

# CGWeek Young Researchers Forum 2020

## Booklet of Abstracts

2020

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 23-26, 2020.

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

- Michael Kerber (chair), TU Graz
- Erin Chambers, Saint Louis University
- Anne Driemel, University of Bonn
- Tamal Dey, Ohio State University
- Wouter Meulemans, TU Eindhoven
- Evanthia Papadopoulou, Università della Svizzera italiana
- Zuzana Patakova, IST Austria
- Sharath Raghvendra, Virginia Tech

There were 22 papers submitted to CG:YRF. Of these, 20 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/>

# Robust Boolean operations on polygons

**Shengtan Mao**

Department of Computer Science, University of North Carolina at Chapel Hill, United States  
maoshengtan2011@gmail.com

**Jack Snoeyink**

Department of Computer Science, University of North Carolina at Chapel Hill, United States  
snoeyink@cs.unc.edu

---

## Abstract

We detail an algorithm for Boolean operations on two polygonal regions. The algorithm is optimal and requires only double the input precision. It is designed so that degeneracies (shared endpoints, overlapping segments) typically requiring special handling are treated as general cases.

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

**Keywords and phrases** Boolean operations, polygons, robustness, segment intersections

## 1 Introduction

The Boolean operations algorithm takes two polygonal regions  $R, S$  as two groups of oriented segments and outputs the resulting region of  $R, S$  under a specified Boolean operation (union, intersection, difference). The orientation determines the inside/outside: the left region of an oriented segment is considered the inside.

The Boolean operations algorithm consists of two smaller algorithms: the segment intersection algorithm and the construction algorithm. The first algorithm finds the segment intersections between the two groups, and the second builds the representation of the resulting region. The segment intersection algorithm is adapted from [1]; it runs in  $O(n \log n + k)$  time, where  $n$  is the total number of oriented segments, and  $k$  is the number of segment intersections. The construction algorithm runs in  $O(n + k)$  time. Both of the smaller algorithms are optimal and require only double the input precision; this allows the Boolean operations algorithm to be optimal and to require only double the input precision. The segment intersection points do not have coordinates since calculating them requires more than double the input precision (they are instead represented by pairs of intersecting segments). So calculating and rounding the coordinates are left to the user and can be made application-specific.

### Output format

The construction algorithm outputs the resulting region as a *nesting tree*. Each node of the nesting tree contains one oriented simple polygon (OSP), and for any pair of parent and child, there does not exist another OSP nested inside the parent that nests the child. The nesting tree can represent nesting, which is essential for representing the resulting region. The construction algorithm with the nesting tree is the main contribution of this paper.

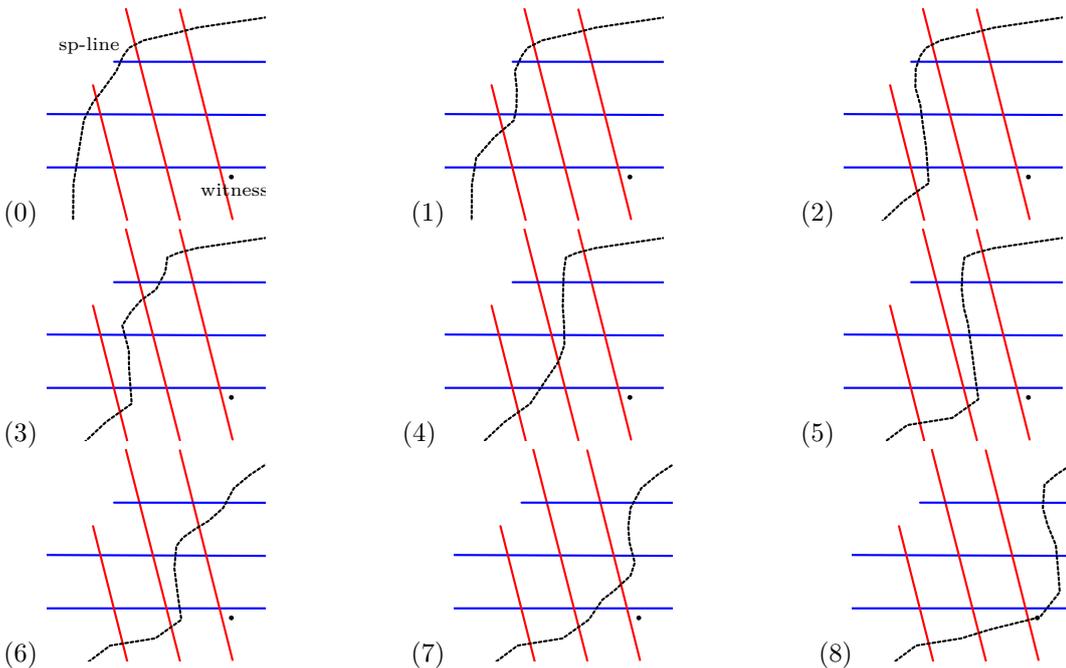
## 2 Segment intersection algorithm

Reference [1] outlines a sweep algorithm that takes two groups of segments and outputs the segment intersections between them. This algorithm can be used for oriented segments.

We adopt the notions of *flags*, *flag order*, and *witness* developed by [1]. A flag is an endpoint of a (generic) segment along with that segment, so we can differentiate between shared endpoints of different segments. The sweep-order for flags is called the flag order. A flag is a witness for a segment intersection if it is the first flag that detects the intersection.

### Sweep-order for flags and segment intersections

The algorithm in [1] outputs segment intersections in batches, but they need to be ordered for the construction algorithm. We modified the algorithm in [1] to achieve this goal. This approach was inspired by, but does not directly utilize, [2]. We describe the sweep-order using a *sweep-pseudo-line* (sp-line). It is a curve that intersects each oriented segment at most once. The sp-line steps through a flag or an intersection on each update. The sp-line steps through flags in flag order. If a flag is a witness, the sp-line steps through the witnessed intersections in column-major order on the partial “grid” formed by the intersections before stepping through the witness (see Figure 1).

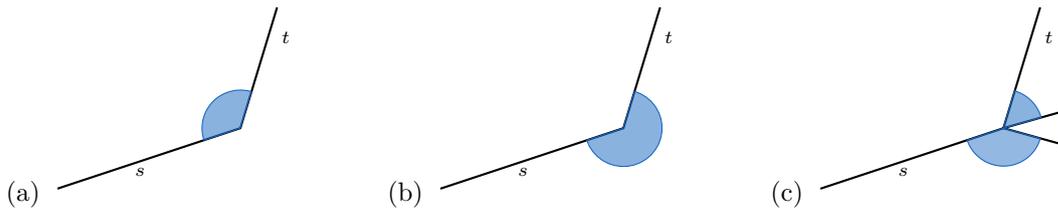


■ **Figure 1** These 9 steps show how the sp-line steps through the 8 intersections

### 3 Boundary segments, event pairs, boundary components

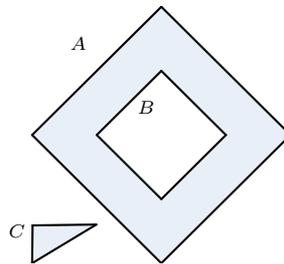
Some oriented segments (or parts of oriented segments) may not bound the inside of the resulting region. From the results of the segment intersection algorithm, we want to extract a single group of oriented segments such that every part of every oriented segment bounds the inside of the resulting region. We call these modified oriented segments *boundary segments*. Note that these modifications involve splitting oriented segments on the intersections, so the endpoints of a boundary segment may not have explicit coordinates, which may cause complications for common algorithms and data structures.

The sweep-order for the construction algorithm orders boundary segment flags in pairs called *event pairs*. An event pair is two boundary segment flags that bound an inside angular region without “gaps” (i.e. an event pair forms the corner of a polygon in the resulting region) (see Figure 2).



■ **Figure 2** The shaded angles indicate the inside angular region. The flags of  $s$  and  $t$  at the vertex form an event pair in (a) and (b) but not in (c) because of the “gap”.

We use *boundary component* to organize the construction of an OSP during the sweep. It consists of the OSP being constructed and two lists of completed OSP. One list consists of completed OSPs directly below and inside (*below-inside*) the OSP being constructed. The other list is for OSPs directly below and outside (*below-outside*) (see Figure 3).



■ **Figure 3** The shaded region is inside.  $B$  is below-inside  $A$ .  $C$  is below-outside  $A$ .

#### 4 Boolean operations algorithm

We describe the overall algorithm using the sp-line and a *boundary-sweep-pseudo-line* (bsp-line). The bsp-line is a curve that intersects each boundary segment at most once. These two pseudo-lines step in an alternating fashion. The sp-line extracts the boundary segments and updates the bsp-line with them. The bsp-line steps when it detects an event pair and updates the boundary component of the OSPs currently intersecting the bsp-line. When a boundary component is completed, we update the *below-inside-tree* and *below-outside-tree*. Each node of either tree contains one OSP, and for any pair of parent and child, the child is below-inside the parent for the below-inside-tree, and the child is below-outside the parent for the below-outside-tree.

We can construct the nesting tree from the below-inside-tree and below-outside-tree. An OSP’s *neutral orientation* is the orientation such that the direction through the rightmost point is counter-clockwise. Given a completed boundary component with OSP  $P$ , if  $P$ ’s orientation (determined by the boundary segments forming  $P$ ) agrees with its neutral orientation, every OSP in the subtree of  $P$  in the below-inside-tree is a child of  $P$  in the nesting tree. If its orientation does not agree, every OSP in the subtree of  $P$  in the below-outside-tree is a child of  $P$  in the nesting tree.

---

#### References

- 1 Mantler A. and Snoeyink J. Intersecting Red and Blue Line Segments in Optimal Time and Precision. *Discrete and Computational Geometry*, 2001. doi:10.1007/3-540-47738-1\_23.
- 2 Iacono J., Khramtcova E., and Langerman S. Searching edges in the overlap of two plane graphs. *Algorithms and Data Structures*, 2017. doi:10.1007/978-3-319-62127-2\_40.

# Voronoi-based similarity distances between arbitrary crystal lattices

Marco Michele Mosca, Dr. Vitaliy Kurlin

Materials Innovation Factory  
University of Liverpool, United Kingdom

## Abstract

This paper introduces new distances based on Voronoi cells of crystal lattices to compare crystal structures. The Crystal Structure Prediction aims to computationally predict most stable crystals with desired properties. Many simulated crystals approximate the same local minimum, hence can be nearly identical. To save resources on further simulations and analysis, datasets of simulated crystals should be filtered by removing very similar structures. The proposed distances between crystal lattices are invariant under rigid motions and can be used for visualizing crystal datasets.

**2012 ACM Subject Classification** Computational Geometry

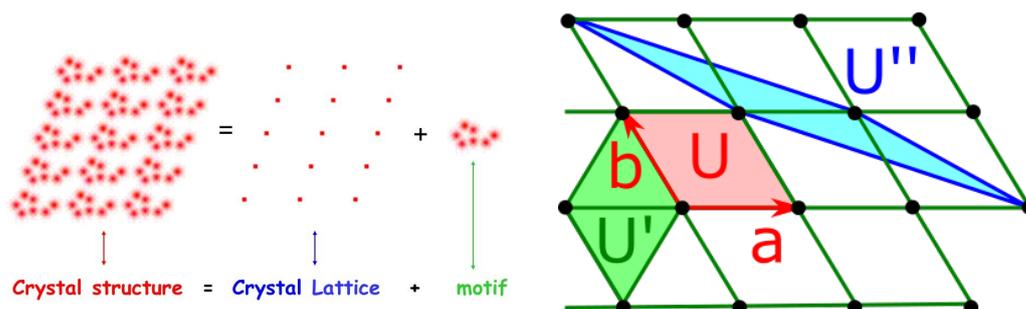
**Keywords and phrases** Voronoi cell, crystal lattice, distance metric, crystal structure prediction

**Related Version** The 15-page version of this abstract has been accepted in the journal Crystal Research and Technology

Crystals are solid materials which consist of an underlying periodic lattice and a set of particles (atoms, molecules or ions) called a *motif*, see Figure 1. More formally, given 3 basis vectors  $\vec{v}_1, \vec{v}_2, \vec{v}_3$ , a *lattice* is the discrete set of their linear combinations with integer coefficients:  $L = \{ \sum_{i=1}^3 t_i \vec{v}_i \in \mathbb{R}^3 \mid t_i \in \mathbb{Z} \}$ . If we restrict the coefficients  $t_i$  of linear

combinations to unit interval  $r_i \in [0, 1]$ , we get the *unit cell*  $U = \{ \sum_{i=1}^3 t_i \vec{v}_i \in \mathbb{R}^3 \mid t_i \in [0, 1] \}$ .

A *unit cell* is an elementary block containing a set of particles and is periodically repeated by translations along the vectors  $\vec{v} \in L$ . Any lattice, hence a crystal, can be defined by infinitely many unit cells, see the right hand side picture of Figure 1. This data representation problem in crystallography is an example of the *curse of ambiguity* saying that the same real-life object can be represented in (often infinitely) many different ways.



■ **Figure 1 Left:** any periodic crystal is defined by a motif consisting of finitely many particles and a lattice  $L$ , which periodically translates the motif. **Right:** any lattice can be defined by infinitely many different unit cells, which makes the conventional crystal representation ambiguous.

The ‘embarrassment of over-prediction’ [1] in the Crystal Structure Prediction (CSP) means that the state-of-the-art CSP software outputs thousands or even millions of simulated crystals, though very few stable crystals can really exist. This over-prediction is due to repeated runs of iterative algorithms that minimize the energy of a crystal, hence stop

at approximate local minima that can be close to the same crystal. Chemists will try to synthesize real crystals in a lab only after they can be sure that potential new materials have desired properties. That is why more and slower simulations of target properties such as solubility or gas adsorption are currently run on each simulated crystal by supercomputers.

A lot of resources and time can be saved if we can identify nearly identical crystals and run further simulations on a much smaller subset of representative crystals. Hence quantifying a similarity between different crystals with the same chemical composition is our main motivation to define new distances, which were unknown even for lattices.

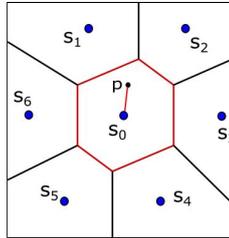
We propose new distances to quantify a similarity between arbitrary lattices considering them modulo any rigid motions (compositions of rotations and translations in  $\mathbb{R}^3$ ). The new distances satisfy the metric axioms and are based on the Voronoi cell of a point in a lattice.

► **Definition 1.** *Given a point  $s_0$  in a lattice  $L$ , which can be assumed to be the origin of  $\mathbb{R}^3$ , the Voronoi cell of  $s_0$  consists of all points  $p$  that are non-strictly closer to  $s_0$  (in the usual Euclidean distance  $d$ ) than to all other points of the lattice  $L$ , i.e.*

$$V(L) = \{p \in \mathbb{R}^3 \mid d(s_0, p) \leq d(s_j, p) \forall s_j \in L - s_0\}$$

The brief notation  $V(L)$  without specifying a point  $s_0$  is justified by the fact that the Voronoi cells around different points of a lattice  $L$  are related by translations.

The Voronoi cell of a crystal lattice point is a centrally symmetric polyhedron, see Figure 2. It is geometrically stable in the sense that under lattice perturbations, geometric characteristics such as the perimeter or area change continuously in the Euclidean topology.



■ **Figure 2** The Voronoi cell of a point in a 2D crystal lattice consisting of blue points.

► **Definition 2.** *Given two crystal lattices  $L$  and  $L'$  and their respective Voronoi cells  $V(L)$  and  $V(L')$  at the origin  $0 \in \mathbb{R}^3$ , we define the minimum offset  $r$  over all rotations  $R$  as*

$$\text{offset}(L, L') = \min\{r \geq 0 : R(V(L)) \subset N(V(L'); r)\}.$$

Here the minimum is taken over all rotations about the origin in  $\mathbb{R}^3$ . The neighborhood  $N(V(L'); r)$  consists of all points that have a maximum distance  $r$  from  $V(L')$ . The extended Hausdorff distance between two lattices will be the symmetric maximum between two offsets:

$$d_H(L, L') = \max\{\text{offset}(L, L'), \text{offset}(L', L)\}$$

We prove that the extended Hausdorff Distance  $d_H$  satisfies all metric axioms and is continuous under perturbations of unit cell parameters. The following scale-invariant distance is additionally invariant under uniform scaling of a lattice.

