesic center of  $\mathcal{H}$ . Furthermore  $\mathcal{H}$  can be found in time  $O((n + m) \log(n + m))$ .*

### 3 Finding the geodesic center of half-polygons

► **Theorem 2.** *There is an  $O(n \log n)$  time algorithm to find the geodesic center of a set of  $O(n)$  half-polygons in a polygon of size  $n$ .*

We follow the approach of Pollack et al. [14]. The main ingredient is an  $O(n)$  time *chord oracle* that, given a chord  $K$  of the polygon, finds the *relative geodesic center* on  $K$ , i.e.,  $\operatorname{argmin}\{r_V(x, U) : x \in K\}$ , and tells us which side of  $K$  contains the geodesic center  $c$ . After triangulating the polygon,  $O(\log n)$  applications of the chord oracle are used to locate the triangle containing the geodesic center. This triangle is then refined to a region within which the geodesic paths to the half-polygons are combinatorially the same, resulting in an intersection radius problem [6], which is solved by Megiddo's techniques [12, 11].

To implement the chord oracle for chord  $K = ab$  we first find shortest path trees from  $a$  and from  $b$  to all the half-polygons. From these we get a set of  $O(n)$  overlapping intervals on  $ab$  each with an easily-computable distance function to an associated half-polygon and with the property that the upper envelope of these functions equals the geodesic radius function. Megiddo's techniques then find the relative geodesic center on  $K$ .

---

### References

---

- 1 Hee-Kap Ahn, Luis Barba, Prosenjit Bose, Jean-Lou De Carufel, Matias Korman, and Eunjin Oh. A linear-time algorithm for the geodesic center of a simple polygon. *Discrete & Computational Geometry*, 56(4):836–859, 2016. URL: <https://doi.org/10.1007/s00454-016-9796-0>.
- 2 Hee-Kap Ahn, Eunjin Oh, Lena Schlipf, Fabian Stehn, and Darren Strash. On romeo and juliet problems: Minimizing distance-to-sight. *Computational Geometry*, 84:12–21, 2019. URL: <https://doi.org/10.1016/j.comgeo.2019.07.003>.
- 3 Esther M Arkin, Alon Efrat, Christian Knauer, Joseph S.B. Mitchell, Valentin Polishchuk, Günter Rote, Lena Schlipf, and Topi Talvitie. Shortest path to a segment and quickest visibility

- queries. *Journal of Computational Geometry*, 7:77–100, 2016. URL: <https://jocg.org/index.php/jocg/article/view/3001>.
- 4 Boris Aronov, Steven Fortune, and Gordon Wilfong. The furthest-site geodesic Voronoi diagram. *Discrete & Computational Geometry*, 9(3):217–255, 1993. URL: <https://doi.org/10.1145/73393.73417>.
  - 5 Luis Barba. Optimal algorithm for geodesic farthest-point Voronoi diagrams. In *35th International Symposium on Computational Geometry (SoCG 2019)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2019. URL: <http://drops.dagstuhl.de/opus/volltexte/2019/10416>.
  - 6 Binay K Bhattacharya, Shreesh Jadhav, Asish Mukhopadhyay, and J-M Robert. Optimal algorithms for some intersection radius problems. *Computing*, 52(3):269–279, 1994. URL: <https://doi.org/10.1007/bf02246508>.
  - 7 Wei-Pang Chin and Simeon Ntafos. Shortest watchman routes in simple polygons. *Discrete & Computational Geometry*, 6(1):9–31, 1991. URL: <https://doi.org/10.1007/bf02574671>.
  - 8 Moshe Dror, Alon Efrat, Anna Lubiw, and Joseph SB Mitchell. Touring a sequence of polygons. In *Proceedings of the thirty-fifth Annual ACM Symposium on Theory of Computing (STOC '03)*, pages 473–482, 2003. URL: <https://doi.org/10.1145/780542.780612>.
  - 9 Leonidas J Guibas and John Hershberger. Optimal shortest path queries in a simple polygon. *Journal of Computer and System Sciences*, 39(2):126–152, 1989. URL: [https://doi.org/10.1016/0022-0000\(89\)90041-x](https://doi.org/10.1016/0022-0000(89)90041-x).
  - 10 John Hershberger and Subhash Suri. A pedestrian approach to ray shooting: Shoot a ray, take a walk. *Journal of Algorithms*, 18(3):403–431, 1995. URL: <https://doi.org/10.1006/jagm.1995.1017>.
  - 11 Nimrod Megiddo. Linear-time algorithms for linear programming in  $R^3$  and related problems. *SIAM Journal on Computing*, 12(4):759–776, 1983. URL: <https://doi.org/10.1137/0212052>.
  - 12 Nimrod Megiddo. On the ball spanned by balls. *Discrete & Computational Geometry*, 4(6):605–610, 1989. URL: <https://doi.org/10.1007/bf02187750>.
  - 13 Eunjin Oh, Luis Barba, and Hee-Kap Ahn. The geodesic farthest-point Voronoi diagram in a simple polygon. *Algorithmica*, 82(5):1434–1473, 2020. URL: <https://doi.org/10.1007/s00453-019-00651-z>.
  - 14 Richard Pollack, Micha Sharir, and Günter Rote. Computing the geodesic center of a simple polygon. *Discrete & Computational Geometry*, 4(6):611–626, 1989. URL: <https://doi.org/10.1007/bf02187751>.
  - 15 Godfried T Toussaint. Computing geodesic properties inside a simple polygon. *Revue d'Intelligence Artificielle*, 3(2):9–42, 1989.
  - 16 Haitao Wang. Quickest visibility queries in polygonal domains. *Discrete & Computational Geometry*, 62(2):374–432, 2019. URL: <https://doi.org/10.1007/s00454-019-00108-8>.
  - 17 Haitao Wang. An optimal deterministic algorithm for geodesic farthest-point Voronoi diagrams in simple polygons. In *International Symposium on Computational Geometry (SoCG 2021)*, 2021. URL: <https://arxiv.org/pdf/2103.00076.pdf>.

# Package delivery using handoffs among collaborating heterogeneous agents

Kien C. Huynh 

Department of Computer Science, Stony Brook University, USA

Joseph S. B. Mitchell  

Department of Applied Mathematics and Statistics, Stony Brook University, USA

---

## Abstract

---

We consider the problem of transporting a package from point  $s$  to point  $t$  in the Euclidean plane utilizing a fleet of  $n$  heterogeneous carrier agents who can hand off the package from one agent to another. The agents are initially deployed at given points in the plane, and each has an associated maximum speed, a fuel constraint, and possibly other constraints on its motion. The objective is to compute a delivery plan that minimizes the time required to get the package to its destination  $t$ .

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

**Keywords and phrases** Package delivery, vehicle routing, collaborating agents, makespan minimization

## 1 Introduction

Given  $n$  agents, initially at points  $\{a_i\}_{i=1}^n$  in the plane or a polygonal domain  $P$ , with different maximum speeds  $\{v_i\}_{i=1}^n$ , and potentially different fuel bounds  $\{b_i\}_{i=1}^n$  and subdomains  $Q_i \subseteq P$  of operation ( $a_i$  can only move inside  $Q_i$ ), we seek to compute an efficient set of trajectories for the agents to be able to transport the package from source  $s$  to target  $t$  in the minimum total time. Since the agents have different speeds, a slower agent can hand off the package to a faster agent to reduce the total makespan, and an agent running out of fuel can hand off the package to an agent with fuel. The hand-off and pickup mechanism are assumed to be instantaneous and the agents are not allowed to leave the package anywhere, even temporarily.

This abstract is based on work in progress, which includes theoretical considerations as well as experimental results, on a variety of package delivery problems in the plane.