► **Definition 3.** Given two Voronoi cells of lattice points  $V(L)$  and  $V(L')$ , we want to find the minimum scale factor over all rotations as follows:

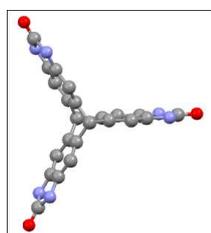
$$\text{scale}(L, L') = \min\{s > 0 : R(V(L)) \subset s * V(L')\}$$

where  $R \in SO(3)$  and  $SO(3)$  is the group of all rotations in  $R^3$ . The scale-invariant distance is defined as:

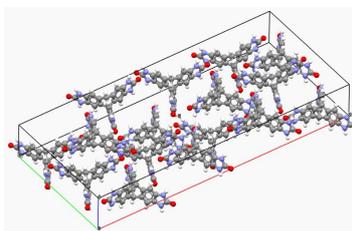
$$d_S(L, L') = \ln\{\max\{\text{scale}(L, L'), \text{scale}(L', L)\}\}$$

To compare crystal lattices, structure similarity is satisfied when  $d_S(L, L')$  or  $d_H(L, L')$  are close to 0.

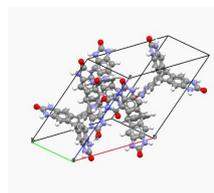
Pulido et al. [2] has demonstrated that organic materials can be discovered by simulating crystals of T2 dataset via optimization of energy and other chemical properties. Crystals of this dataset are made of different conformations of a T2 molecule shown in Figure 3a. Many of them in the dataset can have similar structures that should be filtered to avoid repetitive entries. We ran experiments to find those crystals that can be better distinguished by our metrics than the standard continuous similarity measures Energy and Density. For example, crystals 41 and 47 (very close in Energy and Density) of the T2 dataset have different crystal lattices (Figure 3b and 3c). This difference was found thanks to the Hausdorff and Scaling distance and not by Energy and Density.



(a) T2 molecule.



(b) Unit cell of Crystal structure 41.



(c) Unit cell of Crystal structure 47.

■ **Figure 3** a) T2 molecule. b) and c) Example pair (41,47) with different crystal lattices: close in Energy and Density, but different in Hausdorff and Scaling distances.

---

## References

- 1 A discussion with S. Price about her paper in Faraday discussions. 211:9–30, 2018.
- 2 A. Pulido, L. Chen, T. Kaczorowski, D. Holden, M. Little, S. Chong, B. Slater, D. McMahon, B. Bonillo, C. Stackhouse, A. Stephenson, C. Kane, R. Clowes, T. Hasell, A. Cooper, and G. Day. Functional materials discovery using energy-structure-function maps. *Nature*, 2017.

# Hardness of Approximation for Red-Blue Geometric Covering Problems

Sima Hajiaghahi Shanjani

Department of Computer Science, University of Victoria, Canada  
sima@uvic.ca

---

## Abstract

In Geometric Set Cover a set of points and a family of objects are given, and the goal is to find a minimum sized subset of these objects that covers all the points. This is a fundamental problem which has been studied for over 30 years. It has long been known to be NP-hard and was shown to be APX-hard by Chan and Grant in 2014 [2].

In Red-Blue Geometric Set Cover a set of red points, a set of blue points, and set of objects are given and the goal is to find a subset of the objects that cover all the blue points while covering the minimum number of red points. Chan and Hu in 2015 showed that the problem is NP-hard even when the points are in the plane and objects are axis-aligned unit squares [3]. Here we study Red-Blue Geometric Set Cover when the given objects are axis-aligned rectangles and convex shapes. Here for the first time we show that the problem is APX-hard where the given objects are axis-aligned rectangles. Moreover, we show a hardness of approximation for this problem where the given covering objects are convex objects.

We also study the problem of Boxes Class Cover (BCC): points are given in two colors, red and blue, and the goal is to find a minimum number of axis-aligned rectangles that cover all the blue points but no red. This problem was introduced for the first time in a paper in 2012 by Bereg et al., who showed the problem is NP-hard [1]. No hardness of approximation result has been shown for this problem. Here for the first time we show that this problem is APX-hard.

We also define a restricted version of Max 3SAT, Max RM-3SAT, and we prove that this problem is APX-hard. This problem might be interesting in its own right.

**2012 ACM Subject Classification** Theory of computation → Computational geometry; Theory of computation → Problems, reductions and completeness; Theory of computation → Packing and covering problems

**Keywords and phrases** Computational Geometry, Red-Blue Geometric Set Cover, Axis-Aligned Rectangles, Lower Bound, Inapproximability, APX-hard, Boxes Class Cover

**Funding** Research funded by NSERC Discovery Grant RGPIN 2016-04234

## 1 Results

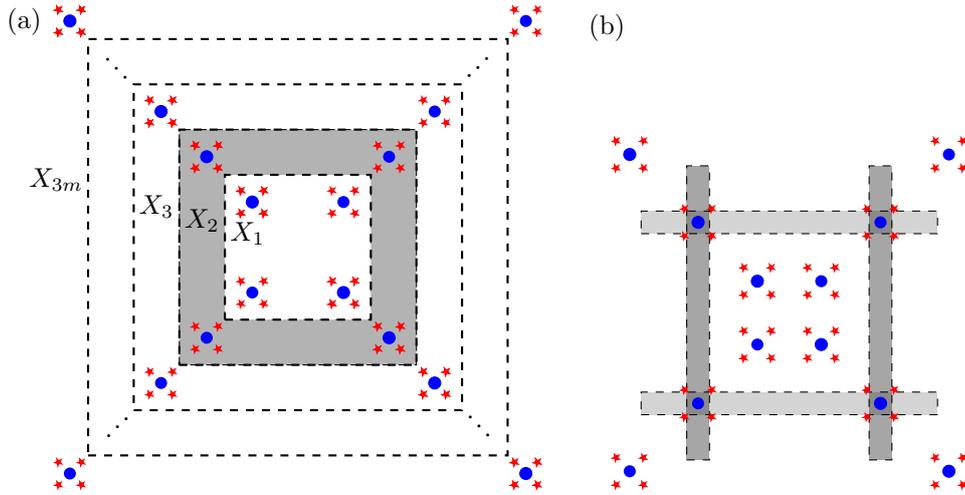
► **Theorem 1.** *The following problems are APX-hard: (i) Boxes Class Cover (ii) Red-Blue Geometric Set Cover, even in the restricted case where all the objects are axis-aligned rectangles and each rectangle contains only one red point and at most 5 blue points.*

► **Theorem 2.** *Red-Blue Geometric Set Cover is NP-hard to approximate within  $(1 - \alpha) \ln n$  factor of optimum for every  $\alpha > 0$ , where the given covering objects are convex objects and  $n$  is the number of blue points.*

## 2 Sketch of the Techniques

► **Definition 3.** *Max Restricted Mixed 3SAT (MAX RM-3SAT). This problem is a variant of MAX 3SAT where all the clauses are of size 2 and 3 and have the following properties:*

1. *All the clauses of size 3 have a literal in negated form and a literal in non-negated form.*



■ **Figure 1** a) The points in the highlighted gray area are *variable points* for  $X_2$ . b) Either the two vertical rectangles or the two horizontal ones are needed to cover each variable's *variable points*.

2. Any variable appears in exactly one clause of size 3, i.e., if  $v_i$  is a variable in this formula, only one of  $v_i$  or  $\bar{v}_i$  can appear in any clause of size 3.
3. Any variable appears in exactly one of the clauses of size 2 in negated form, and in exactly one of the clauses of size 2 in non-negated form.

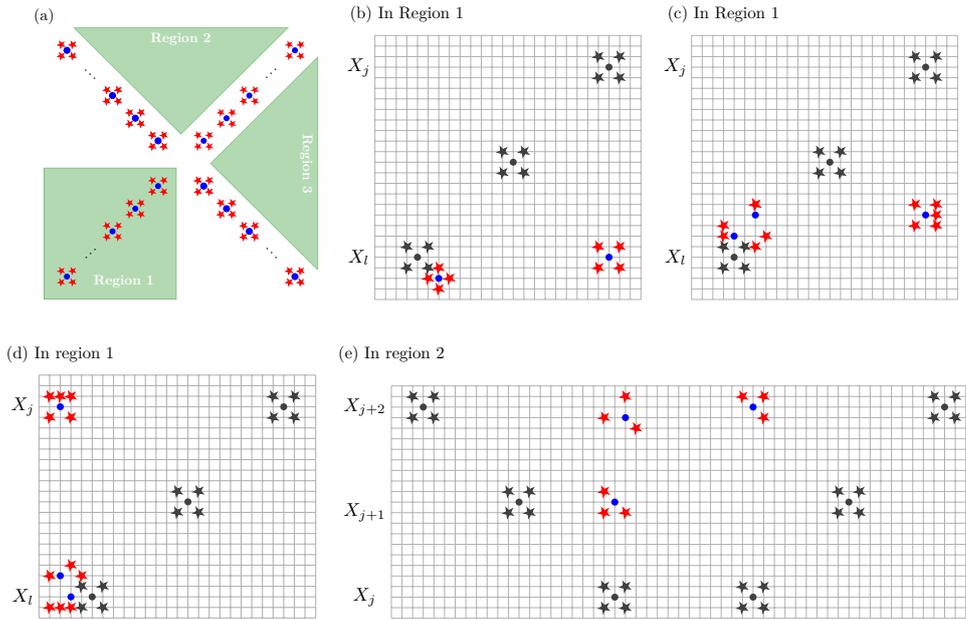
**Sketch of the techniques:** We show the hardness results in Theorem 1 by a series of reductions. In the process of the proof, we first define a new version of MAX 3SAT problem in Definition 3. Then in the first reduction, we transfer an instance of a version of 3SAT to an instance of MAX RM-3SAT by renaming the variables and adding new clauses. We use hardness results on that version of 3SAT and the reduction to show that it is NP-hard to approximate MAX RM-3SAT within a specified constant factor of the optimum.

In the second reduction for BCC, for any instance of MAX RM-3SAT we construct a point structure with red and blue points in polynomial time. Then we show a relation between the number of satisfied clauses in an optimal solution of MAX RM-3SAT and the size of the optimum solution in the corresponding instance of BCC. By considering the hardness result for MAX RM-3SAT, this relation implies the APX-hardness of BCC.

The third reduction is a modified version of the second one. For the reduction from MAX RM-3SAT to Red-Blue Geometric Set Cover, we modify the second reduction in a way that a specific type of rectangles in BCC (called *canonical*), is considered as the family of given candidate rectangles for Red-Blue Geometric Set Cover. The set of blue points is the same blue points. For red points, we observe that there is a space in each of the candidate rectangles to which we can add exactly one distinct red point.

We show the hardness result in Theorem 2 by a reduction from Set Cover. In this reduction, we consider a circle, and for each element  $i$  in Set Cover, we add a blue point  $b_i$  on the circle. For each subset  $s_j$ , we add a red point  $r_j$  on the circle, and we add the convex hull of  $r_j$  and all the  $b_i$ 's that  $i \in s_j$  to the object set for Red-Blue Geometric Set Cover. Set Cover is NP-hard to approximate within a factor of  $(1 - \alpha) \ln m$  of the optimum for every  $\alpha > 0$ , where  $m$  is the number of elements [4]. This implies the result in Theorem 2.

**High-level idea of the structure used in MAX RM-3SAT  $\rightarrow$  BCC:** For  $\Phi$ , an instance of MAX RM-3SAT, we change the order of the clauses to have all the clauses of size



■ **Figure 2** a) Highlighted green areas are divisions of the plane to Region 1-3. b)  $c = (X_j \vee \bar{X}_l)$ , c)  $c = (X_j \vee X_l)$ , d)  $c = (\bar{X}_j \vee \bar{X}_l)$ , e)  $c = (X_j \vee \bar{X}_{j+1} \vee \bar{X}_{j+2})$  (For  $c = (\bar{X}_j \vee X_{j+1} \vee X_{j+2})$ , added points are similar to part (e) but rotated by  $-\pi/2$  in Region 3.).

3 first and then clauses of size 2. We rename the  $j$ th variable of the  $k$ th clause of this order to  $X_{3(k-1)+j}$ . Then, for each variable  $X_i$ ,  $1 \leq i \leq 3m$ , we add 4 blue points on  $(\pm 7i, \pm 7i)$  coordinates and 16 red points on  $(\pm 7i \pm 1, \pm 7i \pm 1)$  coordinates (Figure 1 a). The idea of this structure is to create *variable points* for each variable in a way that an axis-aligned *blue-rectangle* (contains only blue points) cannot cover blue *variable points* of two different variables. We show that BCC on this arrangement of points has to have an optimal solution that covers each variable's blue *variable points* by exactly two rectangles, either both *vertical* or both *horizontal* (Figure 1 b).

The main idea of the reduction from MAX RM-3SAT to BCC is that the choice of vertical vs horizontal corresponds to a true vs a false assignment to the variables. For each clause, we add some red and blue points to force the choice of the covering *blue-rectangles* to be *horizontal* or *vertical* in the optimal solution of BCC based on the structure of the clauses of  $\Phi$ . The locations of these points are different in each type of clauses depending on the size of the clause and the number of negated literals in the clause (Figure 2). In the figures of this paper, we use circles and stars to indicate blue and red points respectively.

## References

- 1 Sergey Bereg, Sergio Cabello, José Miguel Díaz-Báñez, Pablo Pérez-Lantero, Carlos Seara, and Inmaculada Ventura. The class cover problem with boxes. *Comput. Geom.*, 45(7):294–304, 2012. URL: <https://doi.org/10.1016/j.comgeo.2012.01.014>, doi:10.1016/j.comgeo.2012.01.014.
- 2 Timothy M. Chan and Elyot Grant. Exact algorithms and APX-hardness results for geometric packing and covering problems. *Comput. Geom.*, 47(2):112–124, 2014. URL: <https://doi.org/10.1016/j.comgeo.2012.04.001>, doi:10.1016/j.comgeo.2012.04.001.

- 3 Timothy M. Chan and Nan Hu. Geometric red-blue set cover for unit squares and related problems. *Comput. Geom.*, 48(5):380–385, 2015. URL: <https://doi.org/10.1016/j.comgeo.2014.12.005>, doi:10.1016/j.comgeo.2014.12.005.
- 4 Irit Dinur and David Steurer. Analytical approach to parallel repetition. In David B. Shmoys, editor, *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 624–633. ACM, 2014. URL: <https://doi.org/10.1145/2591796.2591884>, doi:10.1145/2591796.2591884.

# Simplicial Oversampling: Accounting for the Higher-order Relations in Data Improves the Solution of the Imbalanced Learning Problem

Oleg Kachan

Skolkovo Institute of Science and Technology, Moscow, Russia  
oleg.kachan@skoltech.ru

---

## Abstract

SMOTE [3] is the established [4] geometric approach to random oversampling to balance classes in the imbalanced classes learning problem [7], followed by many extensions [5, 2, 8]. Its main idea is to introduce synthetic data points in the minor class with each new point being the convex combination of an existing data point and one of its  $k$ -nearest neighbors. This could be viewed as a sampling from the edges of a geometric neighborhood graph.

We propose a generalization of the sampling approach, thus sampling from the maximal simplices (with respect to an ambient space) of the clique complex of a neighborhood graph. That is, a position of a new point is defined by the barycentric coordinates with respect to a simplex spanned by an arbitrary number of data points being sufficiently close. We evaluate the generalized technique and conclude that it outperforms the original SMOTE.

**2012 ACM Subject Classification** ; Computing methodologies → Machine learning algorithms; Mathematics of computing

**Keywords and phrases** Imbalanced classes learning problem, oversampling, Vietoris-Rips complex



■ **Figure 1** *Left:* a neighborhood graph  $G(X)$  modeling a dataset  $X$ , SMOTE introduces new points from the line segments connecting existing data points. *Right:* the clique complex of a neighborhood graph  $(K \circ G)(X)$ , simplicial oversampling introduces new data points from simplices, corresponding to the cliques in a neighborhood graph, respecting the topological features of the space.

## 1 Proposed algorithm

Various neighborhood graphs and their clique complexes could be used to model the data. Given the dataset  $X$ , SMOTE algorithm implicitly uses a  $k$ -nearest neighbors ( $kNN$ ) graph, having  $X$  as its vertices, with an edge between a point  $x \in X$  and a set of its nearest neighbors  $\mathcal{N}(x)$  of cardinality  $k$ . Another popular neighborhood graph is a  $\varepsilon$ -ball neighborhood graph, with an edge between a pair of points  $x$  and  $y$  if they are  $\varepsilon$ -distant, that is  $d(x, y) \leq \varepsilon$ . Its clique complex is known as a *Vietoris-Rips complex* [11] of  $X$ , where a set of points  $U \subseteq X$  is a simplex  $\sigma$  if all the points  $x \in U$  are pairwise  $\varepsilon$ -distant. We can also consider a *continuous  $kNN$  ( $ckNN$ ) graph* [1] which accounts for data local density:

$$G_{ckNN}(X, E), \quad E_{G_{ckNN}}(x, y) = \mathbb{I} \left\{ \frac{d(x, y)}{\delta \sqrt{d(x, x_k) d(y, y_k)}} < 1 \right\}, \quad \text{for } x, y \in X \quad (1)$$

where  $\delta$  is a scale parameter and  $x_k$  and  $y_k$  are  $k$ -nearest neighbors of  $x$  and  $y$ , distances to which serve as local density estimators. Thus, for a fixed  $k$ , it is a modification of  $\varepsilon$ -ball graph, with  $\varepsilon$  being distinct in general for each  $x \in X$ . The clique complex of a ckNN graph is conjectured to capture the true topology of the data at a single value of the scale parameter  $\delta$  contrasted to the persistent homology approach, which considers a range of scales.

The clique complex is obtained from a neighborhood graph by an *expansion* – establishing a correspondence (a bijection) between a  $(k-1)$ -clique in a graph and  $k$ -simplex in a complex. Expressing the desire for a synthetic point to be a convex combination of maximum possible number of existing data points we consider the maximal simplices  $\Sigma_{max}$  only, corresponding to the maximal cliques efficiently found by a maximum clique enumeration algorithm [10].

Geometrically  $k$ -simplex, for all  $k > n$ , is not defined in  $n$ -dimensional space, so the set of maximal simplices  $\Sigma_{max}$  is further refined to the set of *maximal simplices with respect to the ambient space* of dimension  $n$ :

$$\Sigma_{max|n} = \left\{ \bigcup_{j \in J} \rho_j \left| \begin{array}{l} \rho_j = \sigma_j \in \Sigma_{max}, \quad \dim(\sigma_j) > n \\ \rho_j = \{\tau \mid \tau \text{ is } n\text{-face of } \sigma_j \in \Sigma_{max}\}, \text{ otherwise} \end{array} \right. \right\} \quad (2)$$

Given all those considerations we outline the *simplicial oversampling* algorithm:

- construct a neighborhood graph  $G_\theta$  over the points of the minor class  $X_{minor} \subset X$ , given graph parameters  $\theta$ ,
- obtain the clique complex  $K(G_\theta)$  of the graph  $G_\theta$  by finding maximal cliques [10]  $\Sigma_{max}$  of the neighborhood graph and associating a  $(k-1)$ -clique with a  $k$ -simplex,
- consider the set of maximal simplices  $\Sigma_{max|n}$  of the complex  $K(G_\theta)$  with respect to an ambient space of dimension  $n$ , defined by the equation 2,
- randomly select  $m = |X_{major}| - |X_{minor}|$   $k_i$ -simplices  $\{\sigma_i^{(k_i)}\}_{1 < i < m} \subseteq \Sigma_{max|n}$  to sample from, with replacement, according to the uniform distribution,
- from each  $k_i$ -simplex  $\sigma_i^{(k_i)}$  sample a new point  $x_i$  according to uniform distribution over a simplex. This is done efficiently by sampling the barycentric coordinates according to the Dirichlet distribution  $B(x_i) \sim Dir(\alpha)$ , with  $\alpha = (1, \dots, 1) \in \mathbb{R}^{(k+1)}$  and then computing the corresponding Euclidean coordinates with respect to the vertices of the  $k_i$ -simplex.

|                    | Logistic regression |               |               |               | k-NN   |               |               |               |
|--------------------|---------------------|---------------|---------------|---------------|--------|---------------|---------------|---------------|
|                    | Imbal.              | SMOTE         | Simplicial    | Simplicial    | Imbal. | SMOTE         | Simplicial    | Simplicial    |
| Neighborhood graph | kNN                 | kNN           | ckNN          |               | kNN    | kNN           | ckNN          |               |
| Satimage           | 0.7703              | 0.7759        | 0.7855        | <b>0.9114</b> | 0.9363 | 0.9743        | 0.9687        | <b>0.9850</b> |
| Pen digits         | 0.9762              | 0.9811        | 0.9843        | <b>0.9940</b> | 0.9989 | 0.9997        | <b>0.9998</b> | <b>0.9998</b> |
| Abalone            | 0.8463              | 0.8519        | 0.8919        | <b>0.9398</b> | 0.7594 | 0.9382        | 0.9494        | <b>0.9691</b> |
| Spectrometer       | 0.9368              | <b>0.9963</b> | 0.9949        | 0.9948        | 0.9307 | 0.9952        | 0.9974        | <b>0.9978</b> |
| Yeast ML8          | 0.5643              | 0.7649        | 0.7791        | <b>0.8813</b> | 0.5405 | <b>0.8665</b> | 0.8175        | 0.7284        |
| Libras move        | 0.9552              | 0.9995        | 0.9994        | <b>0.9996</b> | 0.9516 | <b>1.0000</b> | <b>1.0000</b> | 0.9985        |
| Coil 2000          | 0.7381              | 0.7961        | 0.8130        | <b>0.8509</b> | 0.6132 | 0.9546        | 0.9681        | <b>0.9767</b> |
| Yeast ME2          | 0.8710              | 0.9130        | 0.9358        | <b>0.9748</b> | 0.7824 | 0.9821        | 0.9826        | <b>0.9921</b> |
| Ozone level        | 0.8973              | 0.9577        | <b>0.9640</b> | 0.9611        | 0.7293 | <b>0.9681</b> | 0.9632        | 0.9636        |
| Abalone 19         | 0.7964              | 0.8964        | 0.9097        | <b>0.9164</b> | 0.5450 | 0.9816        | 0.9841        | <b>0.9860</b> |

■ **Table 1** Evaluation of the simplicial oversampling algorithm. The value of ROC AUC is reported.

## 2 Evaluation and conclusions

We evaluated the original SMOTE algorithm and the proposed simplicial generalization over 10 datasets from the Python package `imbalanced-learn` [6]. All datasets contain two classes of data, class imbalance ratio ranges from 10 to 130. Data dimensionality ranges from 8 to 103. We employed the logistic regression and  $k$ -nearest neighbors classifiers with the default parameters from the `scikit-learn` [9] library. Each dataset was preprocessed by removing the mean and scaling to unit variance. We set the neighbor parameter  $k$  for kNN and ckNN graphs proportional to a cardinality of the minor class of a dataset and the scale parameter  $\delta$  for ckNN graph inversely proportional to a dataset dimension. The mean value of ROC AUC metric for 5-fold cross-validation is reported for each dataset in Table 1.

The evaluation shows that simplicial oversampling outperforms the original SMOTE algorithm, further improved by considering the clique complex of a density-aware ckNN neighborhood graph. Simplicial oversampling is orthogonal to many improvements made by SMOTE extensions and could be combined with them, potentially boosting their performance.

---

### References

- 1 Tyrus Berry and Timothy Sauer. Consistent manifold representation for topological data analysis. *arXiv preprint arXiv:1606.02353*, 2016.
- 2 Chumphol Bunkhumpornpat, Krung Sinapiromsaran, and Chidchanok Lursinsap. Safe-level-smote: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 475–482. Springer, 2009.
- 3 Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.
- 4 Alberto Fernández, Salvador Garcia, Francisco Herrera, and Nitesh V Chawla. Smote for learning from imbalanced data: progress and challenges, marking the 15-year anniversary. *Journal of artificial intelligence research*, 61:863–905, 2018.
- 5 Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. Borderline-smote: a new over-sampling method in imbalanced data sets learning. In *International conference on intelligent computing*, pages 878–887. Springer, 2005.
- 6 Guillaume Lemaître, Fernando Nogueira, and Christos K. Aridas. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *Journal of Machine Learning Research*, 18(17):1–5, 2017. URL: <http://jmlr.org/papers/v18/16-365.html>.
- 7 Victoria López, Alberto Fernández, Salvador García, Vasile Palade, and Francisco Herrera. An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information sciences*, 250:113–141, 2013.
- 8 Tomasz Maciejewski and Jerzy Stefanowski. Local neighbourhood extension of smote for mining imbalanced data. In *2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*, pages 104–111. IEEE, 2011.
- 9 Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.
- 10 Etsuji Tomita, Akira Tanaka, and Haruhisa Takahashi. The worst-case time complexity for generating all maximal cliques and computational experiments. *Theoretical Computer Science*, 363(1):28–42, 2006.
- 11 Afra Zomorodian. Fast construction of the vietoris-rips complex. *Computers & Graphics*, 34(3):263–271, 2010.

# Duality in Persistent Homology of Images

**Adélie Garin** 

Laboratory for Topology and Neuroscience, EPFL, Lausanne, Switzerland  
adelie.garin@epfl.ch

**Teresa Heiss** 

IST Austria (Institute of Science and Technology Austria), Klosterneuburg, Austria  
teresa.heiss@ist.ac.at

**Kelly Maggs**

Mathematical Sciences Institute, The Australian National University, Canberra, Australia  
kelly.maggs@anu.edu.au

**Bea Bleile**

School of Science and Technology, University of New England, Armidale, Australia  
bbleile@une.edu.au

**Vanessa Robins** 

Research School of Physics, The Australian National University, Canberra, Australia  
vanessa.robins@anu.edu.au

---

## Abstract

We derive the relationship between the persistent homology barcodes of two dual filtered CW complexes. Applied to greyscale digital images, we obtain an algorithm to convert barcodes between the two different (dual) topological models of pixel connectivity.

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

**Keywords and phrases** Computational Topology, Topological Data Analysis, Persistent Homology, Duality, Digital Topology

**Funding** This project was initiated at the Second Workshop for Women in Computational Topology hosted by the Mathematical Sciences Institute at ANU, Canberra, in July 2019. This workshop also received funding from AWM, NSF, and AMSI.

*Adélie Garin*: SNSF, CRSII5 177237

*Teresa Heiss*: ERC H2020, No. 788183

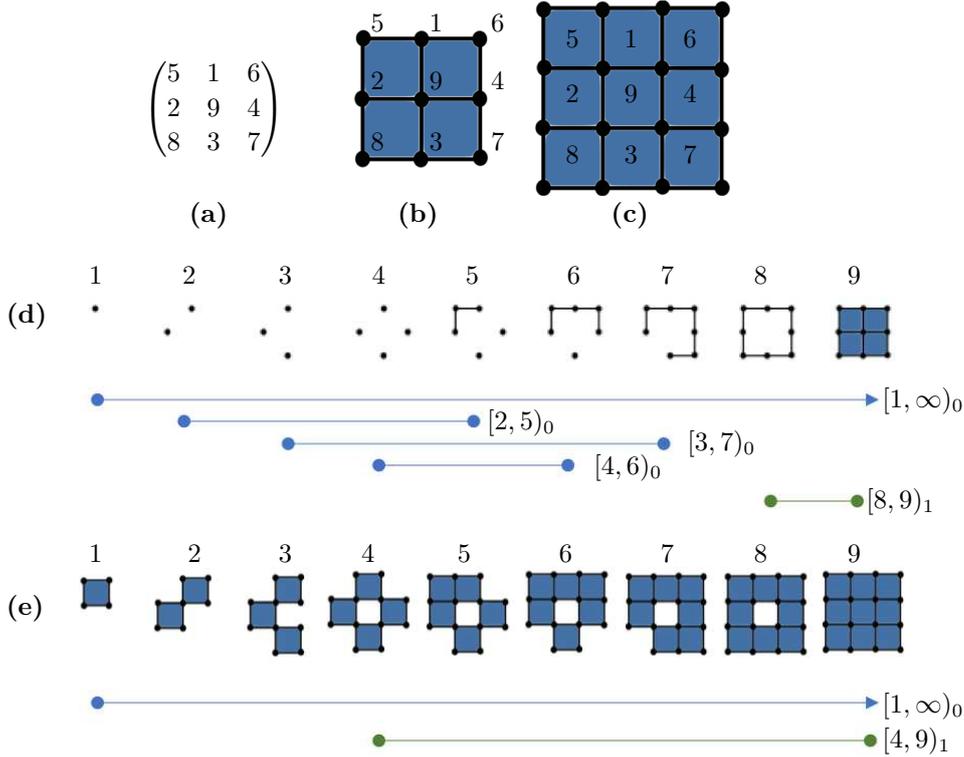
*Vanessa Robins*: ARC Future Fellowship FT140100604

## 1 Introduction

Persistent homology [5, 15] is a stable topological invariant of a filtration, i.e., a nested sequence of spaces  $X_1 \subseteq X_2 \subseteq \dots \subseteq X_n$  ordered by inclusion. The output is a *barcode* or *diagram*,  $Dgm_k$  (see Figure 1d-e), a set consisting of pairs denoted by  $[b, d)_k$  of birth and death indices of each  $k$ -th homology class (representing “holes” of dimension  $k$ ). A *filtered complex* is a pair  $(X, f)$  of a CW complex  $X$  and a cell-wise constant function  $f : X \rightarrow \mathbb{R}$  such that the sublevel sets of  $f$  are subcomplexes. We call two  $d$ -dimensional filtered complexes  $(X, f)$  and  $(X^*, f^*)$  *dual* if (i) each  $k$ -dimensional cell  $\sigma \in X$  corresponds to a  $(d - k)$ -dimensional cell  $\sigma^* \in X^*$ , (ii) the adjacency relations  $\sigma \leq \tau$  of  $X$  are reversed  $\tau^* \leq \sigma^*$  in  $X^*$  and (iii) the filtration order is reversed  $f^*(\sigma^*) = -f(\sigma)$ .

This extended abstract summarises ongoing research that studies the relationship between the persistent homology of two dual filtered complexes. Our results can be seen as versions or extensions of Alexander duality [11]. We simultaneously generalise existing results for simplicial or polyhedral complexes [7], which were confined by a number of restrictions

including to spheres (instead of general manifolds) [3, 6], specific functions [6], or standard homology [3]. While our results are similar to those obtained in the study of extended persistence [2], our constructions and proofs differ significantly. We use a pair of dual complexes filtered by complementary functions, whereas [2] uses a single simplicial complex filtered by sublevel and (relative) superlevel sets. Moreover, our results extend to the case of abstract chain complexes derived from discrete Morse theory [9, 12] and refine, for example, the dual V-paths and discrete Morse functions foreshadowed in [1]. Ultimately this enables us to adapt the image skeletonization and partitioning methods of [4] to a dual version.



■ **Figure 1** (a) The greyscale pixel values of an image represented as an array; (b) and (d) the V-constructed filtered cubical complex (where the pixels are vertices) and its barcode; (c) and (e) the T-constructed filtered cubical complex (where the pixels are the 2-cells) and its barcode.  $Dgm_0$  consists of the blue bars (representing connected components) and  $Dgm_1$  of the green bars (loops).

The first application of our results is to digital image analysis. Images are represented as rectangular arrays of numbers and their topological structure is best captured by a filtered cell complex. Here we focus on two cubical complexes that we refer to as the *T-construction* and *V-construction*, see Figure 1. The T-construction [10] treats pixels as *top dimensional cells* (squares in 2D, cubes in 3D) while the V-construction [13] considers pixels as *vertices*. In both cases, the function values from the original array are extended to all cells of the cubical complex to obtain the filtered complexes  $I_T$  and  $I_V$  respectively. Note that the T-construction corresponds to the indirect adjacency (or closed topology) of classical digital topology and the V-construction to the direct adjacency (or open topology). We present a relationship between the barcodes of  $I_T$  and  $I_V$  below. Previous work in [8] obtains similar results for digital images using extended persistent homology [2]. That approach is

different as it defines a single simplicial complex which is compatible with the piecewise linear foundations of extended persistence. Our Theorem 3.1 shows how to compute the barcode of the T-construction using software designed for the V-construction and vice-versa. Thus, the most suitable software can be chosen independently of construction types. Furthermore, computing higher-dimensional persistent homology barcodes (e.g.  $Dgm_2$  in 3D images) may be optimised by using lower-dimensional ones of the complementary construction ( $Dgm_0$ ).