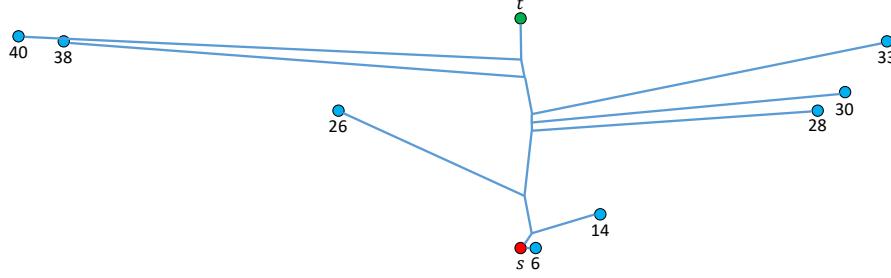
## 2 Agents with unlimited fuel

We begin with the case in which there are no fuel bounds ( $b_i = \infty$ ). To determine an optimal solution we need to determine the best subset and sequence of agents to use as well as the locations of the handoff points. We know, in this case, that each handoff must be to a faster agent; thus, once we determine which subset of agents to utilize, the sequence is determined. Handoff locations in the Euclidean plane are potentially challenging to compute exactly; even for the case of 2 agents involved in the transport, we show that one must solve a quartic equation. Currently, the hardness of the problem is not yet known. If handoff locations are given as inputs and restricted to a discrete set of points, e.g., at vertices of a graph  $G$ , the problem can be solved exactly in polynomial time [1, 2]. The best known running time is  $O(nv \log nv + ne)$ , for a graph  $G$  with  $v$  vertices and  $e$  edges. There is also a somewhat

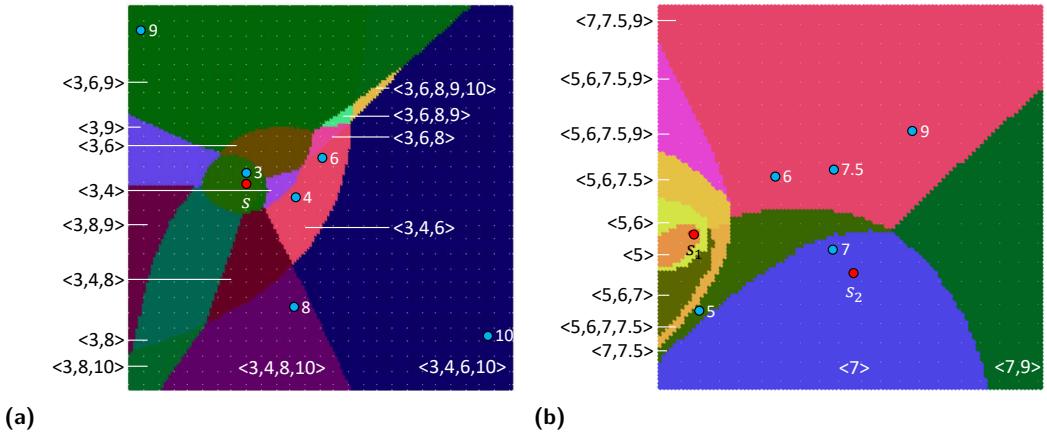
---

This is an abstract of a presentation given at CG:YRF 2021. 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.

## Package delivery using handoffs



**Figure 1** An example solution for the unlimited fuel case. This was computed on a discrete set of handoff points. Blue dots are agents, each labeled with their speed.



**Figure 2** Example: Blue dots are agent initial locations, labeled with speeds. We color points ( $t$ ) so that points that are delivered by the same subset/sequence of agents share the same color. Each tuple, for example  $\langle 3, 4, 6 \rangle$ , shows the best subset/sequence of agents that are used for delivery to points in the region. Left: There is a single source,  $s$ . Right: There are two sources,  $s_1$  and  $s_2$ , each with the same product to be delivered to  $t$ , so agents can transport from either source.

slower dynamic programming algorithm on graphs, taking  $O(n^2e + nv^2 + \text{APSP})$  time, where APSP is the time required for all-pairs shortest paths. Using an appropriately selected grid resolution ( $\delta = \frac{\epsilon v_{\min}(|a_0s| + |st|)}{v_{\max}n\sqrt{2}}$  where  $a_0$  is location of first agent) of candidate handoff points within a region containing segment  $st$  that is bounded based on an upper bound on optimal makespan, we are able to give a PTAS for the single package handoff problem in the plane. In a more careful analysis, we know that  $O(\frac{n^3}{\epsilon^2})$  candidate points are always enough for a  $(1 + \epsilon)$  approximation. For polygonal domains, these results also apply if we are to use a discrete set of handoff points and construct a visibility graph from them.

We have implemented and experimented with the DP algorithm, applying it to geometric instances based on a simple grid of candidate handoff points. Figure 1 shows an example of optimal delivery plan. For each grid point, we can label (color code) them according to the optimal combination of agents (and choice of source, if more than one source) utilized in an optimal delivery. Figure 2 shows two such visualizations; the boundary between regions consist of handoff locations. The resulting subdivision, the *handoff map*, is a form of shortest path map or Voronoi diagram, decomposing the plane into regions according to the

combinatorial type of the handoff strategy (and the choice of source) for optimal transport from a source to each point  $t$  in the plane. We are actively investigating the combinatorial complexity of handoff maps and hope to report upper/lower bounds at the YRF.

Without discretizing with a grid, we examine exact solutions involving the best use of two agents (from among the  $n$ ), leading to a quartic equation for the optimal handoff points. We prove that this yields a 3-approximation.

We also give an  $O(n \log n)$  time algorithm to compute an optimal use of the  $n$  agents to transport the package from  $s$  to  $t$  under the constraint that the package moves on segment  $st$ ; we prove this yields a 2-approximation.

### 3 Multiple packages

If there are multiple packages at multiple sources  $\{s_j\}_{j=1}^p$  which need to be delivered to multiple corresponding targets  $\{t_j\}_{j=1}^p$ , then our problem is NP-hard (by a reduction from the EUCLIDEAN STACKER CRANE PATH problem [5], with a single agent, assuming all packages have the same weight and the agent can carry a single package). If we require the agent to return to their original location and there is only one agent, then there is a 9/5-approximation based on the EUCLIDEAN STACKER CRANE TOUR problem [5].

### 4 Agents with bounded fuel

When agents have bounded fuel, i.e. agent  $a_i$  can only travel a distance at most  $b_i < \infty$ , the problem becomes significantly harder. Even determining feasibility is challenging, not to mention optimizing total delivery time. The problem is weakly NP-complete even in 1D [4]. Even if all agents have the same fuel bound,  $b$ , the problem is NP-complete [3] in a graph; we are investigating the complexity in the Euclidean plane. We obtain bounds on the fuel bound,  $b$ , namely:  $\frac{1}{2}w_{\max} < b \leq (2 + \frac{1}{2^{n^2}-1})w_{\max}$ , where  $w_{\max}$  is the length of the longest edge in the path connecting  $s$  to  $t$  in the minimum spanning tree of the agents and  $s, t$  (unless  $st$  is an edge of the MST, in which case a (tight) upper bound on  $b$  is given by  $d(a_s, s) + d(s, t)$  where  $a_s$  is the agent closest to  $s$ ).

---

### References

- 1 Andreas Bärtschi, Daniel Graf, and Matús Mihalák. Collective fast delivery by energy-efficient agents. *arXiv preprint arXiv:1809.00077*, 2018.
- 2 Iago A. Carvalho, Thomas Erlebach, and Kleitos Papadopoulos. An efficient algorithm for the fast delivery problem. In *International Symposium on Fundamentals of Computation Theory*, pages 171–184. Springer, 2019.
- 3 Jérémie Chalopin, Shantanu Das, Matúš Mihalák, Paolo Penna, and Peter Widmayer. Data delivery by energy-constrained mobile agents. In *International Symposium on Algorithms and Experiments for Sensor Systems, Wireless Networks and Distributed Robotics*, pages 111–122. Springer, 2013.
- 4 Jérémie Chalopin, Riko Jacob, Matúš Mihalák, and Peter Widmayer. Data delivery by energy-constrained mobile agents on a line. In *International Colloquium on Automata, Languages, and Programming*, pages 423–434. Springer, 2014.
- 5 Greg N. Frederickson, Matthew S. Hecht, and Chul E. Kim. Approximation algorithms for some routing problems. In *17th Annual Symposium on Foundations of Computer Science*, pages 216–227. IEEE, 1976.