## 2 Results

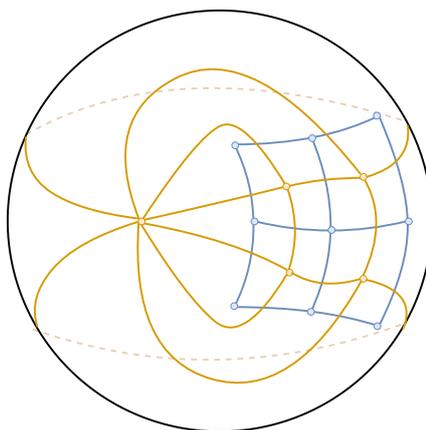
Let  $(X, f)$  and  $(X^*, f^*)$  be dual  $d$ -dimensional filtered CW complexes. Dual face relations lead to relationships at the *filtered chain complex level*, and we show the existence of a shifted filtered chain isomorphism between the absolute filtered cochain complex of  $(X, f)$  and the relative filtered chain complex of  $(X^*, f^*)$ . This induces a *natural* isomorphism between the absolute persistent cohomology of  $(X, f)$  and the relative persistent homology of  $(X^*, f^*)$ . Relying on the work of [14], we ultimately extend these results to a bijection between the absolute persistent homology barcodes of  $(X, f)$  and  $(X^*, f^*)$ .

► **Theorem 2.1.** *Let  $(X, f)$  and  $(X^*, f^*)$  be dual  $d$ -dimensional filtered complexes. There is a bijection between the absolute persistent homology barcode of  $(X, f)$  and  $(X^*, f^*)$ , given by:*

$$\begin{aligned} [p, q] \in Dgm_k(X, f) &\longleftrightarrow [-q, -p] \in Dgm_{d-k-1}(X^*, f^*) \\ [p, \infty) \in Dgm_k(X, f) &\longleftrightarrow [-p, \infty) \in Dgm_{d-k}(X^*, f^*). \end{aligned}$$

## 3 Application to Images

Applying Theorem 2.1 requires dual filtered complexes, but the T- and V-constructions for images are dual *only within the interior* of the image domain. To accommodate this problem and obtain properly dual complexes, we glue a top-dimensional cell to the boundary of the T-construction. The dual complex is then the V-construction with a cone over its boundary.



■ **Figure 2** The blue grid is a cubical complex built from a 2 by 2 array using the T-construction. Gluing a 2-cell to its boundary results in a sphere whose dual is drawn in orange. The orange complex is the V-construction with a cone over the boundary.

To implement this construction we add a single layer of pixels with value  $\infty$  around the boundary of the image array and take the one-point compactification of this padded domain.

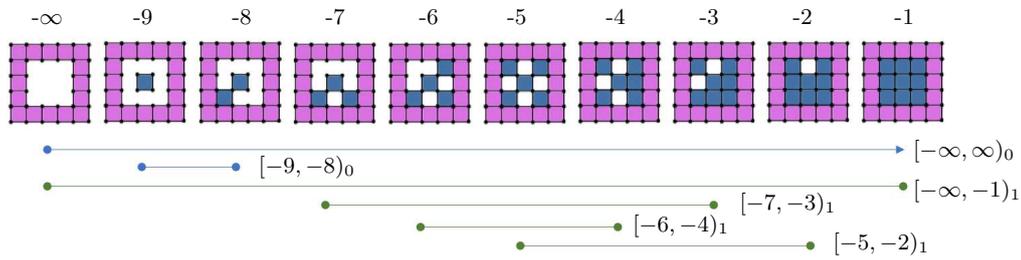
Let  $I$  be the original image array and  $I^\infty$  the padded image. Then  $Dgm_k(I_T) = Dgm_k(I_T^\infty)$  and  $Dgm_k(I_V) = Dgm_k(I_V^\infty)$ . Also note that the one-point compactifications of  $I_T^\infty$  and  $(-I^\infty)_V$  are dual filtered complexes. With a few more accounting steps, we obtain

► **Theorem 3.1.** *Let  $I$  be a  $d$ -dimensional digital image. There are bijections between the barcodes of  $I_T$  and  $(-I^\infty)_V$  and the barcodes of  $I_V$  and  $(-I^\infty)_T$  given by:*

$$\begin{aligned} [p, q] \in Dgm_k(I_V) &\longleftrightarrow [-q, -p] \in \widetilde{Dgm}_{d-k-1}((-I^\infty)_T) \\ [p, q] \in Dgm_k(I_T) &\longleftrightarrow [-q, -p] \in \widetilde{Dgm}_{d-k-1}((-I^\infty)_V) \end{aligned}$$

where  $\widetilde{Dgm}$  denotes the reduced homology, that is, the 0-dimensional bar  $[-\infty, \infty)_0$  is removed.

Figure 3 ( $(-I^\infty)_T$ ) and Figure 1d ( $I_V$ ) illustrate the first bijection of the theorem.



■ **Figure 3** We add a layer of pixels to the image of Figure 1a, consider the negative filtration with the T-construction (we obtain the filtered complex  $(-I^\infty)_T$ ), and compute its barcode; c.f. Fig.1d.

## References

- 1 Ulrich Bauer. *Persistence in discrete Morse theory*. PhD thesis, Goettingen University, 2011.
- 2 David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Extending persistence using Poincaré and Lefschetz duality. *Foundations of Computational Mathematics*, 9:79–103, 02 2009.
- 3 Cecil Jose A. Delfinado and Herbert Edelsbrunner. An incremental algorithm for betti numbers of simplicial complexes on the 3-sphere. *Computer Aided Geometric Design*, 12(7):771–784, 1995.
- 4 Olaf Delgado-Friedrichs, Vanessa Robins, and Adrian Sheppard. Skeletonization and partitioning of digital images using discrete Morse theory. *IEEE transactions on pattern analysis and machine intelligence*, 37(3):654–666, 2015.
- 5 Herbert Edelsbrunner and John Harer. Persistent homology, a survey. *Discrete & Computational Geometry - DCG*, 453, 01 2008.
- 6 Herbert Edelsbrunner and Michael Kerber. Alexander duality for functions: the persistent behavior of land and water and shore. In *Proceedings of the twenty-eighth annual symposium on Computational geometry*, pages 249–258, 2012.
- 7 Herbert Edelsbrunner and Katharina Ölsböck. Tri-partitions and bases of an ordered complex. *Discrete & Computational Geometry*, pages 1–17, 2020.
- 8 Herbert Edelsbrunner and Olga Symonova. The adaptive topology of a digital image. In *Proceedings of the 2012 9th International Symposium on Voronoi Diagrams in Science and Engineering, ISVD 2012*, pages 41–48. IEEE, 06 2012.
- 9 Robin Forman. Morse theory for cell complexes. *Advances in Mathematics*, 134(1):90 – 145, 1998.
- 10 Teresa Heiss and Hubert Wagner. Streaming algorithm for Euler characteristic curves of multidimensional images. In *International Conference on Computer Analysis of Images and Patterns*, pages 397–409. Springer, 2017.

- 11 James R. Munkres. *Elements of algebraic topology*. Addison-Wesley Publishing Company, Menlo Park, CA, 1984.
- 12 Vidit Nanda. *Discrete Morse theory for filtrations*. PhD thesis, Rutgers University-Graduate School-New Brunswick, 2012.
- 13 Vanessa Robins, Peter Wood, and Adrian P Sheppard. Theory and algorithms for constructing discrete Morse complexes from grayscale digital images. *IEEE transactions on pattern analysis and machine intelligence*, 33, 05 2011.
- 14 Vin Silva, Dmitriy Morozov, and Mikael Vejdemo-Johansson. Dualities in persistent (co)homology. *Inverse Problems - INVERSE PROBL*, 27, 07 2011.
- 15 Afra Zomorodian and Gunnar Carlsson. Computing persistent homology. *Discrete & Computational Geometry*, 33(2):249–274, Feb 2005.

# An Enumeration and Isotopy Classification of Oriented Textile Structures

Matt Bright, Vitaliy Kurlin

Computer Science and Materials Innovation Factory  
University of Liverpool, United Kingdom

---

## Abstract

---

We have developed a first method for encoding isotopy classes of links embedded in a thickened torus by 1-dimensional strings of symbols. We have designed a linear time algorithm to determine which codes give rise to realizable textiles in the plane. The new encoding and realizability algorithm have allowed us to enumerate for the first time all oriented single component textiles up to three crossings.

**2012 ACM Subject Classification** Mathematics of computing → Topology

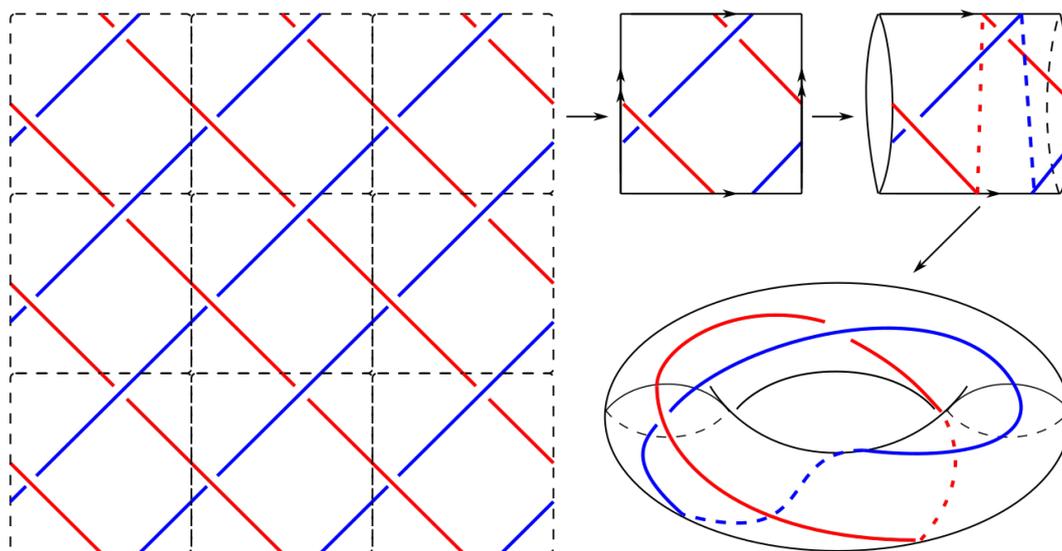
**Keywords and phrases** Computational Topology, Topological Classification, Knot, Link, Periodic Structure, Ambient Isotopy

**Related Version** A 12 page version has been submitted to the SMI2020 conference

## 1 Representation of Textile Structures in a Torus Diagram

A *textile structure* is a collection of infinite continuous curves crossing over each other in a thickened plane that is *doubly periodic*, in that the pattern of crossings is invariant under translations along two linearly independent vectors that form a *unit cell* in the plane.

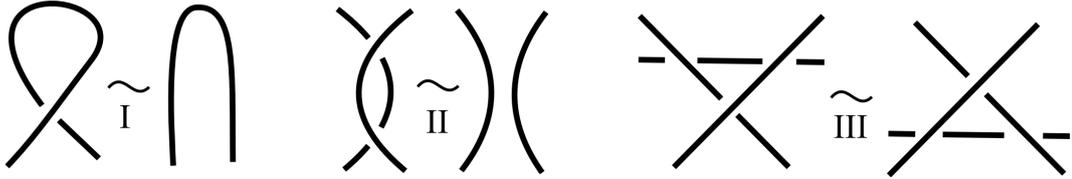
We can represent such structures as a collection of (possibly oriented) circles in a thickened torus by identifying the opposite edges (with the same orientations) of a unit cell as shown in Figure 1 (see [1] for an earlier introduction to periodic textile structures).



■ **Figure 1** Representation of a textile as a *link* (a collection of circles) in the thickened torus

## 2 Encoding Textile Structures by 1-dimensional Strings

Textile structures and links in a thickened torus are considered modulo ambient isotopies that are continuous deformations of the ambient space. Traditional representations include planar diagrams related by Reidemeister moves in Fig. 2. Computation with such structures requires an approach to encoding their topological data. To this end we have extended the *Gauss Code* [2] to the situation of an oriented torus diagram, the *Textile Code*.

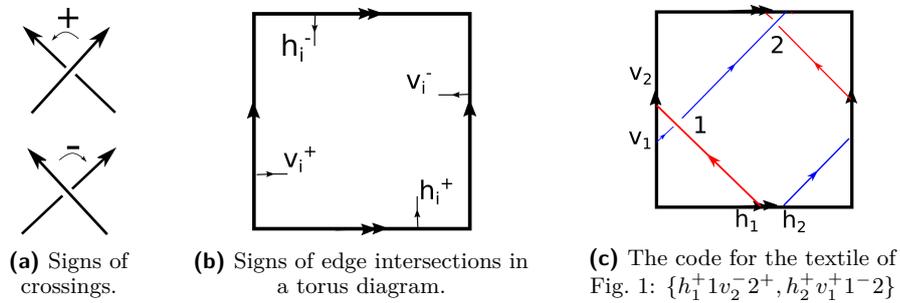


■ **Figure 2** Reidemeister moves generate isotopies of links in  $\mathbb{R}^3$  and the thickened torus.

The code consists of a collection of words, one for each component in the textile. Words consist of symbols  $i, i^\pm$  for  $i \in \mathbb{N}$ , or symbols  $h_i^\pm, v_i^\pm$  for  $i \in \mathbb{N}$  built up as follows:

As we proceed along an oriented arc in a diagram, we add the index of a crossing when it is encountered as an overcrossing, and the index of a crossing with a  $\pm$  superscript indicating its *sign* defined in Fig. 3a when we encounter it as an undercrossing.

When an arc intersects a horizontal or vertical edge of a diagram, we add an  $h_i^\pm$  (resp.  $v_i^\pm$ ) symbol whose subscript indicates its ordering along the edge of a diagram, and whose superscript indicates the direction the arc is travelling at the intersection, see Fig. 3b.



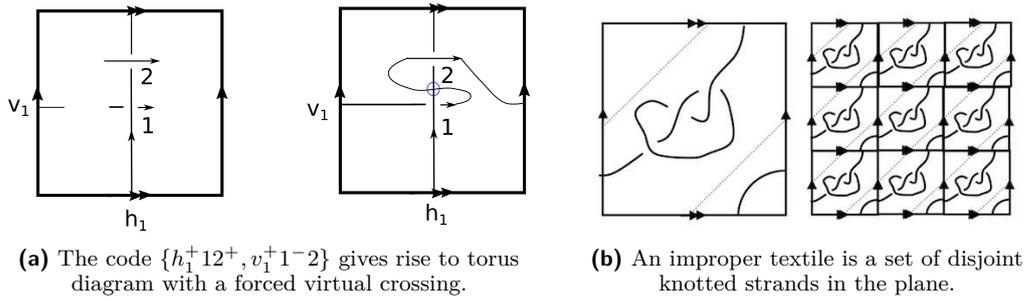
■ **Figure 3** Textile code consists of symbols of crossings and intersections with signs.

Since all components in a textile are cyclic, this encoding is defined up to cyclic permutations.

This information also allows textile diagrams to be reconstructed from codes Fig. 4a we see that some codes induce *virtual* crossings where no crossing indicator appears in the code.

It is also possible for a code to give rise to a torus diagram that represents a disjoint union of knotted strands. We define diagrams whose planar representations are not disjoint in this way as *proper* (see Fig. 4b.)

Clearly in both cases the structures arising would not be viable textiles. The algorithm described in this work detects codes which give virtual crossings.

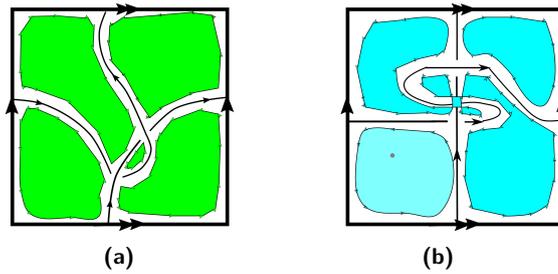


■ **Figure 4** Textile codes that are not realizable by proper textile structures.

### 3 Enumeration of Realizable, Proper, Distinct Textiles

The main result of our work has been the development of an algorithm of time complexity  $O(n)$ , where  $n$  is the number of symbols in of a given textile code, which is able to determine whether or not that code gives rise to a textile with no virtual crossings.