# On the adjacency structures of planar point-set triangulations

Logan D. Graham

Department of Applied Mathematics & Statistics, Stony Brook University, U.S.A.

logan.graham@stonybrook.edu

---

## Abstract

---

Considering triangulations whose triangles share a common adjacency structure to be isomorphic, we characterize the set of all triangulations of finite point-sets in spaces homeomorphic to the Euclidean plane. We achieve this by defining a new family of graphs and by proving necessary and sufficient conditions for inclusion therein. Specifically, we characterize the set of all graphs weakly dual to triangulated finite point-sets in the Euclidean plane.<sup>1</sup>

**2012 ACM Subject Classification** Mathematics of computing → Graphs and surfaces; Theory of computation → Computational geometry

**Keywords and phrases** Triangulation, Graph Embedding, Weak Dual, Topological Graph Theory

**Digital Object Identifier** 10.4230/LIPIcs.CVIT.2016.23

**Funding** Part of this research was supported by the Institute for Advanced Computational Science STRIDE Ph.D. Fellowship, NSF grant DGE-1633299.

**Acknowledgements** We thank the SoCG CG:SHOP 2019 organizers, out of whose problem-statements our questions tangentially developed. In particular, we thank our advisor, Joseph S. B. Mitchell, for asking penetrating questions and suggesting exciting follow-up directions.

## 1 Preliminaries

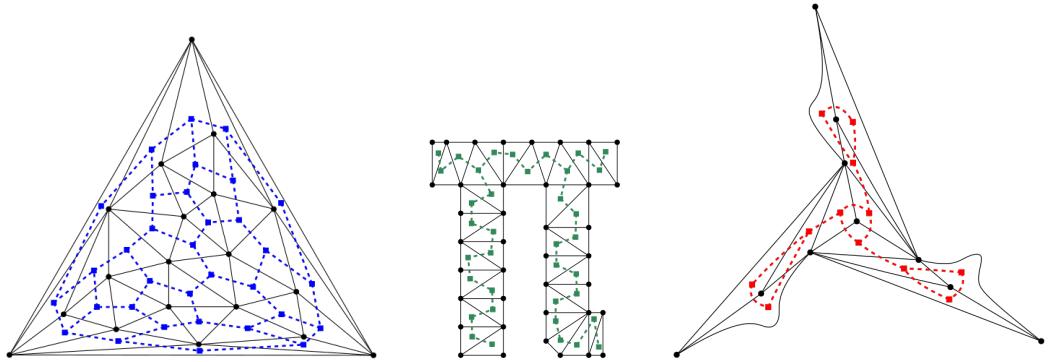
In a planar embedding of a graph, a face is *strictly convex* if its boundary is convex and no three points along its boundary are collinear. Consider a planar embedding of a connected, simple graph (i.e., that without duplicate edges or self-loops) in which all faces, except for at most one – the *infinite face*  $f_\infty$  – are strictly convex with boundaries formed by exactly three edges. Suppose that the infinite face’s boundary  $\partial f_\infty$  is a simple, non-degenerate, closed curve. Such an embedding is a *planar point-set triangulation*.<sup>2,3</sup> Therein, the degree-three faces  $f$  such that  $f \neq f_\infty$  are called *triangles*. A *polygonal triangulation* is a planar point-set triangulation in which all vertices are included in the cycle embedded as the infinite face’s boundary. Whereas planar point-set triangulations can feature vertices that are embedded anywhere in  $\mathbb{R}^2 \setminus f_\infty$ , polygonal triangulations must have all vertices embedded in  $\partial f_\infty$ . The former is a generalization of the latter, as  $\partial f_\infty \subseteq \{\mathbb{R}^2 \setminus f_\infty\}$ . It is standard in the graph-theoretic literature to define a *triangulation* as a graph that, when embedded in the plane, has a 3-regular dual [4, 7, 11] (e.g., the left embedding in Figure 1). Crucially, see that

---

<sup>1</sup> This is an abstract of a presentation given at CG:YRF 2021. 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.

<sup>2</sup> A planar point-set triangulation is a strictly convex drawing in which (i) convexity is relaxed for the infinite face  $f_\infty$ , and (ii) all faces, aside from  $f_\infty$ , are bounded by exactly three edges. See [1, 2, 3, 8, 9] for a taste of the pertinent subset of the graph drawing literature.

<sup>3</sup> Since all faces are convex, except for potentially  $f_\infty$ , every edge  $e$  in a planar point-set triangulation such that  $e \notin \partial f_\infty$  must be drawn as a line segment. W.l.o.g., we may assume that the edges that form  $\partial f_\infty$  are drawn as line segments as well, thanks to the famous result by Fáry [5] (which he proved by reasoning about triangulations, ironically). Some of our figures reflect this assumption.



**Figure 1** planar point-set triangulations (black, solid) and their weak duals (color, dashed).

our definition of a planar point-set triangulation generalizes the standard graph-theoretic definition of a triangulation.

Fix an embedding  $\tilde{G}$  of a planar graph  $G$ . Construct a new graph  $G^*$  whose vertices correspond to  $\tilde{G}$ 's faces. In  $G^*$ , include an edge  $uv$  for each edge shared by the faces reciprocal to  $u$  and  $v$  in  $\tilde{G}$ .  $G^*$  is  $\tilde{G}$ 's *dual graph*. Removing from  $G^*$  the vertex  $v_\infty$  corresponding to  $\tilde{G}$ 's infinite face, along with each of  $v_\infty$ 's incident edges, yields  $\tilde{G}$ 's *weak dual graph*.<sup>4</sup>

## 2 Planar point-set triangulation weak dual graphs

A graph that can be embedded in the plane such that each of its vertices is incident to the infinite face is called *outerplanar*. An outerplanar graph is said to be *maximal outerplanar* if the addition of an edge joining two non-adjacent vertices yields a non-outerplanar graph. A graph is maximal outerplanar if and only if it is a polygonal triangulation, and such graphs can be recognized in linear time [6]. We now characterize this family's weak dual graphs.

► **Remark 1.** There exists a polygonal triangulation  $\mathcal{T}$  to which a tree  $T$  is weakly dual if and only if  $\Delta(T) \leq 3$ . The same result applies to  $\mathcal{T}$  in which the infinite face is convex.<sup>5</sup>

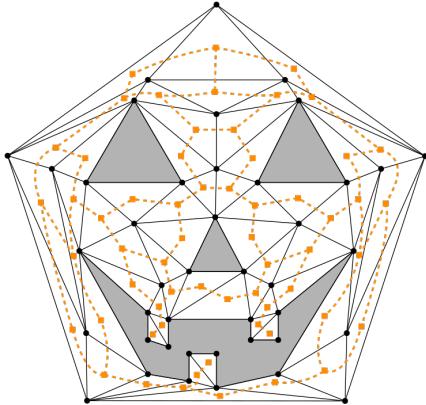
It is clear that, by running breadth-first search in linear time, one may decide whether or not an arbitrary graph is weakly dual to a (convex) polygonal triangulation.

Consider a planar embedding of a connected, simple graph. Suppose that every face therein is bounded by exactly three edges, except for  $k \geq 0$  faces  $h_1, \dots, h_k$  – called *holes* – and the infinite face  $f_\infty$ , each bounded by at least three edges. For ease of exposition, we distinguish the infinite face from the holes  $h_1, \dots, h_k$ , but it may be thought of as such, as its properties are the same as those for holes. Furthermore, suppose that each of the boundaries  $\partial f_\infty, \partial h_1, \dots, \partial h_k$  are pairwise-disjoint, simple, closed curves. Such an embedding is a *planar point-set triangulation with  $k$  holes*. In the weak dual of a planar point-set triangulation with holes, we omit the dual's vertices corresponding to  $h_1, \dots, h_k$  and to  $f_\infty$ . See Figure 2.

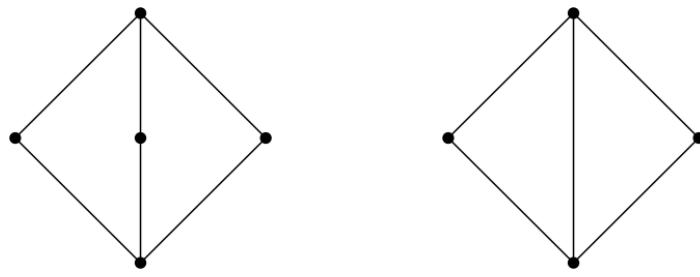
► **Lemma 2.** *There exists a planar point-set triangulation with holes to which a connected graph  $G$  is weakly dual if and only if, for every connected subgraph  $G' \subseteq G$ , there exists a planar point-set triangulation with holes to which  $G'$  is weakly dual.*

<sup>4</sup> The particular embedding  $\tilde{G}$  of  $G$  is important. Two embeddings of the same graph may have non-isomorphic dual graphs (or non-isomorphic weak dual graphs). See [4, 7, 11].

<sup>5</sup> For  $G$  with at least one vertex, we denote the maximum of its vertices' degrees by  $\Delta(G)$ , as in [11].



■ **Figure 2** a planar point-set triangulation with four holes (filled) and its weak dual (color, dashed).



■ **Figure 3** there does not exist a planar point-set triangulation to which either of these graphs are weakly dual.  $K_{2,3}$  (left) violates Theorem 4's condition (iii); the diamond (right) violates (iv).

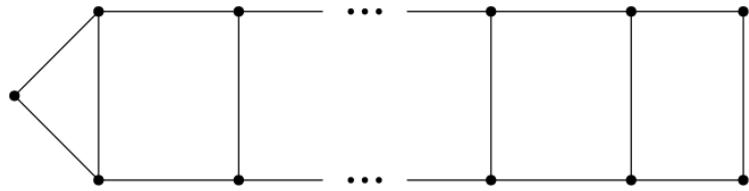
► **Lemma 3.** *Let  $G$  be a graph that is weakly dual to a planar point-set triangulation  $\mathcal{T}$  with  $k$  holes. Then, a vertex of degree  $i$  in  $G$  corresponds to a triangle in  $\mathcal{T}$  that shares exactly  $3 - i$  sides with either the infinite face's boundary or a hole's boundary. Moreover, if  $G$  has order at least four, then it must contain at least  $3(k + 1)$  vertices of degree less than three.*

Weakening the notion of outerplanarity, a graph  $G$  is said to be *degree  $k$ -or-less outerplanar* if it can be embedded in  $\mathbb{R}^2$  such that each of its vertices of degree  $k$  or less are incident to  $f_\infty$ . Dating back to at least Tutte [10], graph theorists have focused on understanding the set of all triangulations with varying definitions of "triangulation" and "all." As far as adjacency structure is concerned, we now characterize the set of all planar point-set triangulations.

► **Theorem 4.** *There exists a planar point-set triangulation to which a graph  $G$  is weakly dual if and only if (i)  $G$  is connected, (ii)  $\Delta(G) \leq 3$ , (iii)  $G$  is degree-two-or-less outerplanar, and (iv)  $G$  does not contain an induced subgraph of order at least four with fewer than three vertices of degree less than three.*

► **Corollary 5.** *The largest order- $n$  graph that is weakly dual to a planar point-set triangulation has  $n + \frac{n-3}{2}$  edges. As  $n \rightarrow \infty$ , this is half the size of the largest order- $n$  planar graphs.*

In the full version of this work, we have extended our analyses to triangulations with holes, quadrangulations, other polygon meshes in the plane, and tetrahedralizations. Among other follow-up goals, we strive to specify a linear-time recognition algorithm based on Theorem 4.



**Figure 4** the largest order- $n$  graph, referenced in Corollary 5, that satisfies Theorem 4's conditions.

---

## References

---

- 1 Imre Bárány and Günter Rote. Strictly convex drawings of planar graphs. *Documenta Mathematica*, 11:369–391, 2006.
- 2 Norishige Chiba, Tadashi Yamanouchi, and Takao Nishizeki. Linear algorithms for convex drawings of planar graphs. *Progress in Graph Theory*, 173:153–173, 1984.
- 3 Hubert De Fraysseix, János Pach, and Richard Pollack. How to draw a planar graph on a grid. *Combinatorica*, 10(1):41–51, 1990.
- 4 Reinhard Diestel. *Graph Theory*. Springer, 2017.
- 5 István Fáry. On straight-line representation of planar graphs. *Acta Scientiarum Mathematicarum*, 11(229-233):2, 1948.
- 6 Sandra L. Mitchell. Linear algorithms to recognize outerplanar and maximal outerplanar graphs. *Information Processing Letters*, 9(5), 1979.
- 7 U. S. R. Murty and Adrian Bondy. *Graph Theory*. Springer, 2008.
- 8 Günter Rote. Strictly convex drawings of planar graphs. In *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 728–734, 2005.
- 9 William T. Tutte. Convex representations of graphs. *Proceedings of the London Mathematical Society*, 3(1):304–320, 1960.
- 10 William T. Tutte. A census of planar triangulations. *Canadian Journal of Mathematics*, 14:21–38, 1962.
- 11 Douglas B. West. *Introduction to Graph Theory*. Prentice Hall, 2001.

# Efficient two-parameter persistence computation via cohomology

**Ulrich Bauer**

TU Munich, Germany

ulrich.bauer@tum.de

**Fabian Lenzen**

TU Munich, Germany

fabian.lenzen@tum.de

**Michael Lesnick**

SUNY Albany, US

mlesnick@albany.edu

---

## Abstract

In one-parameter persistent homology computation, clearing has proven an effective optimisation scheme. Our goal is to apply clearing also for two-parameter persistent homology. It involves the computation of homology via cohomology, which is not straightforward for two and more parameters since the cochain and cocycle modules are not generally free. We therefore develop a formula for a free resolution of two-parameter cohomology that allows for a clearing scheme. We show how matrices representing such a resolution also represent a free resolution of persistent homology.

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

**Keywords and phrases** Persistent homology, persistent cohomology, two-parameter persistence, clearing

**Digital Object Identifier** 10.4230/LIPIcs...

## 1 Introduction

Persistent homology [5, 10] is the study of the changes of homology  $H_\bullet(X_*)$  (with coefficients in a field  $k$ ) along a filtration  $\cdots \subseteq X_0 \subseteq X_1 \subseteq \cdots \subseteq X$  of a topological space. It is customarily assumed that  $X_*$  is a filtration by finite simplicial subcomplexes of a complex  $X = \bigcup_z X_z$ . The arising functor  $H_\bullet(X_*) : \mathbf{Z} \rightarrow \text{Vect}, z \mapsto H_\bullet(X_z)$  from the poset  $\mathbf{Z}$  to the category of  $k$ -vector spaces can equivalently be regarded as a finitely generated graded  $k[x]$ -module. The classification of the latter asserts that there is an essentially unique decomposition  $H_\bullet(X_*) \cong \bigoplus_{b \in B_f}(x^b) \oplus \bigoplus_{(b,d) \in B_i}(x^b)/(x^d)$  into indecomposable modules, indexed by two multi-sets  $B_f \subseteq \mathbf{Z}, B_i \subseteq \mathbf{Z}^2$  that form the barcode of  $H_\bullet(X_*)$ .

Computation of persistent homology can be performed by the *standard algorithm* [10], a Gaussian column reduction scheme of which efficient implementations abound [9, table 2]. Implementations featuring clearing are particularly efficient if the input is a Vietoris-Rips complex  $VR_*(Y)$  on a point cloud  $Y$  embedded in a metric space. It is a  $\mathbf{Z}$ -filtered complex with  $n$ -simplices  $VR_{r_z}^n = \{(y_0, \dots, y_n) \subseteq Y \mid \forall i, j: d(y_i, y_j) \leq r_z\}$  for a fixed sequence  $r \in \mathbf{R}^\mathbf{Z}$ .

Filtering  $X$  along two or  $n$  directions yields a system of subcomplexes  $X_n \subseteq X$  for every  $n \in \mathbf{Z}^n$  such that  $X_m \subseteq X_n$  for all  $m, n \in \mathbf{Z}^n$  with  $m \leq n$ . Simplicial chains  $C_\bullet(X_*)$  form a

---

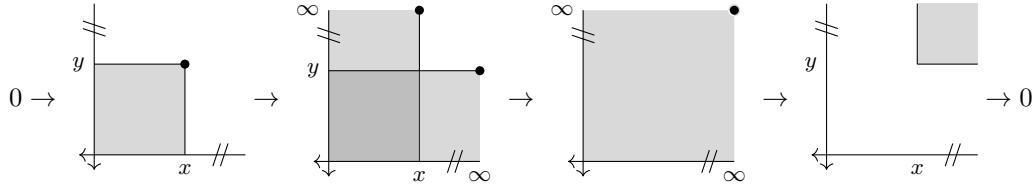
This is an abstract of a presentation given at CG:YRF 2021. It has been made public for the benefit of the community and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear in a conference with formal proceedings and/or in a journal.