The algorithm is based on considering a textile diagram as a graph inscribed on the torus surface. A diagram with no virtual crossings will divide the surface into consistently oriented cycles, as illustrated in Fig. 5, which can be read directly from the encoding.



■ **Figure 5** Division of the torus surface by a real and a virtual textile

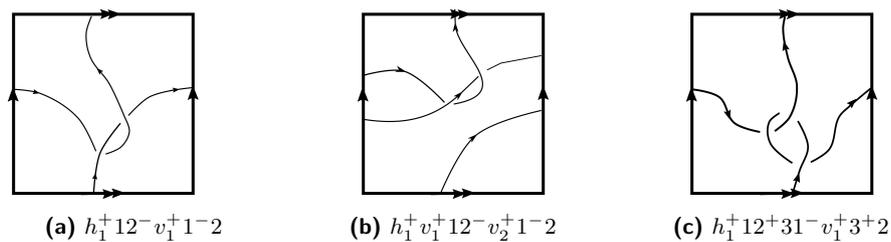
We have implemented this algorithm in C++ and employed it to enumerate all codes giving rise to realizable textile diagrams with up to three crossings. We have then been able to determine, using an existing invariant for links in thickened surfaces [3] and further examination of diagrams which of these are proper and distinct up to ambient isotopy.

| Crossings | $v$ symbols | $h$ symbols | Codes  | Realizable textiles | Distinct Textiles |
|-----------|-------------|-------------|--------|---------------------|-------------------|
| 2         | 1           | 1           | 3840   | 368                 | 8                 |
| 2         | 2           | 1           | 23040  | 2816                | 16                |
| 3         | 1           | 1           | 161280 | 24960               | 8                 |

■ **Table 1** Enumeration of all oriented single component textiles with up to three crossings.

Table 1 shows our reduction from the set of all valid textile codes, through to realizable textiles and then finally to proper and isotopically distinct textiles with different numbers of crossing symbols and edge intersections. Figure 6 shows examples of a code giving rise to two oriented 2-crossing textile and one 3-crossing textile.

As the number of symbols in a code increases, the number of possible codes becomes exponentially large. A key challenge is to directly determine whether two codes give rise to



■ **Figure 6** Examples of codes giving rise to distinct oriented textiles with two and three crossings.

isotopic textiles, so that we may eliminate as many codes as possible before applying our algorithm.

---

### References

- 1 S. Grishanov et al. A topological study of textile structures. part i: An introduction to topological methods. *Textile research journal*, 79(8):702–713, 2009.
- 2 Vitaliy Kurlin. Gauss paragraphs of classical links and a characterization of virtual link groups. *Math. Proc. Cambridge Phil. Society*, 145(1):129–140, 2008.
- 3 MV Zenkina. An invariant of knots in thickened surfaces. *Journal of Mathematical Sciences*, 214(5):728–740, 2016.

# Decomposition and Partition Algorithms for Tissue Dissection

**Maike Buchin** 

Ruhr University Bochum, Germany  
maike.buchin@rub.de

**Leonie Selbach** 

Ruhr University Bochum, Germany  
leonie.selbach@rub.de

---

## Abstract

---

We consider decomposing simple polygons and partitioning graphs under specific constraints. Our research is motivated by an application in the field of protein diagnostics, that is processing tissue samples with laser capture microdissection (LCM) for proteomic analysis [5]. LCM is used to separate disease-specific regions of interest (ROI) from heterogeneous tissue samples. For this dissection to be successful the ROI has to satisfy certain constraints in size and morphology. We consider different algorithmic approaches for the segmentation of the ROI such that each fragment fulfills the criteria to be successfully dissected.

**2012 ACM Subject Classification** Applied computing → Bioinformatics; Theory of computation → Computational geometry; Theory of computation → Dynamic programming; Mathematics of computing → Graph algorithms

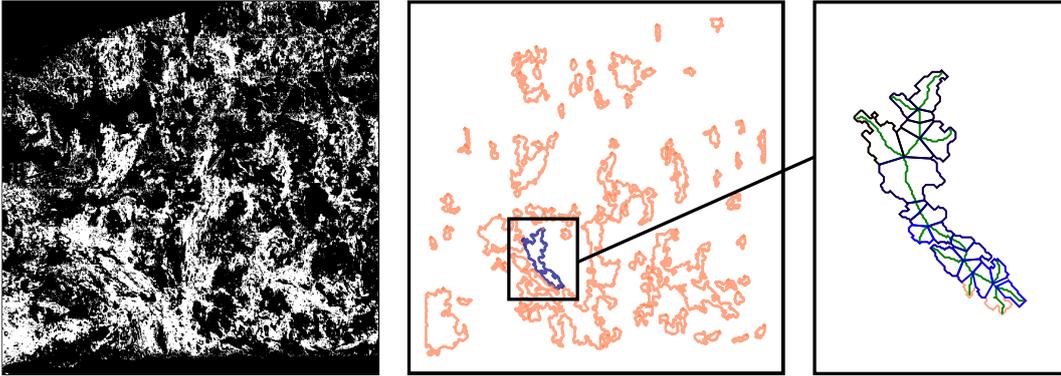
**Keywords and phrases** Polygon decomposition, Graph partitioning, Laser capture Microdissection

## 1 Polygon decomposition

We receive a binary mask of the tissue slide with the foreground being the ROI. We can interpret each connected component as a simple polygon (see Fig. 1). The problem is the following: Given a simple polygon  $P$ , compute an optimal feasible decomposition of  $P$  – meaning that every subpolygon should be feasible. There are different feasibility criteria and optimization goals possible, as well as combinations of these. We considered, for example, constraining the size, convexity or fatness of the subpolygons. Furthermore, we considered minimizing the number or the length of cut edges or maximizing the fatness of the polygons.

Our approach is a skeleton-based decomposition where we compute a discrete and pruned version of the medial axis [1]. The medial axis is the set of points that have more than one closest point on the boundary of the object. Thus, every skeleton point (or pixel) has at least two corresponding points, called contact points, on the boundary. In our approach, we only allow cuts that are line segments between a skeleton point and its corresponding contact points (see Fig. 1 and 2). This results in simple cuts and a flexible framework allowing to integrate different criteria. For a polygon without holes, the skeleton  $S$  is given as an acyclic graph consisting of arcs  $S_k$ , called skeleton branches, which meet at branching points. We considered two cases: Either decomposing each subpolygon belonging to one skeleton branch on its own, or decomposing the whole polygon at once [2].

In the first case, we have the polygons with linear skeletons  $S_k$ . Thus, two skeleton points  $i$  and  $j$  together can generate a subpolygon – denoted by  $P_k(i, j)$ . We can easily decompose the polygon by dynamic programming: There exists a feasible decomposition of the polygon  $P_k(1, j)$  if either  $P_k(1, j)$  is feasible or there exists an  $i < j$  such that  $P_k(i, j)$  is feasible and  $P_k(1, i)$  can be decomposed. Given an optimization goal, we can choose the optimal  $i$ .



■ **Figure 1** Polygon decomposition as a model for tissue fragmentation.

► **Theorem 1.** *Given a simple polygon  $P$  with a discrete skeleton  $S$  consisting of  $n$  skeleton points, one can compute an optimal feasible decomposition of  $P$  based on the skeleton branches in  $\mathcal{O}(n^2F)$  time.*

The factor  $F$  is based on the feasibility criteria and how efficient one can decide if a subpolygon is feasible – it might depend on e.g. the number of polygon or skeleton vertices.

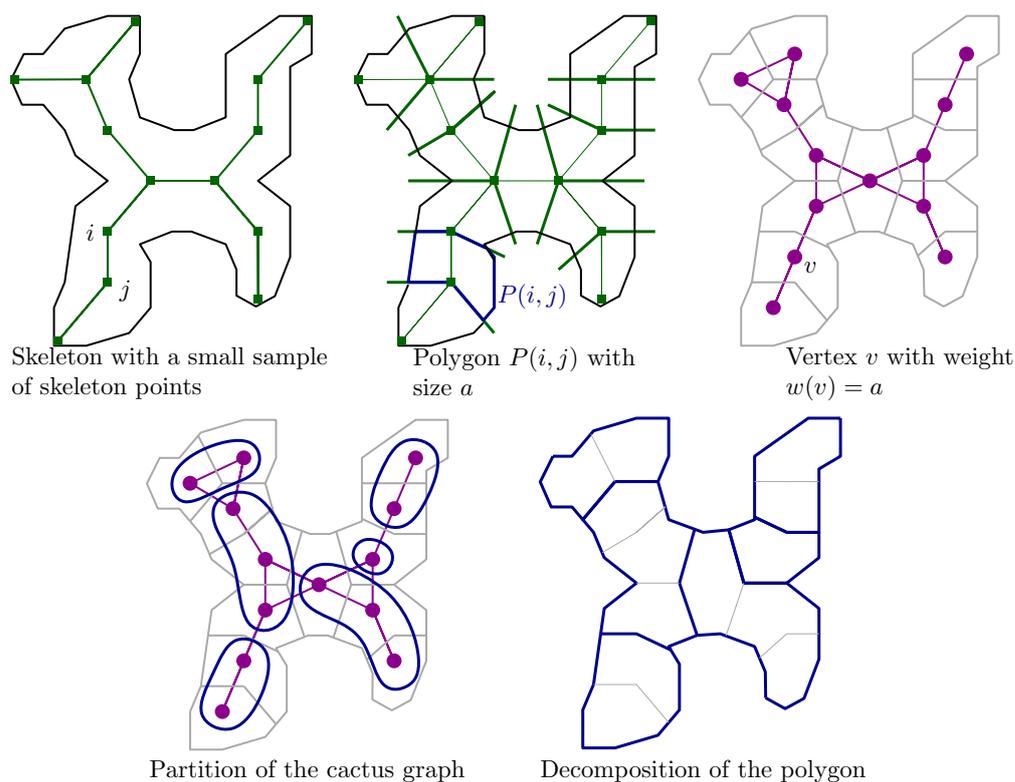
In the second case, a subpolygon can be generated by more than two skeleton points. There are two important observations: Every subpolygon can be represented as a union of subpolygons generated by two skeleton points and the maximal number of skeleton points that generate a polygon is equal to the number of leaves in the skeleton tree. The skeleton computed for our application has the property that the maximal degree of a skeleton point is three. For that case, we developed a method which also used dynamic programming. We select some branching point as the root and then consider a rooted skeleton tree. We use a bottom-up approach and compute if there exists a feasible decomposition of the subpolygon ending in a skeleton point  $j$ , which we denote by  $P(j)$ . This is the case if either  $P(j)$  is feasible or there exists a feasible subpolygon  $P'$  ending in  $j$  and feasible decomposition of all connected components of  $P(j) \setminus P'$ . Thus, we have to consider all possible combinations of skeleton points in the subtree of  $j$  that together with  $j$  can form such a polygon  $P'$ .

► **Theorem 2.** *Given a simple polygon  $P$  with a discrete skeleton  $S$  consisting of  $n$  skeleton points with degree at most 3, one can compute an optimal feasible decomposition of  $P$  based on  $S$  in  $\mathcal{O}(n^kF)$  time, where  $k$  is the number of leaves in the skeleton tree.*

It remains open whether this problem is NP-hard if the degree in the skeleton is not bounded or the polygon has holes, which leads to a cyclic skeleton.

## 2 Graph partitioning

If we only restrict the area of each subpolygon, we can reduce the problem of finding a minimal feasible skeleton-based decomposition of a simple polygon  $P$  to finding a minimal  $(l, u)$ -partition of a weighted cactus graph, that is a graph where every two simple cycles share at most one vertex. The parameters  $l$  and  $u$  are the lower and upper bounds for the area resp. the weight of each cluster in the partition. Take the maximal decomposition  $D_{max}$  of the polygon based on a skeleton  $S$  by including all possible cut line segments. Build a graph such that every vertex  $v$  corresponds to a polygon  $P_v$  in  $D_{max}$  with the weight  $w(v)$  being the area of  $P_v$ . Include edges  $(u, v)$  if the polygons  $P_u$  and  $P_v$  are adjacent – meaning



■ **Figure 2** Reduction of skeleton-based polygon decomposition to cactus graph partitioning.

they share a line segment. With our skeleton this generates a cactus graph with  $n - 1$  vertices and cycles of length 3 (see Fig. 2), but the following results refer to general cactus graphs.

We proved that a  $(l, u)$ -partition of a cactus graph with the minimal (as well as maximal or some fixed) number of clusters can be computed in polynomial time [3, 4]. The idea of our approach is the following: First, we represent the graph by a tree and store all cycles. Then, we compute so-called extendable  $(l, u)$ -partitions for each subtree  $T_v^i$  (the subtree rooted in the node  $v$  with its first  $i$  children) in a bottom-up manner. Every partition can be computed dynamically by combining partitions of  $T_v^{i-1}$  and  $T_{v_i}$  (subtree rooted in the  $i$ -th child of  $v$ ). We include a procedure that deals with cycles by considering different configurations in the tree where in each configuration one edge is deleted and another is added.

► **Theorem 3.** *Given a weighted cactus graph  $G$  and integers  $l, u \geq 0$  with  $l \leq u$ , one can compute a minimal resp. maximal  $(l, u)$ -partition of  $G$  in  $\mathcal{O}(n^6)$  time. Given an integer  $p > 0$ , one can compute a  $(l, u)$ -partition with exactly  $p$  clusters in  $\mathcal{O}(p^4 n^2)$  time.*

It remains open if other decomposition criteria can be modeled as graph partitioning problems as well. Moreover, it would be interesting from a theoretical standpoint whether those graph partition problems are NP-hard or polynomial-time solvable for other graph classes.

---

## References

- 1 Xiang Bai, Longin Jan Latecki, and Wen-Yu Liu. Skeleton pruning by contour partitioning with discrete curve evolution. *IEEE transactions on pattern analysis and machine intelligence*, 29(3), 2007.

- 2 Maïke Buchin, Axel Mosig, and Leonie Selbach. Skeleton-based decomposition of simple polygons. In *Abstracts of 35th European Workshop on Computational Geometry*, 2019. URL: <http://www.eurocg2019.uu.nl/papers/3.pdf>.
- 3 Maïke Buchin and Leonie Selbach. A polynomial-time partitioning algorithm for weighted cactus graphs, 2020. [arXiv:2001.00204](https://arxiv.org/abs/2001.00204).
- 4 Maïke Buchin and Leonie Selbach. A polynomial-time partitioning algorithm for weighted cactus graphs. In *Abstracts of 36th European Workshop on Computational Geometry*, 2020. URL: [http://www1.pub.informatik.uni-wuerzburg.de/eurocg2020/data/uploads/papers/eurocg20\\_paper\\_17.pdf](http://www1.pub.informatik.uni-wuerzburg.de/eurocg2020/data/uploads/papers/eurocg20_paper_17.pdf).
- 5 Frederik Großerueschkamp, Thilo Bracht, Hanna C Diehl, Claus Kuepper, Maïke Ahrens, Angela Kallenbach-Thieltges, Axel Mosig, Martin Eisenacher, Katrin Marcus, Thomas Behrens, et al. Spatial and molecular resolution of diffuse malignant mesothelioma heterogeneity by integrating label-free ftir imaging, laser capture microdissection and proteomics. *Scientific reports*, 7:44829, 2017.

# Minimal Delaunay triangulations of hyperbolic surfaces

**Matthijs Ebbens**

Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence, University of Groningen, Netherlands  
y.m.ebbens@rug.nl

**Hugo Parlier**

Mathematics Research Unit, University of Luxembourg, Luxembourg  
hugo.parlier@uni.lu

**Gert Vegter**

Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence, University of Groningen, Netherlands  
g.vegter@rug.nl

---

## Abstract

Motivated by recent work on Delaunay triangulations of hyperbolic surfaces, we consider the minimal number of vertices of such triangulations. In particular, we will show that every hyperbolic surface has a Delaunay triangulation where edges are given by distance paths and where the number of vertices grows linearly as function of the genus  $g$ . We will show that the order of this bound is attained for some families of surfaces. Finally, we will give an example showing that the  $\Omega(\sqrt{g})$  lower bound in the more general case of topological surfaces is tight for hyperbolic surfaces as well.

**2012 ACM Subject Classification** Computational Geometry; Graphs and Surfaces

**Keywords and phrases** Delaunay triangulation, hyperbolic surface, pair of pants decomposition

**Funding** *Hugo Parlier*: Research partially supported by ANR/FNR project SoS, INTER/ANR/16/11554412/SoS, ANR-17-CE40-0033.

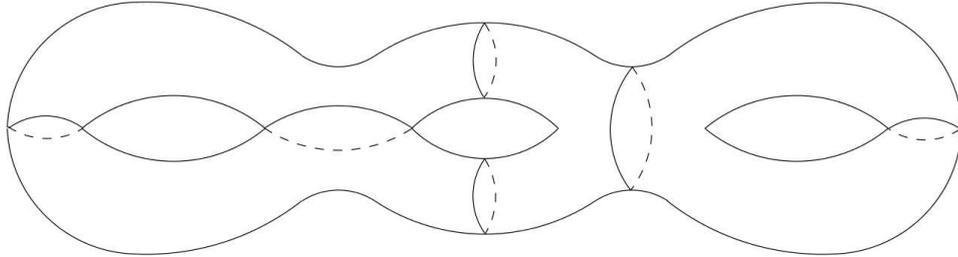
## 1 Introduction

Recently, the classical topic of Delaunay triangulations has been studied in the context of hyperbolic surfaces. Bowyer's incremental algorithm for computing simplicial Delaunay triangulations in the Euclidean plane [3] has been generalized to orientable hyperbolic surfaces and implemented for some specific cases [2, 8]. Moreover, it has been shown that the flip graph of geometric (not necessarily simplicial) Delaunay triangulations on a hyperbolic surface is connected [5].

We consider the minimal number of vertices of *simplicial* Delaunay triangulations of a hyperbolic surface of genus  $g$ . Motivated by the interest in graph embeddings on metric surfaces where edges are represented by shortest paths between their endpoints [6, 7], we restrict ourselves to *distance* Delaunay triangulations (DDT), where edges are shortest distance paths. In particular, we will give an upper bound for the minimal number of vertices of DDTs and show that the order of this upper bound is attained for some families of hyperbolic surfaces. Moreover, we will construct a class of hyperbolic surfaces to show that the only known  $\Theta(\sqrt{g})$  lower bound for the number of vertices in the more general case of simplicial triangulations of topological surfaces of genus  $g$  is tight for DDTs as well.

## 2 Preliminaries

There are several models for the hyperbolic plane [1]. In the Poincaré disk model, the hyperbolic plane is represented by the unit disk  $\mathbb{D}$  in the complex plane equipped with a specific Riemannian metric of constant Gaussian curvature  $-1$  such that hyperbolic lines, i.e., geodesics, are given by diameters of  $\mathbb{D}$  and circle segments intersecting the boundary of  $\mathbb{D}$  orthogonally. A hyperbolic surface is a connected 2-dimensional manifold which is locally isometric to an open subset of the hyperbolic plane [13]. The metric on the hyperbolic plane induces a metric on hyperbolic surfaces, which allows us to speak of geodesics on hyperbolic surfaces. We will always assume our surfaces to be orientable and compact. A set of  $3g - 3$  mutually disjoint simple closed geodesics decomposes a hyperbolic surface of genus  $g$  into  $2g - 2$  pairs of pants, i.e., topological spheres with three holes (see Figure 1) [4]. Pants decompositions can be used to provide a parametrization of Teichmüller space, a natural deformation space of hyperbolic surfaces. On hyperbolic surfaces we consider *distance* Delaunay triangulations (DDT). Each DDT satisfies the following three properties: (1) it is a simplicial complex; (2) it satisfies the empty circumcircle property (Delaunay); (3) its edges are shortest distance paths on the hyperbolic surface.



■ **Figure 1** Decomposition of a genus 3 surface into 4 pair of pants using 6 closed geodesics.

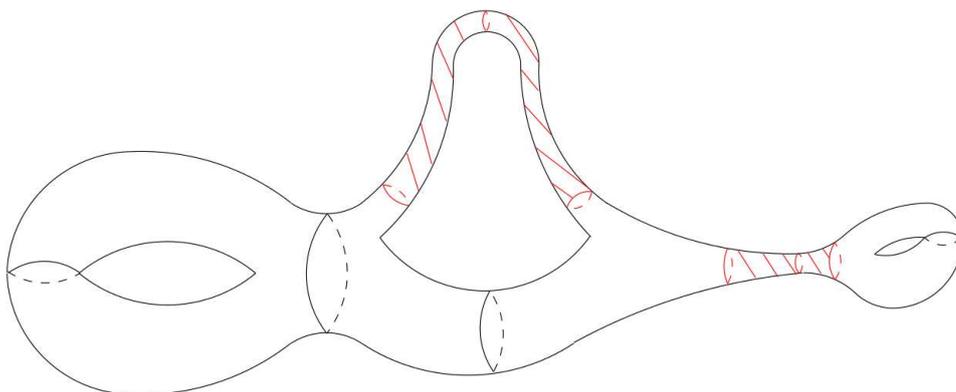
## 3 Results

In this section, we will present our results, starting from an upper bound for the number of vertices of a DDT of a hyperbolic surface.

► **Theorem 1.** *There exists a constant  $A \in \mathbb{R}$  such that for every closed hyperbolic surface of genus  $g$  there exists a DDT with at most  $Ag$  vertices.*

To give a sketch of the proof, let  $\varepsilon > 0$  be a fixed constant and assume first that for a given hyperbolic surface  $\mathbb{M}_g$  of genus  $g$  the injectivity radius  $\text{injrads}(x)$ , i.e., the radius of the largest embedded disk at  $x$ , is at least  $\varepsilon$  for all  $x \in \mathbb{M}_g$ . Consider a maximal set of disjoint, embedded disks of radius  $\frac{1}{8}\varepsilon$  on  $\mathbb{M}_g$ . It can be shown that the Delaunay triangulation of the point set consisting of all centers of the disks is simplicial and that its edges are distance paths. Furthermore, by a packing argument the cardinality of this point set is at most  $Ag$  for some constant  $A$  depending only on  $\varepsilon$ . Secondly, assume that there exists  $x \in \mathbb{M}_g$  with  $\text{injrads}(x) < \varepsilon$ . If we had chosen  $\varepsilon$  sufficiently small in the beginning, then  $\{x \in \mathbb{M}_g \mid \text{injrads}(x) < \varepsilon\}$  consists of a collection of cylinders (see Figure 2). We can triangulate such cylinders by putting a fixed number of points on each of their boundary components, while we triangulate the remainder of the surface as above, finishing the proof.

Now, we will construct a class of hyperbolic surfaces that attains the order of the upper bound in Theorem 1. Fix some interval  $[a, b] \subset \mathbb{R}$ . Let  $G_g$  be the 3-regular graph with  $2g - 2$



■ **Figure 2** Decomposition of a hyperbolic surface into thick part and narrow cylinders.

vertices depicted in Figure 3. Each vertex  $v_i$  of  $G_g$  represents a pair of pants  $Y_i$  and different pairs of pants are glued together along their boundary geodesics according to the edges of  $G_g$ . The lengths of the  $3g - 3$  boundary geodesics are chosen to be in  $[a, b]$ . Note that the set  $S_g(a, b)$  of hyperbolic surfaces of genus  $g$  that can be constructed in this way contains an open subset of the corresponding moduli space.



■ **Figure 3** 3-regular graph  $G_g$ .

► **Theorem 2.** *There exists a constant  $B > 0$  only depending on  $a, b$  such that for any hyperbolic surface  $\mathbb{M}_g \in S_g(a, b)$  a minimal DDT of  $\mathbb{M}_g$  has at least  $Bg$  vertices.*

The intuition behind this construction is as follows. Let  $\mathbb{M}_g \in S_g(a, b)$  and let  $\mathcal{T}$  be any DDT of  $\mathbb{M}_g$ . Recall that vertex  $v_i$  in  $G_g$  corresponds to a pair of pants  $Y_i$ . Let  $e = (u, v) \in E(\mathcal{T})$  with  $u \in Y_i, v \in Y_j$  for  $i < j$ . If  $j - i$  is large, then because of the Delaunay property there will be a large empty circumcircle containing  $e$ . This means that most of the pairs of pants between  $Y_i$  and  $Y_j$  will not contain any vertices of  $\mathcal{T}$ . Hence, roughly speaking, a vertex can only have edges to vertices in a few neighbouring pairs of pants. Because a triangulation of a surface of genus  $g$  has a certain minimum number of edges, we also need a certain minimum number of vertices, since there can only be so many edges per vertex.

► **Remark 3.** The minimal number of vertices for any simplicial triangulation of a *topological* surface of genus  $g$  is  $\Theta(\sqrt{g})$  [9]. Using a topological embedding of a complete graph [12, 14] and adding a specific hyperbolic metric [11], we find that there exists a family of hyperbolic surfaces  $\{\mathbb{M}_{g_i}\}$  with  $g_i \rightarrow \infty$  such that the  $\mathbb{M}_{g_i}$  have a DDT with  $\Theta(\sqrt{g_i})$  vertices. Hence, the  $\Theta(\sqrt{g})$  lower bound is tight for DDTs of hyperbolic surfaces as well.

## References

- 1 Alan F. Beardon. *The geometry of discrete groups*, volume 91 of *Graduate Texts in Mathematics*. Springer-Verlag, 2012.
- 2 Mikhail Bogdanov, Monique Teillaud, and Gert Vegter. Delaunay triangulations on orientable surfaces of low genus. In *Leibniz International Proceedings in Informatics*, editor, *32nd International Symposium on Computational Geometry (SoCG 2016)*, pages 20:1–20:17, 2016.

- 3 Adrian Bowyer. Computing Dirichlet tessellations. *The Computer Journal*, 24(2):162–166, 1981.
- 4 Peter Buser. *Geometry and spectra of compact Riemann surfaces*. Springer-Verlag, 2010.
- 5 Vincent Despré, Jean-Marc Schlenker, and Monique Teillaud. Flipping geometric triangulations on hyperbolic surfaces. *arXiv preprint arXiv:1912.04640*, 2019.
- 6 István Fáry. On straight-line representation of planar graphs. *Acta scientiarum mathematicarum*, 11(229-233):2, 1948.
- 7 Alfredo Hubard, Vojtěch Kaluža, Arnaud De Mesmay, and Martin Tancer. Shortest path embeddings of graphs on surfaces. *Discrete & Computational Geometry*, 58(4):921–945, 2017.
- 8 Jordan Iordanov and Monique Teillaud. Implementing Delaunay triangulations of the Bolza surface. In *Proceedings of the Thirty-third International Symposium on Computational Geometry*, pages 44:1–44:15, 2017.
- 9 Mark Jungerman and Gerhard Ringel. Minimal triangulations on orientable surfaces. *Acta Mathematica*, 145(1):121–154, 1980.
- 10 Bojan Mohar and Carsten Thomassen. *Graphs on surfaces*, volume 10. JHU Press, 2001.
- 11 Hugo Parlier and Camille Petit. Chromatic numbers of hyperbolic surfaces. *Indiana University Mathematics Journal*, pages 1401–1423, 2016.
- 12 Gerhard Ringel and John W.T. Youngs. Solution of the Heawood map-coloring problem. *Proceedings of the National Academy of Sciences*, 60(2):438–445, 1968.
- 13 John Stillwell. *Geometry of surfaces*. Springer-Verlag, 1992.
- 14 Charles M. Terry, Lloyd R. Welch, and John W.T. Youngs. The genus of K12s. *Journal of Combinatorial Theory*, 2(1):43–60, 1967.

# Quantifying barley morphology using Euler characteristic curves

**Erik Amézquita** 

Department of Computational Mathematics, Science & Engineering, Michigan State University  
amezqui3@msu.edu

**Michelle Quigley** 

Department of Horticulture, Michigan State University  
quigle30@msu.edu

**Tim Ophelders** 

Department of Computational Mathematics, Science & Engineering, Michigan State University  
ophelder@msu.edu

**Elizabeth Munch** 

Department of Computational Mathematics, Science & Engineering  
Department of Mathematics, Michigan State University  
muncheli@msu.edu

**Daniel H. Chitwood** 

Department of Computational Mathematics, Science & Engineering,  
Department of Horticulture, Michigan State University  
chitwoo9@msu.edu

---

## Abstract

---

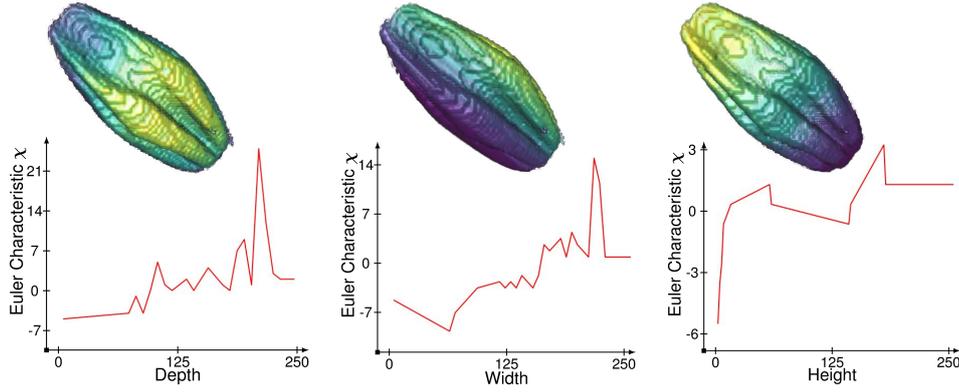
Shape is foundational to biology. Observing and documenting shape has fueled biological understanding, and from this perspective, it is also a type of data. The vision of topological data analysis, that data is shape and shape is data, will be relevant as biology transitions into a data-driven era where meaningful interpretation of large data sets is a limiting factor. We focus on quantifying the morphology of barley spikes and seeds using topological descriptors based on the Euler characteristic and relate the output back to genetic information.

**2012 ACM Subject Classification** Computing methodologies → Volumetric models

**Keywords and phrases** Topological Data Analysis; Euler characteristic transform; barley inflorescence

## 1 Introduction

Shape is data and data is shape. Biologists are accustomed to thinking about how the shape of biomolecules, cells, tissues, and organisms arise from the effects of genetics, development, and the environment. Traditionally, biologists use morphometrics to compare and describe shapes. The shape of leaves and fruits is quantified based on homologous landmarks (similar features due to shared ancestry from a common ancestor) or harmonic series from a Fourier decomposition of their closed contour. While these methods are useful for comparing many shapes in nature, they can not always be used: there may not be homologous points between samples or a harmonic decomposition of a shape is not appropriate. Topological data analysis (TDA) offers a more comprehensive, versatile way to quantify plant morphology. In particular, Euler characteristic curves [11] serve as a succinct, computationally feasible topological signature that allows downstream statistical analyses. For example, Li et al. [4] computed a morphospace for all leaves and used ECCs to predict plant family and location. Others have used the same filter and ECCs to determine the genetic basis of leaf shape in apple [9] and tomato [5] as well as the genetic basis of cranberry shape [3]. ECCs are sensitive enough to detect both complex and subtle effects of rootstock and climate on grapevine



■ **Figure 1** Three different Euler Characteristic Curves (ECCs) from three different filters. (top) X-ray CT scan of a barley seed. The symmetry of the seed encourages a filter by depth, width and height values. Slicing the barley seed in different directions produce (bottom) different corresponding ECCs.

leaf shape [8]. ECCs have also been used to measure the hairiness and shape of spikelets (arrangements of grass flowers) [7] and patterns of vegetation from satellite imagery [6].

## 2 Methods

In this project we are studying the morphology of barley seed and barley spikes (the branching inflorescence). The data arises from an artificial evolution experiment in which parental barley genotypes have been segregating for more than 60 years, corresponding to 60 different generations. In collaboration with Dr. Dan Koenig (UC Riverside), we have access to the seeds of progeny resulting from crosses of the original founders from each generation. Using X-ray CT scanning technology, we have created voxel-based 3D reconstructions of over 875 spikes, from which we have isolated individual seeds from each spike. Given the large number of seeds and voxels per seed, we use of the Euler Characteristic Transform as in [11] to quantify the morphology of each barley spike.