© Ulrich Bauer, Fabian Lenzen, Michael Lesnick;  
licensed under Creative Commons License CC-BY



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



**Figure 1** Free resolution of the cochain module in  $\text{Vect}^{\mathbf{Z}^{2,\text{op}}}$  on the right. Structure maps of these modules go down and left; i.e., a submodule is a lower left corner and a quotient is an upper right corner. The black dots correspond to basis elements of the three free  $\text{Vect}^{\mathbf{Z}^{2,\text{op}}}$ -modules.

functor  $\mathbf{Z}^n \rightarrow \text{Vect}$ , which can equivalently be regarded as a graded  $k[x_1, \dots, x_n]$ -module. The category of the latter is of global dimension  $n$ , which complicates the computation of the respective  $H_\bullet(X_*)$  [2], and is known to be representation infinite [6], which rules out the possibility of an easy classification.

For  $n = 2$ , the module  $H_\bullet(X_*) \in \text{Vect}^{\mathbf{Z}^2}$  can be computed using the algorithm [8]. Kernels of maps of free modules are free; in particular the cycle module  $Z_\bullet(X_*)$  is. The algorithm computes and minimises a matrix representing the free presentation  $Z_\bullet(X_*) \rightarrow C_\bullet(X_*) \rightarrow H_\bullet(X_*) \rightarrow 0$ . This algorithm has been already subject to improvements [7]. However, optimisation schemes that have led to significant improvements in the one parameter case have yet be found analogues for in two parameters, notably *clearing*.

Clearing [1, 3] uses that a basis of  $Z_n(X_*)$  can be obtained by extending a vector space basis of the boundaries  $B_n(X)$ , instead computing it from scratch. This requires that  $B_n(X)$  be known before  $Z_n(X_*)$  is computed, which is only feasible for cohomology instead of homology. Since persistent cohomology and homology uniquely determine each other [4], this poses no restriction.

The applicability of a column reduction scheme for homology computation requires that the chain and cycle module  $C_\bullet(X_*)$ ,  $Z_\bullet(X_*) \in \text{Vect}^{\mathbf{Z}}$  be free. Applicability for cohomology computation requires the relative cochain and cocycle module  $C^\bullet(X, X_*)$ ,  $Z^\bullet(X, X_*) \in \text{Vect}^{\mathbf{Z}^\text{op}}$  be free as modules over the oppositely ordered poset  $\mathbf{Z}^\text{op}$ . For two and more parameters, neither  $C^\bullet(X_*)$  nor  $C^\bullet(X, X_*)$  is free in general though.

To be able to apply a clearing scheme for two parameters nevertheless, we replace  $C^\bullet(X_*) \in \text{Vect}^{\mathbf{Z}^{2,\text{op}}}$  by a free resolution of modules over the extended grid  $\bar{\mathbf{Z}}^{2,\text{op}}$  as in Figure 1. We compute a free resolution of  $H^\bullet(X_*)$ , employing the bi-graded kernel algorithm [8]. The conversion to an injective resolution of  $H^\bullet(X_*)$ , which then can be dualised to obtain a free resolution of  $H_\bullet(X_*) \in \text{Vect}^{\mathbf{Z}^2}$ , turns out not to involve any computation at all.

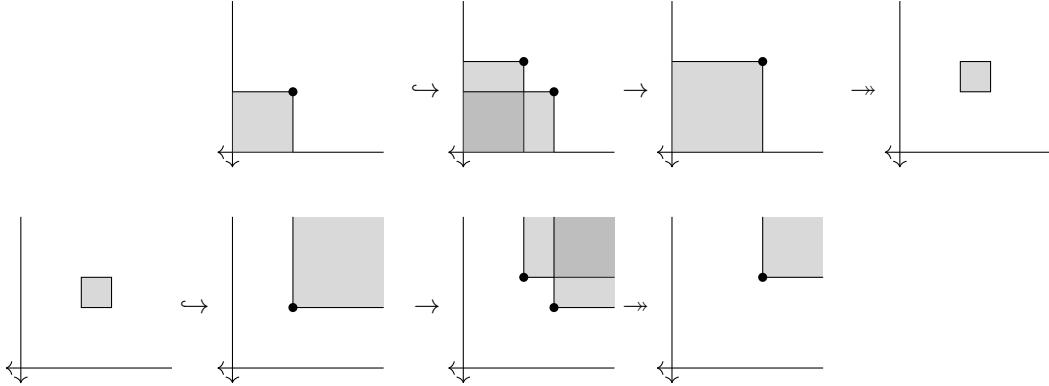
## 2 Results

We assume that  $X_*$  is one-critically filtered; i.e., every  $\sigma \in X$  has a unique lowest filtration value  $\deg z \in \mathbf{Z}^2$  at which it enters the filtration. A pullback  $E \times_F G$  of free two-parameter persistence modules is free since it can be written as a kernel of a map of free modules.

► **Theorem 1.** *Let  $L \xrightarrow{f} M \xrightarrow{g} N$  be maps of modules in  $\text{Vect}^{\bar{\mathbf{Z}}^{2,\text{op}}}$  with  $gf = 0$ . Let  $f_\bullet: L_\bullet \rightarrow M_\bullet$  and  $g_\bullet: M_\bullet \rightarrow N_\bullet$  be lifts of these maps to free resolutions of the respective modules. Then there are maps rendering*

$$0 \longrightarrow L_0 \times_{M_0} M_1 \longrightarrow L_0 \oplus M_1 \oplus N_2 \longrightarrow M_0 \times_{N_0} N_1 \longrightarrow \ker g / \text{im } f \longrightarrow 0$$

*a free resolution of  $\ker g / \text{im } f$ .*



**Figure 2** The generator degrees in a minimal free resolution (top row) determine the degrees in an injective resolution (bottom row) of the rectangle-supported module.

We have omitted the maps in the pullbacks in order to preserve readability. We make explicit how we choose free resolutions of the cochain modules  $C^i(X_*)$  and lifts of the coboundary operator; see Figure 1 for the idea. Applying Theorem 1 to  $C^{n-1}(X_*)$ ,  $C^n(X_*)$ ,  $C^{n+1}(X_*)$  gives a resolution  $H_\bullet^n$  of  $H^n(X_*)$ . The maps occurring in Theorem 1 are particularly easy to write down.

For a fixed basis of the resolution  $H_\bullet^n$ , we assign to each rank-one summand  $G$  of  $H_\bullet^n$  an injective module  $\tilde{G}$ . The procedure extends to an assignment  $F_i^n \mapsto \tilde{F}_\bullet^n$  of a based injective module to each free module in the resolution as sketched in Figure 2.

► **Theorem 2.** *If  $0 \rightarrow F_2 \xrightarrow{f_2} F_1 \xrightarrow{f_1} F_0 \rightarrow M \rightarrow 0$  is a free resolution and  $A_2, A_1$  respectively represent  $f_2, f_1$  w. r. t. chosen bases of  $F_\bullet$ , then the same matrices represent an injective resolution  $M \rightarrow \tilde{F}_2 \rightarrow \tilde{F}_1 \rightarrow \tilde{F}_0 \rightarrow 0$  w. r. t. certain bases. If one resolution is minimal, then so is the other.*

From the free resolution  $H_\bullet^n$  of  $H^n(X_*)$  we obtain an injective resolution  $H^n(X_*) \hookrightarrow \tilde{H}_\bullet^n$  and thus a free resolution  $(H_\bullet^n)^* \twoheadrightarrow H_n(X_*)$ , obtained by the degree-wise dualisation  $(-): M \mapsto \text{Hom}_k(M, k)$ .

---

## References

---

- 1 Ulrich Bauer, Michael Kerber, and Jan Reininghaus. Clear and Compress: Computing Persistent Homology in Chunks. In Peer-Timo Bremer, Ingrid Hotz, Valerio Pascucci, and Ronald Peikert, editors, *Topological Methods in Data Analysis and Visualization III*, pages 103–117. Springer International Publishing, Cham, 2014. doi:10.1007/978-3-319-04099-8\_7.
- 2 Mickaël Buchet and Emerson G. Escolar. Realizations of indecomposable persistence modules of arbitrarily large dimension. In Bettina Speckmann and Csaba D. Tóth, editors, *34th International Symposium on Computational Geometry (SoCG 2018)*, volume 99 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 15:1–15:13, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. doi:10.4230/LIPIcs.SoCG.2018.15.
- 3 Chao Chen and Michael Kerber. Persistent Homology Computation with a Twist. In *27th European Workshop on Computational Geometry*, 2011. URL: <https://eurocg11.inf.ethz.ch/abstracts/22.pdf>.
- 4 Vin de Silva, Dmitriy Morozov, and Mikael Vejdemo-Johansson. Dualities in persistent (co)homology. *Inverse Problems*, 27(12):124003, December 2011. doi:10.1088/0266-5611/27/12/124003.