Consider each voxel-based image as a cubical complex  $X$  of dimension  $d = 3$  as in [10]. For a fixed direction  $\nu \in S^{d-1}$ , and a height value  $h$ , we define

$$X(\nu)_h = \{\Delta \in X : \langle x, \nu \rangle \leq h \text{ for all } x \in \Delta\}, \quad (1)$$

to be the subcomplex containing all cubical simplices below height  $h$  in the direction  $\nu$ . The Euler characteristic at height  $h$  is  $\chi(X(\nu)_h) = V - E + S - C$  where  $V, E, S, C$  are the number of vertices, edges, squares and cubes in  $X(\nu)_h$  respectively. The Euler characteristic curve (ECC) of direction  $\nu$  is defined as  $\{\chi(X(\nu)_h)\}_{h \in \mathbb{R}}$ . Turner et al. [11] proved that the collection of all ECCs corresponding to all possible directions effectively summarizes all the morphological information of 3D shapes in general. Moreover, with such collection we would be able to reconstruct the original object. A finite bound on the number of necessary directions for general 3D shapes has been proven [2], although the idea of efficiently reconstructing arbitrary 3D objects solely from ECCs [1] remains elusive.

Using a reduced number of ECCs as descriptors combined with known machine learning techniques, we explore the differences between founders phenotypes and their resulting progeny. This will provide insight into how selection pressures in a common environment alter the morphology of barley varieties originating from across the globe. From the X-ray CT scans we also extract traditional measurements for each barley seed, such as their volume,

surface area, and angle between the seed and the stalk (rachis). Using these measurements as descriptors, we can compare how machine learning techniques fare when using topological information vs. traditional, morphometric information.

### 3 Conclusions

Natural variation in barley, like all crops, encompasses differences in yield and adaptation to diverse climates and terrains. Understanding how differences in morphology affect these traits is vital to improve barley through breeding. TDA combined with X-ray CT scans offers a novel insight into the plant form and its evolution. As a long term plan, we will compare the topological descriptors to available genetic information of each barley sample. This analysis can further our understanding of the relationship between phenotype and genotype.

---

#### References

- 1 Robin Lynne Belton, Brittany Terese Fasy, Rostik Mertz, Samuel Micka, David L. Millman, Daniel Salinas, Anna Schenfisch, Jordan Schupbach, and Lucia Williams. Learning simplicial complexes from persistence diagrams. 2018. [arXiv:1805.10716v2](#).
- 2 Justin Curry, Sayan Mukherjee, and Katharine Turner. How many directions determine a shape and other sufficiency results for two topological transforms. 2018. [arXiv:1805.09782](#).
- 3 Luis Diaz-García, Giovanni Covarrubias-Pazarán, Brandon Schlautman, Edward Grygleski, and Juan Zalapa. Image-based phenotyping for identification of QTL determining fruit shape and size in american cranberry (*vaccinium macrocarpon* l.). *PeerJ*, 6(e5461), 2018. [doi:10.7717/peerj.5461](#).
- 4 Mao Li, Hong An, Ruthie Angelovici, Clement Bagaza, Albert Batushansky, Lynn Clark, Viktoriya Coneva, Michael J. Donoghue, Erika Edwards, Diego Fajardo, Hui Fang, Margaret H. Frank, Timothy Gallaher, Sarah Gebken, Theresa Hill, Shelley Jansky, Baljinder Kaur, Phillip C. Klahs, Laura L. Klein, Vasu Kuraparthi, Jason Londo, Zoë Migicovsky, Allison Miller, Rebekah Mohn, Sean Myles, Wagner C. Otoni, J. C. Pires, Edmond Rieffer, Sam Schmerler, Elizabeth Spriggs, Christopher N. Topp, Allen Van Deynze, Kuang Zhang, Linglong Zhu, Braden M. Zink, and Daniel H. Chitwood. Topological data analysis as a morphometric method: Using persistent homology to demarcate a leaf morphospace. *Frontiers in Plant Science*, 9:553, 2018. [doi:10.3389/fpls.2018.00553](#).
- 5 Mao Li, Margaret H. Frank, Viktoriya Coneva, Washington Mio, Daniel H. Chitwood, and Christopher N. Topp. The persistent homology mathematical framework provides enhanced genotype-to-phenotype associations for plant morphology. *Plant Physiology*, 177(4):1382–1395, 2018. [doi:10.1104/pp.18.00104](#).
- 6 Luke Mander, Stefan C. Dekker, Mao Li, Washington Mio, Surangi W. Punyasena, and Timothy M. Lenton. A morphometric analysis of vegetation patterns in dryland ecosystems. *Royal Society Open Science*, 4(2):160443, 2017. [doi:10.1098/rsos.160443](#).
- 7 Christine A. McAllister, Michael R. McKain, Mao Li, Bess Bookout, and Elizabeth A. Kellogg. Specimen-based analysis of morphology and the environment in ecologically dominant grasses: the power of the herbarium. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 374(1763):20170403, 2019. [doi:10.1098/rstb.2017.0403](#).
- 8 Zoë Migicovsky, Zachary N. Harris, Laura L. Klein, Mao Li, Adam McDermaid, Daniel H. Chitwood, Anne Fennell, Laszlo G. Kovacs, Misha Kwasniewski, Jason P. Londo, Qin Ma, and Allison J. Miller. Rootstock effects on scion phenotypes in a ‘Chambourcin’ experimental vineyard. *Horticulture Research*, 6(64), 2019. [doi:10.1038/s41438-019-0146-2](#).
- 9 Zoë Migicovsky, Mao Li, Daniel H. Chitwood, and Sean Myles. Morphometrics reveals complex and heritable apple leaf shapes. *Frontiers in Plant Science*, 8:2185, 2018. [doi:10.3389/fpls.2017.02185](#).

- 10 V. Robins, P. J. Wood, and A. P. Sheppard. Theory and algorithms for constructing discrete Morse complexes from grayscale digital images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1646–1658, August 2011. doi:10.1109/TPAMI.2011.95.
- 11 K. Turner, S. Mukherjee, and D. M. Boyer. Persistent homology transform for modeling shapes and surfaces. *Information and Inference*, 3(4):310–344, 12 2014. doi:10.1093/imaiai/iau011.

# On the Power of Concatenation Arguments

Gill Barequet

Gil Ben-Shachar

Dept. of Computer Science, Technion—Israel Inst. of Technology, Haifa 3200003, Israel  
{barequet,gilbe}@cs.technion.ac.il

Martha Carolina Osegueda

Dept. of Computer Science, Univ. of California, Irvine, CA 92717  
mosegued@uci.edu

---

## Abstract

We present several concatenation arguments, and show their applications to setting bounds on the growth constants of polyominoes and polycubes, and to families (*e.g.*, trees) of polyominoes and polycubes whose enumerating sequences are pseudo sub- or super-multiplicative.

**2012 ACM Subject Classification** Mathematics of computing → Combinatoric problems

**Keywords and phrases** Polyominoes, Polycubes

**Funding** Work on this paper by the first and second authors has been supported in part by ISF Grant 575/15. Work on this paper by the first author has also been supported in part by BSF Grant 2017684. Work on this paper by the third author has been supported in part by NSF Grant 1815073.

**Acknowledgements** The authors would like to thank Günter Rote and Vuong Bui for many helpful comments on a preliminary draft of this paper.

## 1 Introduction

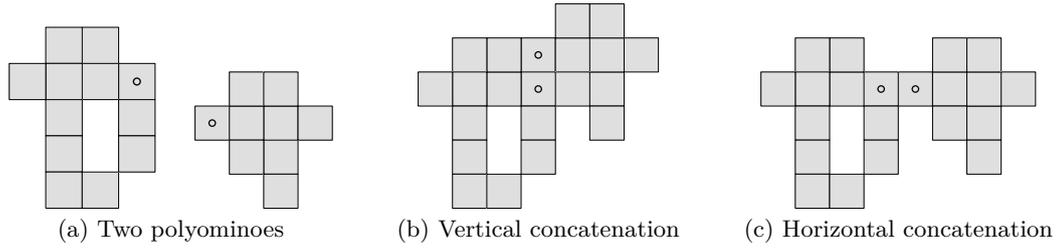
A *polycube* of size  $n$  is a connected set of  $n$  cells on  $\mathbb{Z}^d$ , where connectivity is through  $(d-1)$ -dimensional ( $(d-1)$ -D, in short) facets. 2D polycubes are also called *polyominoes*. The study of polycubes began in statistical physics [3, 9]. Let  $A_d(n)$  count  $d$ -D polycubes of size  $n$ . Klarner [6] showed that  $\lambda_d := \lim_{n \rightarrow \infty} \sqrt[n]{A_d(n)}$  exists. Later, Madras [7] proved that the growth constant  $\frac{A_d(n+1)}{A_d(n)}$  converges to  $\lambda_d$  as  $n \rightarrow \infty$ . The best known lower [2] and upper [1] bounds on  $\lambda_2$  are 4.0025 and 4.5252, resp. We derive bounds on the growth constants of families of polycubes whose enumerating sequences are pseudo sub- or super-multiplicative.

## 2 Preliminaries

### 2.1 Concatenation and Sub-/Super-multiplicative Sequences

A sequence  $(Z(n))$  is *super-multiplicative* (resp., *sub-multiplicative*) if  $Z(m)Z(n) \leq Z(m+n)$  (resp.,  $Z(m)Z(n) \geq Z(m+n)$ )  $\forall m, n \in \mathbb{N}$ . It is known [8] that a super-multiplicative (resp., sub-multiplicative) sequence  $Z(n)$ , with the property that  $Z'(n) = \sqrt[n]{Z(n)}$  is bounded from above (resp., below), has a *growth constant*,  $\lambda_Z$ . That is, the quantity  $\lim_{n \rightarrow \infty} Z'(n)$  exists.

Define a total order on cells of the cubical lattice: first by  $x_1$  ( $x$  in 2D), then by  $x_2$  ( $y$  in 2D), and so on. In 2D, the vertical (resp., horizontal) *concatenation* of two polyominoes  $P_1, P_2$  is the positioning of  $P_2$  such that its smallest cell lies immediately *above* (resp., *to the right of*) the largest cell of  $P_1$  (see Figure 1). Similarly, two  $d$ -D polycubes can be concatenated in  $d$  ways. Concatenating two polycubes always yields a valid polycube, and different pairs of polycubes of sizes  $m, n$  always yield by concatenation different polycubes of size  $m+n$ .



■ **Figure 1** Concatenations of two polyominoes.

A folklore concatenation argument shows that  $A_2^2(n) < A_2(2n)$ , i.e.,  $\sqrt[n]{A_2(n)} < \sqrt[2n]{A_2(2n)}$ . Hence, the sequence  $A^* = \left( \sqrt[n_0 2^i]{A_2(n_0 2^i)} \right)_{i=0}^{\infty}$  is monotone increasing  $\forall n_0 \in \mathbb{N}$ . Since  $(A_2(n))$  is super-multiplicative, and  $A' = \left( \sqrt[n]{A_2(n)} \right)$  is bounded from above [4],  $A_2(n)$  has a growth constant  $\lambda_2$ . Since any subsequence of  $A'$  also converges to  $\lambda_2$ , and any such subsequence  $A^*$  is monotone increasing, any element of it,  $\sqrt[n_0]{A_2(n_0)}$ , is a lower bound on  $\lambda_2$ . Empirically, the tightest lower bound is set by  $A_2(56)$ , the largest known term of  $A_2(n)$  [5]. This is probably since  $A'$  is monotone increasing, but this was never proved. This type of argument holds for any super- or sub-multiplicative sequence, and we develop it further in the sequel.

## 2.2 Pseudo Super- and Sub-Multiplicativity

A sequence  $(Z(n))$  is *pseudo super-multiplicative* (resp., *sub-multiplicative*) if for all  $m, n \in \mathbb{N}$  and some positive sub-exponential function  $P(\cdot)$ , we have that  $P(m+n)Z(m)Z(n) \leq Z(m+n)$  (resp.,  $Z(m)Z(n) \geq P(m+n)Z(m+n)$ ). (Hereafter, we consider cases where  $P(\cdot)$  is a polynomial.) Using  $\lim_{n \rightarrow \infty} \sqrt[n]{P(n)} = 1$  and known values of  $Z(n)$ , we obtain bounds on  $\lambda_Z$ .

► **Theorem 1.** Assume that for a sequence  $(Z(n))$ , the limit  $\mu := \lim_{n \rightarrow \infty} \sqrt[n]{Z(n)}$  exists. Let  $c_i$  ( $c_1 \neq 0$ ) be some constants, and  $\diamond \in \{\leq, \geq\}$ . Then:

- (a) (multiplicative polynomial) If  $c_1 n^{c_2} Z^2(n) \diamond Z(2n) \forall n \in \mathbb{N}$ , then  $\sqrt[n]{c_1 (2n)^{c_2} Z(n)} \diamond \mu \forall n \in \mathbb{N}$ .
- (b) (index shift) If  $c_1 Z^2(n + c_3) \diamond Z(2n) \forall n \in \mathbb{N}$ , then  $\sqrt[n]{c_1 Z(n + 2c_3)} \diamond \mu \forall n \in \mathbb{N}$ .  
Equivalently, if  $c_1 Z^2(n) \diamond Z(2n + c_3) \forall n \in \mathbb{N}$ , then  $\sqrt[n]{c_1 Z(n - c_3)} \diamond \mu \forall n > c_3$ .

The proof is based on variants of the folklore concatenation argument described above.

## 3 Methods of Concatenation

We describe a few concatenation methods that attach polycubes  $P_1, P_2$  through  $P_1$ 's largest cell,  $a_1$ , and  $P_2$ 's smallest cell,  $a_2$ . The sequence is  $Z(n)$ , and its growth constant is  $\lambda_Z$ . Each method provides a relation and, through Thm. 1, a lower bound on  $\lambda_Z$ , see Table 1.

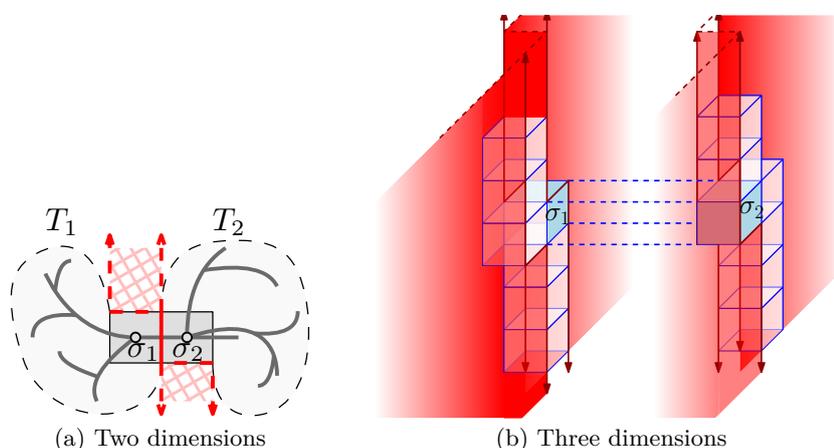
- [E] The most elementary method of concatenation attaches cell  $a_1$  to cell  $a_2$  in a *single* way.
- [C] Attach  $a_1$  to  $a_2$  in all  $c$  (lattice dependent) ways, maintaining that  $a_1$  is smaller than  $a_2$ .
- [M] Attach  $a_1$  and  $a_2$  indirectly, by Method [C] on both sides of any  $k$ -sized polycube.
- [O] *Overlap* cells  $a_1$  and  $a_2$ .
- [MO] Attach  $a_1$  and  $a_2$  indirectly, by Method [O] on both sides of any  $k$ -sized polycubes.

■ **Table 1** Relations resulting from each method, and lower bounds on  $\lambda_Z$  obtained by Theorem 1.

| Method | Relation                               | Lower Bound on $\lambda_Z$                  | Similar To |
|--------|--|---|------------|
| [E]    | $Z^2(n) \leq Z(2n)$                    | $\sqrt[n]{Z(n)} \leq \lambda_Z$             | —          |
| [C]    | $c \cdot Z^2(n) \leq Z(2n)$            | $\sqrt[n]{c \cdot Z(n)} \leq \lambda_Z$     | —          |
| [M]    | $Z(k) \cdot Z^2(n) \leq Z(2n + k)$     | $\sqrt[n]{c^2 \cdot Z(n)} \leq \lambda_Z$   | [C]        |
| [O]    | $Z^2(n) \leq Z(2n - 1)$                | $\sqrt[n]{Z(n + 1)} \leq \lambda_Z$         | —          |
| [MO]   | $Z(k) \cdot Z^2(n) \leq Z(2n + k - 2)$ | $\sqrt[n]{Z(k)Z(n - k + 2)} \leq \lambda_Z$ | [O]        |

## 4 Applications

**General.** We applied methods [E,C,O1] to polycubes, and found lower bounds on  $\lambda_d$  ( $d \leq 9$ ), see Table 3. (In 2D, the bounds are inferior to the bound obtained by the stronger *twisted cylinders* method [2].) In Section 5 we improve all known bounds in  $d \geq 3$  dimensions.