## Efficient two-parameter persistence computation via cohomology

- 5 Edelsbrunner, Letscher, and Zomorodian. Topological Persistence and Simplification. *Discrete & Computational Geometry*, 28(4):511–533, November 2002. doi:10.1007/s00454-002-2885-2.
- 6 I. M. Gel'fand and V. A. Ponomarev. Remarks on the classification of a pair of commuting linear transformations in a finite-dimensional space. *Functional Analysis and Its Applications*, 3(4):325–326, October 1969. doi:10.1007/BF01076321.
- 7 Michael Kerber and Alexander Rolle. Fast Minimal Presentations of Bi-graded Persistence Modules, October 2020. arXiv:2010.15623.
- 8 Michael Lesnick and Matthew Wright. Computing Minimal Presentations and Bigraded Betti Numbers of 2-Parameter Persistent Homology, March 2019. arXiv:1902.05708.
- 9 Nina Otter, Mason A Porter, Ulrike Tillmann, Peter Grindrod, and Heather A Harrington. A roadmap for the computation of persistent homology. *EPJ Data Science*, 6(1):17, December 2017. doi:10.1140/epjds/s13688-017-0109-5.
- 10 Afra Zomorodian and Gunnar Carlsson. Computing persistent homology. *Discrete & Computational Geometry*, 33:249–274, February 2005. doi:10.1007/s00454-004-1146-y.

# Moving Robots One by One is Hard

Tzvika Geft

Blavatnik School of Computer Science, Tel-Aviv University, Israel

Dan Halperin

Blavatnik School of Computer Science, Tel-Aviv University, Israel

---

## Abstract

---

In multi-robot motion planning (MRMP) the aim is to plan the motion of several robots operating in a common workspace, while avoiding collisions with obstacles or with fellow robots. The problem is known to be hard in various settings. We show that a restricted and natural version of the problem, where each robot is allowed to move only once, which we call *monotone* MRMP, is NP-complete.

**2012 ACM Subject Classification** Computing methodologies → Multi-agent planning

**Keywords and phrases** multi-robot motion planning, multi-agent pathfinding, pebble motion on graphs

**Related Version** A full version of the paper is available at <https://arxiv.org/abs/2104.07011>.

**Funding** This work has been supported in part by the Israel Science Foundation (grant no. 1736/19), by NSF/US-Israel-BSF (grant no. 2019754) and by the Israel Ministry of Science and Technology (grant no. 103129).

## 1 Introduction

In multi-robot motion planning (MRMP) the aim is to plan the motion of several robots operating in a common workspace, while avoiding collisions with obstacles or with each other. We consider the *labeled* case, where each robot moves from its start to its *own* target position. MRMP in two-dimensional continuous domains has been shown to be hard in various settings [1, 4, 5, 7, 8]. In all the MRMP hardness results we are aware of the robots are allowed to make multiple moves. For example, one robot  $A$  performs a single *move* consisting of starting and then stopping along  $A$ 's path, then another robot moves, and then robot  $A$  resumes its motion. Moreover, the number of moves made by robots in some hardness constructions is exponential in the number of robots [4, 7]. We are therefore motivated to consider the computational complexity of MRMP when the motion plans are simpler.

We study *monotone* MRMP, where we restrict robots to move only once, i.e., robots must move one by one to their targets with no intermediate stops. We show that monotone MRMP remains hard, namely NP-complete. Our hardness construction uses a rectangular workspace with unit-square robots and obstacles. The construction applies to both continuous and discrete domains, such as robots operating on a grid with holes. The applicability of the hardness result to discrete domains stands in contrast to unrestricted (non-monotone) graph-based MRMP, for which feasibility is decidable in polynomial time [6, 10, 11]. Our result also establishes the hardness of optimal decoupling of multi-robot motion [9]. In the full version we relate our result to decoupling, which is a standard approach to practically addressing MRMP, and to *reconfiguration* problems [2, 3] that are analogous to MRMP.

---

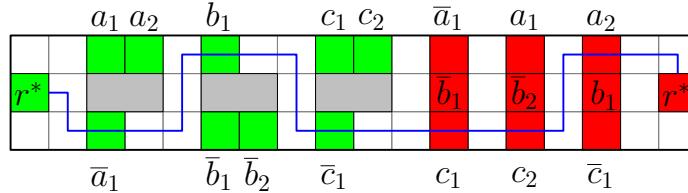
This is an abstract of a presentation given at CG:YRF 2021. It has been made public for the benefit of the community and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear in a conference with formal proceedings and/or in a journal.

## Moving Robots One by One is Hard

### 2 Result

We prove the NP-completeness of the following decision problem. We are given a set  $R$  of  $n$  unit square robots in a planar workspace, where each  $r \in R$  has a start and target position, denoted by  $s_r$  and  $t_r$ , respectively. The goal is to decide whether there is a *monotone motion plan*, i.e., a sequence of moves in which each robot  $r \in R$  moves once from  $s_r$  to  $t_r$  without inducing collisions, while other robots are stationary. If such a motion plan exists we say that the given instance is *feasible*.

To prove hardness we use a reduction from 3SAT, the problem of deciding satisfiability of a formula in conjunctive normal form with 3 literals in each clause. Given a 3SAT formula  $\phi$ , we construct a corresponding monotone MRMP instance  $M$  that is feasible if and only if  $\phi$  has a satisfying assignment. The workspace in which the robots can move is a three-row high rectangular portion of the unit grid in which some of the cells are immovable obstacles; see Figure 1. All the start and target positions are cells on the grid. There is one robot for each literal in  $\phi$ , which together are called *literal robots*, and one *assignment verifier* robot  $r^*$ . The literal robots' start positions are located in *assignment gadgets*. Each assignment gadget initially contains robots representing literals of a single variable of  $\phi$ . The top and bottom rows of the gadget contain robots representing only positive and negative literals, respectively, while the middle row is composed of obstacle cells. The literal robots' target positions are located in *clause gadgets*, each of which is a column of three cells, one for each literal in the clause. All the assignment gadgets are located to the left of the clause gadgets (the order of gadgets of the same type is arbitrary). To the left of every gadget is a column of empty cells to allow robots to enter the gadget at any of the three workspace rows. After all the gadgets are placed, for each assignment gadget we determine the order of the sources within each of its two non-obstacle rows. In each such row, the left to right order of the start positions is set to match the left to right order of the corresponding targets, which we refer to as the *intra-literal order property*. Finally,  $r^*$  has to move across the workspace, from the leftmost to the rightmost column.



**Figure 1** The monotone MRMP instance for the formula  $(\bar{a} \vee \bar{b} \vee c) \wedge (a \vee \bar{b} \vee c) \wedge (a \vee b \vee \bar{c})$ . The start and target positions are colored in green and red, respectively, and are labeled with their respective literals. Literal labels have unique indices in order to distinguish between appearances of the same literal. The path  $P$  (blue) is shown for the assignment  $a = T, b = F, c = T$ , for which the corresponding motion plan has robots moving in the following order:  $\bar{c}_1, b_1, \bar{a}_1, r^*, a_2, a_1, \bar{b}_2, c_2, \bar{b}_1, c_1$ .

We provide one direction of the full reduction proof. The other direction is similar and can be found in the full version.

► **Theorem 1.** *Let  $\phi$  be a 3SAT formula with a satisfying assignment  $\mathcal{A}$ . Then the corresponding MRMP instance  $M$  is feasible.*

**Proof.** Let  $R^+$  (resp.  $R^-$ ) be the set of robots corresponding to literals that evaluate to true (resp. false) according to  $\mathcal{A}$ . That is, for each assignment gadget,  $R^+$  contains robots that are all initially either in the top or the bottom row of the gadget, according to  $\mathcal{A}$ . We

show that the robots can move in the rough order  $R^-$ ,  $r^*$ ,  $R^+$ , which is made more precise below. Let  $P$  be a weakly  $x$ -monotone path from  $s_{r^*}$  to  $t_{r^*}$  that passes through the sources of  $R^-$  and targets of  $R^+$  (without passing through obstacles or other sources or targets); see Figure 1.  $P$  exists because each clause gadget must contain a target of some robot in  $R^+$ , or else  $\mathcal{A}$  does not satisfy  $\phi$ . In the motion plan, each  $r \in R^-$  follows the subpath of  $P$  from  $s_r$  (through which  $P$  passes) up to the empty column before the gadget containing  $t_r$ , from which  $r$  easily reaches  $t_r$ . The order in which the robots in  $R^-$  move is the right to left order of their sources, which guarantees no collisions with another robot located at its source. Since the robots in  $R^-$  move before  $R^+$ , the targets through which  $P$  passes are unoccupied when the robots in  $R^-$  move, guaranteeing no collisions in the clause gadgets. Next,  $r^*$  moves using  $P$ , which consists of empty cells at this point. Finally, each  $r \in R^+$  connects to  $P$  at the nearest empty column to its right, from which point it continues similarly to  $R^-$ . The order of motion of the robots in  $R^+$  is the right to left order of their targets, which guarantees no collisions in the clause gadgets. Note that due to the intra-literal order property we also have no collisions among  $R^+$  within assignment gadgets.  $\blacktriangleleft$

---

## References

---

- 1 Thomas Brocken, G. Wessel van der Heijden, Irina Kostitsyna, Lloyd E. Lo-Wong, and Remco J. A. Surtel. Multi-robot motion planning of k-colored discs is PSPACE-hard. In *FUN*, volume 157 of *LIPICS*, pages 15:1–15:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.
- 2 Gruia Călinescu, Adrian Dumitrescu, and János Pach. Reconfigurations in graphs and grids. *SIAM J. Discret. Math.*, 22(1):124–138, 2008.
- 3 Adrian Dumitrescu and Minghui Jiang. On reconfiguration of disks in the plane and related problems. *Comput. Geom.*, 46(3):191–202, 2013.
- 4 Robert A. Hearn and Erik D. Demaine. PSPACE-completeness of sliding-block puzzles and other problems through the nondeterministic constraint logic model of computation. *Theor. Comput. Sci.*, 343(1-2):72–96, 2005.
- 5 John E. Hopcroft, Jacob Theodore Schwartz, and Micha Sharir. On the complexity of motion planning for multiple independent objects; PSPACE-hardness of the "warehouseman's problem". *The International Journal of Robotics Research*, 3(4):76–88, 1984.
- 6 Daniel Kornhauser, Gary L. Miller, and Paul G. Spirakis. Coordinating pebble motion on graphs, the diameter of permutation groups, and applications. In *FOCS*, pages 241–250. IEEE Computer Society, 1984.
- 7 Kiril Solovey and Dan Halperin. On the hardness of unlabeled multi-robot motion planning. *Int. J. Robotics Res.*, 35(14):1750–1759, 2016.
- 8 Paul G. Spirakis and Chee-Keng Yap. Strong NP-hardness of moving many discs. *Inf. Process. Lett.*, 19(1):55–59, 1984.
- 9 Jur van den Berg, Jack Snoeyink, Ming C. Lin, and Dinesh Manocha. Centralized path planning for multiple robots: Optimal decoupling into sequential plans. In *Robotics: Science and Systems*. The MIT Press, 2009.
- 10 Jingjin Yu. A linear time algorithm for the feasibility of pebble motion on graphs. *CoRR*, abs/1301.2342, 2013.
- 11 Jingjin Yu and Daniela Rus. Pebble motion on graphs with rotations: Efficient feasibility tests and planning algorithms. In *Workshop on the Algorithmic Foundations of Robotics, WAFR*, pages 729–746, 2014.

# On the range of two-distance graphs

Péter Ágoston 

Eötvös Loránd University, Budapest, Hungary and MTA-ELTE Lendület Combinatorial Geometry (CoGe) Research Group

---

## Abstract

---

The topic of this paper is related to the well-known notion of unit distance graphs. Take a graph with its edges coloured red and blue such that for some  $d$  it can be mapped into the plane with all vertices going to distinct points, the red edges to segments of length 1 and the blue edges to those of length  $d$ . We define the range of this graph to be the set of such numbers  $d$ . It is easy to show that the range of any edge-bicoloured graph is semialgebraic, and we now prove that any semialgebraic set with a positive upper and lower bound is the range of a suitable graph.

**2012 ACM Subject Classification** Theory of computation→Randomness, geometry and discrete structures→Computational geometry, Mathematics of computing→Discrete mathematics→Graph theory

**Keywords and phrases** distance graph, combinatorial geometry, semialgebraic set

## 1 Introduction

**Definition 1.1.** We call a graph a unit distance graph (UDG), if it can be drawn to  $\mathbb{R}^2$  so that all vertices go to distinct points and all neighbouring pairs of vertices have Euclidean distance 1. We call such a drawing a unit distance representation (UDR) of the graph.

From now on, we suppose that all graphs are finite and simple unless stated otherwise.

**Definition 1.2.** Call a graph an edge-bicoloured graph (EBG), if there is a fixed colouring of its edges with two colours.

From now on, we will suppose that these colours are red and blue.

**Definition 1.3.** Call an EBG  $G$  a  $(1, d)$ -graph for some  $d \in \mathbb{R}_{\geq 0}$ , if the vertices of the graph can be represented in the plane by distinct points so that those connected with a red edge go to points with distance 1 and those connected with a blue edge go to points with distance  $d$ . Call such a representation a  $(1, d)$ -representation of  $G$ .

**Definition 1.4.** For an EBG  $G$ , define its range  $\text{ran}(G)$  as the set of numbers for which  $G$  is a  $(1, d)$ -graph. Let the range of a graph be the union of the ranges of its edge-bicolourings.

We call a graph with or without an edge-bicolouring a two-distance graph, if its range is not empty. Two-distance graphs have been studied in the past in several papers. [3][4]

When speaking about a  $(1, d)$ -representation of a graph, we often do not differentiate between vertices, edges and their images.

**Lemma 1.5.** *For EBGs  $H \subseteq G$  (the colouring is inherited),  $\text{ran}(G) \subseteq \text{ran}(H)$ .*

**Lemma 1.6.** *For EBGs  $G_1$  and  $G_2$ ,  $\text{ran}(G_1 \dot{\cup} G_2) = \text{ran}(G_1) \cap \text{ran}(G_2)$ .*

Deciding whether a number  $d$  is in the range of an EBG or not is  $\mathbb{R}$ -complete, since deciding whether a graph is a UDG or not is  $\mathbb{R}$ -complete. [5]

$\chi(\mathbb{R}^2)$  denotes the minimal number of colours needed to colour  $\mathbb{R}^2$  without a monochromatic pair of distance 1. Finding  $\chi(\mathbb{R}^2)$  is a famous problem [2] and Bukh conjectured [1] that by also forbidding a transcendental distance, we get the same number. If true, this could make it interesting to find graphs whose range only contains a transcendental number.



© Péter Ágoston;  
licensed under Creative Commons License CC-BY 4.0

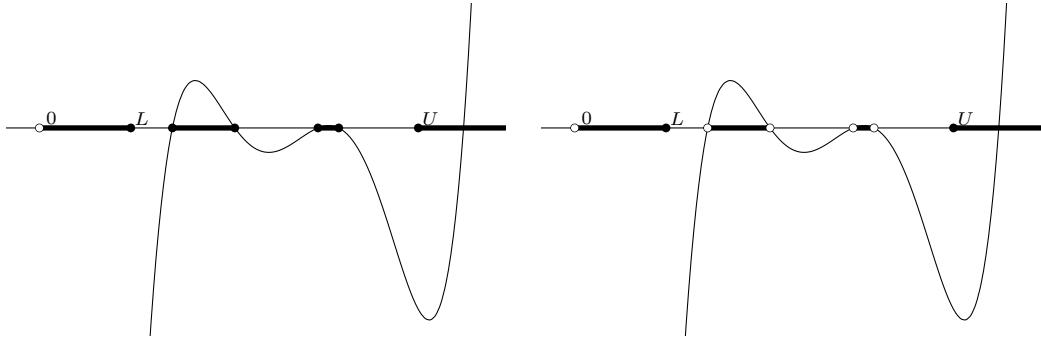
Leibniz International Proceedings in Informatics



LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

This is an abstract of a presentation given at CG:YRF 2021. 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.

## On the range of two-distance graphs



**Figure 1** A polynomial  $p(x)$  with  $S_0(p, L, U)$  (left) and  $S_1(p, L, U)$  (right) denoted by bold

**Definition 1.7.** Take the set of solutions  $(x_1, \dots, x_d)$  to a finite sequence of polynomial equations and inequalities of the form  $p(x_1, \dots, x_d) = 0$  and  $p(x_1, \dots, x_d) > 0$ . If a set can be generated as the union of such sets, it is called a semialgebraic set.  $S \subseteq \mathbb{R}$  is semialgebraic exactly if it can be generated as the union of finitely many intervals with algebraic endpoints.

**Proposition 1.8.** *The range of an EBG  $G$  is always a semialgebraic set.*

Our main result says this condition is tight if  $\text{ran}(G)$  has positive lower and upper bounds:

**Theorem 1.9.** *For a set  $S \subseteq \mathbb{R}_{>0}$  with a positive lower and upper bound ( $\lambda$  and  $v$ ), there exists an EBG  $G$  with  $\text{ran}(G) = S$  if and only if  $S$  is semialgebraic.*

## 2 Preliminary algebraic statements

**Definition 2.1.** Call a polynomial **even**, if all of its coefficients with odd index are 0. In other words, a polynomial  $p$  is even, if it is an even function ( $p(x) = p(-x)$ ).

**Definition 2.2.** Take a polynomial  $p$ , and  $L \leq U$ ,  $\{L, U\} \subset \mathbb{R}_{\geq 0} \cup \{+\infty\}$ . Define:

$$S_0(p, L, U) = \{x \in \mathbb{R}_{>0} \mid (p(x) \geq 0) \vee (x \leq L) \vee (x \geq U)\} \quad \text{and}$$

$$S_1(p, L, U) = \{x \in \mathbb{R}_{>0} \mid (p(x) > 0) \vee (x \leq L) \vee (x \geq U)\} \quad (\text{Figure 1}).$$

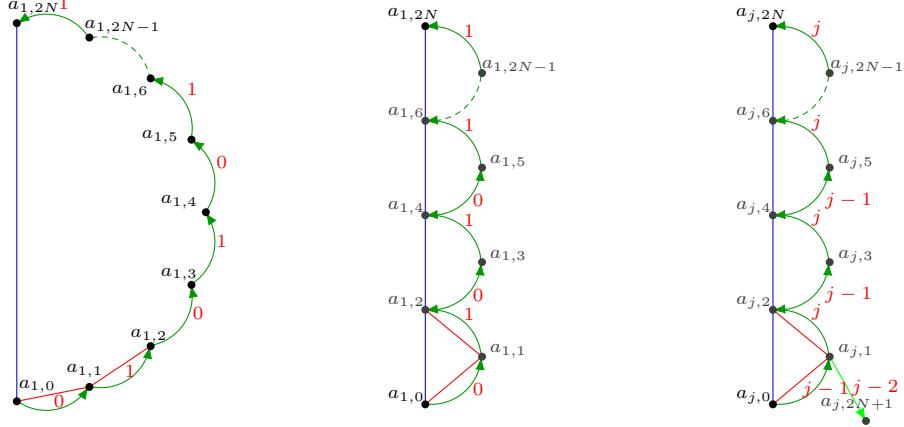
**Proposition 2.3.** *Take a semialgebraic set  $\sigma \subseteq [\lambda, v]$ . For some  $n \in \mathbb{N}$  there exist even polynomials  $p_1, \dots, p_{n+1}$  with integer coefficients and a negative leading coefficient, numbers  $L_1, \dots, L_n, U_1, \dots, U_n \in \mathbb{Q}_{>0}$  and numbers  $\zeta_1, \dots, \zeta_{n+1} \in \{0, 1\}$  so that*  

$$\sigma = \left( \bigcap_{i=1}^n S_{\zeta_i}(p_i, L_i, U_i, i) \right) \cap S_{\zeta_{n+1}}(p_{n+1}, 0, +\infty).$$

## 3 Proof of Theorem 1.9

**Proposition 3.1.** *For any even polynomial  $p \in \mathbb{Z}[x]$  with integer coefficients and a negative leading coefficient, there exists an EBG  $G(p)$ , whose range is  $S_0(p, 0, +\infty)$ .*

*Sketch of the proof:* We define partly virtual EBGs (PVEBG), in which we also allow directed green edges, divided into groups. In a  $(1, d)$ -representation of a PVEBG, we require green edges from the same group to have the same vector, besides the criteria for EBGs and



**Figure 2** The components of  $A$ :  $A_1$  (left), the only  $(1, d)$ -representations (up to isometry) of  $A_1$  (middle) and of  $A_j$  ( $2 \leq j \leq \deg(p)$ ) (right) ( $N$  is large enough and groups are denoted by numbers).

we define its range analogously to EBGs. In case some boundedness conditions apply, an EBG with the same range can be created by connecting green edges by red grids.

The most crucial component of creating  $G(p)$  is graph  $A$  (Figure 2).

For small enough  $d$ ,  $A$  has exactly one  $(1, d)$ -representation up to transformations which are isometries on the components. If we draw the complex plane so that  $\overrightarrow{a_{1,0}a_{1,1}} = 1$  and  $\overrightarrow{a_{1,1}a_{1,2}} = \varepsilon$ , the members of the group marked by  $j$  will have vector  $\varepsilon^j$  and  $|N \cdot (1 + \varepsilon)| = d$ . This helps constructing points having distance of some even polynomial of  $d$ , and ultimately, constructing  $G(p)$ .

**Proposition 3.2.** *For any even polynomial  $p \in \mathbb{Z}[x]$  with a negative leading coefficient, there exists an EBG  $G'(p)$ , whose range is  $S_1(p, 0, +\infty)$ .*

**Proposition 3.3.** *For an EBG  $G$ , positive rational numbers  $L_a, U_a$  and arbitrary real numbers  $L_b, U_b$  ( $L_b < L_a < U_a < U_b$ ), if  $\text{ran}(G) \cap (L, U] \neq \emptyset$ , then there exists an EBG  $G_{L_a, L_b}^{U_a, U_b}$  for which  $\text{ran}(G_{L_a, L_b}^{U_a, U_b}) \cap (L_b, U_b) = ((0, L] \cup \text{ran}(G) \cup [U, +\infty)) \cap (L_b, U_b)$ .*

Using the notations of Proposition 2.3, with the help of Proposition 3.3, we can construct  $(1, d)$ -graphs  $G(p_i)_{\lambda, L_i}^{v, U_i}$  for  $1 \leq i \leq n$  and  $\zeta_i = 0$ , while in case of  $\zeta_i = 1$ , we construct  $G'(p_i)_{\lambda, L_i}^{v, U_i}$ , whose range coincides with  $S_{\zeta_i}(p_i, L_i, U_i)$  on the interval  $[\lambda, v]$ . And finally, we can take  $G'(p_{n+1})$ , whose range is empty outside of  $[\lambda, v]$ , thus the intersection of the ranges of these graphs is  $\sigma$  because of Proposition 2.3. Thus, because of Lemma 1.6, their disjoint union has  $\sigma$  as its range. So all semialgebraic sets from  $[\lambda, v]$  are the range of some EBG. ◀

I thank D. Pálvolgyi for the problem and M. Laczkovich for proving Proposition 1.8.

---

## References

- 1 B. BUKH: Measurable Sets With Excluded Distances, *GAFA Geom. funct. anal.* **18**, 668–697 (2008)
- 2 A. D. N. J. DE GREY: The chromatic number of the plane is at least 5, arXiv: 1804.02385 (2018)

## On the range of two-distance graphs

- 3 G. EXOO, D. ISMAILESCU: A 6-chromatic two-distance graph in the plane, arXiv:1909.13177 (2019)
- 4 J. PARTS: A small 6-chromatic two-distance graph in the plane, arXiv:2010.12656 (2020)
- 5 M. SCHAEFER: Realizability of Graphs and Linkages, *Thirty Essays on Geometric Graph Theory* (2013), 461–482.