■ **Figure 2** Concatenating trees.

**Trees.** Figs. 2(a,b) show two tree polycubes, concatenated by Method [E]. The only valid option is to align cells  $\sigma_1, \sigma_2$  with the most dominant axis of the lexicographic order, ensuring a buffer space for avoiding cycles. Let  $A_{d;T}(n)$  denote the number of  $n$ -cell tree polycubes in  $d$ -D, thus  $A_{d;T}^2(n) \leq A_{d;T}(2n)$ , and, by Thm. 1(b),  $\lambda_{d;T} \geq \sqrt[n]{A_{d;T}(n)} \forall n \in \mathbb{N}$ . See Table 2.

**Convex polyominoes.** We show matching lower/upper bounds on their growth constant.

■ **Table 2** Lower bounds on the growth constants of tree polycubes of various dimensions.

| Dimensions | Known Values | OEIS Sequence | Method [E] |
|------------|--------------|---------------|------------|
| 2          | 44           | A066158       | 3.4045     |
| 3          | 17           | A118356       | 5.5592     |
| 4          | 10           | A191094       | 6.7698     |
| 5          | 10           | A191095       | 8.8035     |
| 6          | 8            | A191096       | 9.4576     |
| 7          | 7            | A191097       | 10.0909    |
| 8          | 7            | A191098       | 11.4891    |

■ **Table 3** Lower bounds on  $\lambda_d$ , through each method. (Best previously-published bounds are underlined, our improved bounds appear in bold.)

| Dimensions | Known Values | OEIS Sequence | Concatenation Methods |         |         | Other Methods      | Recursive Bounding |
|------------|--------------|---------------|-----------------------|---------|---------|--------------------|--------------------|
|            |              |               | [E]                   | [C]     | [O]     |                    |                    |
| 2          | 56           | A001168       | 3.7031                | 3.7492  | 3.7923  | <u>4.00253</u> [2] | 3.7944             |
| 3          | 19           | A001931       | <u>6.0211</u>         | 6.3795  | 6.6526  | —                  | <b>6.6621</b>      |
| 4          | 16           | A151830       | <u>8.4627</u>         | 9.2286  | 9.7576  | —                  | <b>9.7714</b>      |
| 5          | 15           | A151831       | <u>10.9093</u>        | 12.1449 | 12.9398 | —                  | <b>12.9569</b>     |
| 6          | 15           | A151832       | <u>13.5237</u>        | 15.2396 | 16.2888 | —                  | <b>16.3087</b>     |
| 7          | 14           | A151833       | <u>15.5985</u>        | 17.9245 | 19.2690 | —                  | <b>19.2927</b>     |
| 8          | 12           | A151834       | <u>16.6477</u>        | 19.7976 | 21.4975 | —                  | <b>21.5298</b>     |
| 9          | 12           | A151835       | <u>18.8417</u>        | 22.6277 | 24.6060 | —                  | <b>24.6416</b>     |

## 5 Recursive Bounding

We now present a recursive scheme for improving bounds obtained by all methods described above. Let us demonstrate the scheme by a concrete example. As observed earlier, the sequence enumerating  $d$ -D polycubes is super-multiplicative and it has a growth constant, hence, by Method [O], any term of the form  $n^{-1}\sqrt[A_d(n)]{n}$  is a lower bound on  $\lambda_d$ . We can prove relations which are tighter than the super-multiplicativity condition, for example:

► **Theorem 2.** Let  $h = \lfloor (n+1)/2 \rfloor$ . Then, for every  $n \geq 4$ , we have that

$$A_d(n) \geq A_d(h)A_d(n-h+1) + \frac{d(d-1)^2}{2}(A_d(h-1)A_d(n-h-1) + A_d(h-2)A_d(n-h)). \quad (1)$$

Rel. (1) does not yield “chains” of bounds, but we can recursively bound values of  $A_d(n)$ . Knowing  $A_d(n) \forall n \leq n_0$ , we construct  $B(n)$ , s.t.  $B(n) \leq A_d(n) \forall n$ : for  $1 \leq n \leq n_0$ ,  $B(n) = A_d(n)$ ; and for  $n > n_0$ , set  $B(n)$  recursively to the value obtained by Rel. (1). We ran this method until our resources were exhausted (around  $n \approx 12M$ ) and chose the best value encountered. This procedure improved the lower bounds on  $\lambda_d$  in  $3 \leq d \leq 9$  dimensions; see Table 3.

---

## References

- 1 G. BAREQUET AND M. SHALAH, Improved upper bounds on the growth constants of polyominoes and polycubes, *Proc. 14th Latin American Theoretical Informatics Symposium*, São Paulo, Brazil, *Lecture Notes in Computer Science*, #, Springer, May 2020, to appear.
- 2 G. BAREQUET, M. SHALAH, AND G. ROTE,  $\lambda > 4$ : An improved lower bound on the growth constant of polyominoes, *Comm. of the ACM*, 59 (2016), 88–95.
- 3 S.R. BROADBENT AND J.M. HAMMERSLEY, Percolation processes: I. Crystals and mazes, *Proc. Cambridge Philosophical Society*, 53 (1957), 629–641.
- 4 M. EDEN, A two-dimensional growth process, *Proc. 4th Berkeley Symp. on Mathematical Statistics and Probability*, IV, Berkeley, CA, 223–239, 1961.
- 5 I. JENSEN, Counting polyominoes: A parallel implementation for cluster computing, *Proc. Int. Conf. on Computational Science*, III (Melbourne, Australia and St. Petersburg, Russia, 2003), *Lecture Notes in Computer Science*, 2659, Springer, 203–212.
- 6 D.A. KLARNER, Cell growth problems, *Canadian J. of Mathematics*, 19 (1967), 851–863.
- 7 N. MADRAS, A pattern theorem for lattice clusters, *Annals of Combinatorics*, 3 (1999), 357–384.
- 8 G. PÓLYA AND G. SZEGO, *Aufgaben und Lehrsätze aus der Analysis*, vol. 1, Julius Springer, Berlin, 1925.
- 9 H.N.V. TEMPERLEY, Combinatorial problems suggested by the statistical mechanics of domains and of rubber-like molecules, *Physical Review* 2, 103 (1956), 1–16.

# On The Number of Compositions of Two Polycubes

**Andrei Asinowski**

Inst. für Mathematik, Alpen-Adria-Universität Klagenfurt, Universitätsstraße 65–67,  
9020 Klagenfurt am Wörthersee, Austria.  
andrei.asinowski@aau.at

**Gill Barequet**

**Gil Ben-Shachar**

Dept. of Computer Science, Technion—Israel Inst. of Technology, Haifa 3200003, Israel  
{barequet,gilbe}@cs.technion.ac.il

**Martha Carolina Osegueda**

Dept. of Computer Science, Univ. of California, Irvine, CA 92717  
mosegued@uci.edu

**Günter Rote**

Inst. für Informatik, Freie Universität Berlin, Takustraße 9, D-14195 Berlin, Germany  
rote@inf.fu-berlin.de

---

## Abstract

We provide tight bounds on the minimum and maximum possible numbers of compositions of two polycubes, either when each is of size  $n$ , or when their total size is  $2n$ , in two and higher dimensions.

**2012 ACM Subject Classification** Mathematics of computing → Combinatoric problems

**Keywords and phrases** Polyominoes, Polycubes

**Funding** Work on this paper by the first author has been supported in part by the Austrian Science Fund (FWF), in the framework Project 28466-N35. Work on this paper by the second and third authors has been supported in part by ISF Grant 575/15. Work on this paper by the second author has also been supported in part by BSF Grant 2017684. Work on this paper by the fourth author has been supported in part by NSF Grant 1815073.

## 1 Introduction

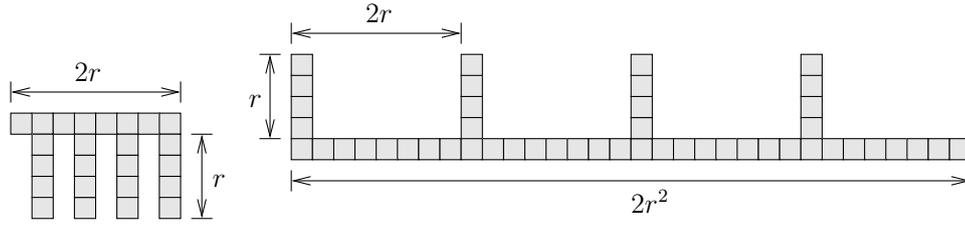
A  $d$ -dimensional ( $d$ -D, in short) *polycube* (*polyomino* in 2D) is a connected set of cells on  $\mathbb{Z}^d$ , where connectivity is through  $(d-1)$ -D faces. Shapes on the cubical and other lattices play an important role in mathematics [3] and physics [2]. The *size* of a polycube is the number of its  $d$ -D cells. A *composition* of two  $d$ -D polycubes is their relative placement, s.t. they touch each other through  $(d-1)$ -D faces but do not overlap. The number of compositions of polycubes is used for proving bounds on their growth constant [1]. Our main question is:

Given two polycubes of **total size  $2n$** , how many compositions do they have?

Table 1 summarizes our results.

■ **Table 1** The number of compositions of two polycubes of total size  $2n$ .

| Num. of Compositions | Two Dimensions                 | $d > 2$ Dimensions                       |
|----------------------|--------------------------------|--|
| Minimum              | $\Theta(n^{1/2})$              | $\Omega(n^{1-1/d}/d), O(2^d dn^{1-1/d})$ |
| Maximum              | $\Omega(n^2/\log^2 n), O(n^2)$ | $\Theta(dn^2)$                           |



■ **Figure 1** A pair of 1-level combs.

## 2 Two Dimensions

### 2.1 Minimum Number of Compositions

► **Theorem 1.** *All pairs of polyominoes of total size  $2n$  have  $\Omega(n^{1/2})$  compositions. There exist pairs of polyominoes of total size  $2n$  with  $\Theta(n^{1/2})$  compositions.*

**Proof.** Consider two polyominoes  $P_1, P_2$  of total size  $2n$ . Assume w.l.o.g. that  $|P_1| \geq n$ . Assume, also w.l.o.g., that the width of  $P_1$  is greater than (or equal to) its height. Hence, the width of  $P_1$  is at least  $n^{1/2}$ . Then,  $P_2$  may touch  $P_1$  from below or above in at least  $2n^{1/2}$  ways: Put  $P_2$  below (or above)  $P_1$  so that the left column of  $P_2$  is aligned with the  $i$ th column of  $P_1$  ( $1 \leq i \leq n^{1/2}$ ) and translate  $P_2$  upward (or downward) until it touches  $P_1$ .

The lower bound on the minimum case is tight. Assume that  $n$  is a square integer number and consider two square polyominoes with side-length  $k = n^{1/2}$ . The total size of the two polyominoes is  $2n$ , and they can be composed in  $4(2k - 1) = 4(2n^{1/2} - 1) < 8n^{1/2}$  ways. ◀

### 2.2 Maximum Number of Compositions

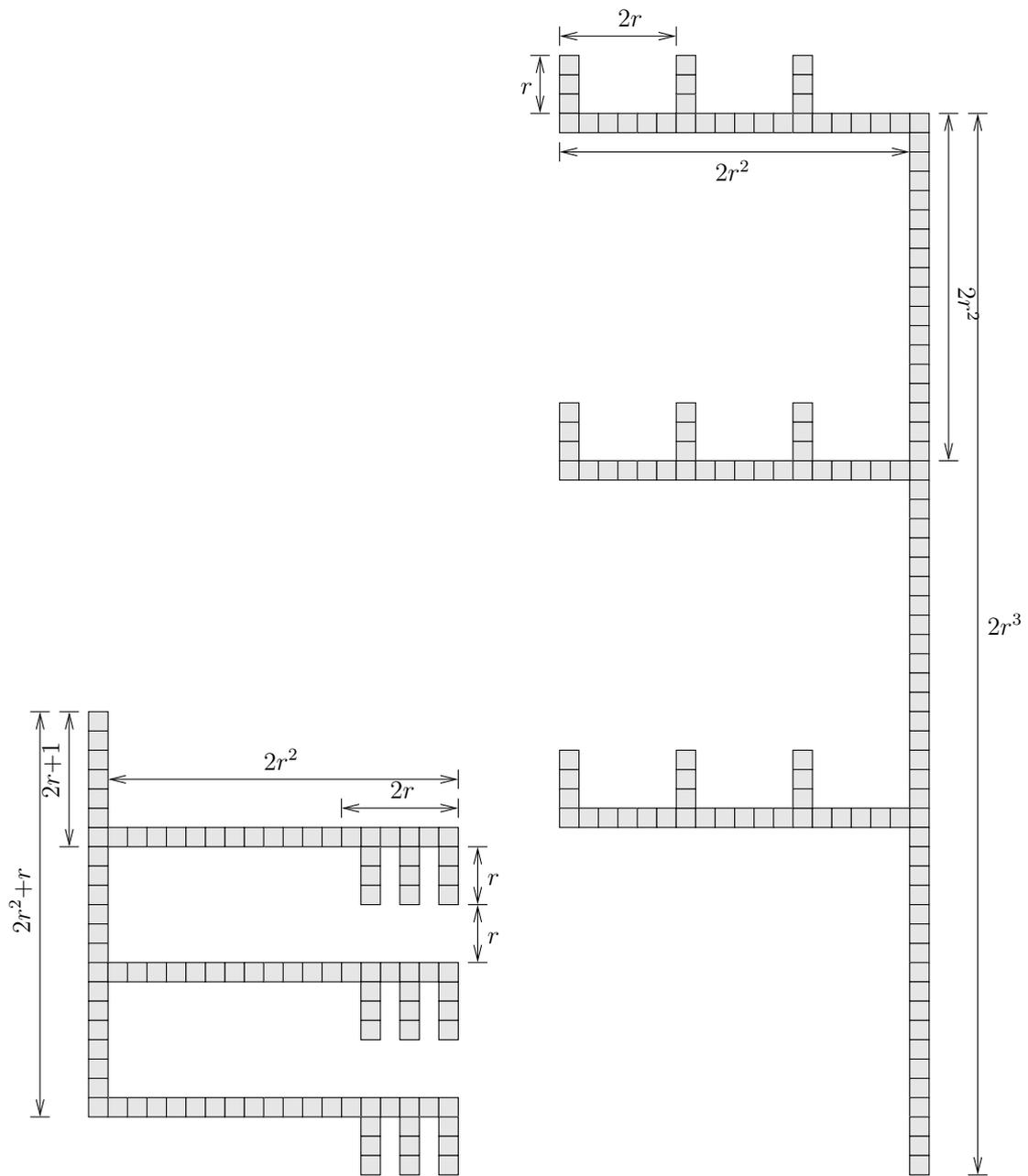
► **Theorem 2.** *There exist pairs of polyominoes, of total size  $2n$ , with  $\Omega(\frac{n^2}{\log^2 n})$  compositions.*

**Proof.** (sketch) We construct a pair of two multiple-level “combs.” Figure 1 shows a pair of 1-level combs of total size  $\Theta(r^2)$  and  $\Theta(r^3)$  compositions. Setting  $n = r^2$  and adjusting the constants, we obtain a pair of polyominoes of total size  $2n$  and  $\Theta(n^{3/2})$  compositions. This construction continues iteratively. In each level, we add to the combs one level of teeth. Figure 2 shows a pair of 2-level combs of total size  $\Theta(r^3)$  and  $r^5$  compositions. Setting  $n = r^3$  and adjusting the constants, we obtain a pair of polyominoes of total size  $2n$  and  $\Theta(n^{5/3})$  compositions. Analysis of  $k$ -level combs shows that they have  $\Theta(n^{2-1/(k+1)})$  compositions, implying the lower bound  $\Omega(n^{2-\varepsilon})$  (for any  $\varepsilon > 0$ ) on the maximum number of compositions for two polyominoes of total size  $2n$ . However, we can do even better than that. Instead of fixing  $k$  and then letting  $n \rightarrow \infty$ , we do the opposite: For a given  $n$ , we set  $k$  s.t.  $n = (k+2)3^k$ . Careful calculation shows that the total number of compositions is  $\Omega\left(\frac{n^2}{\log^2 n}\right)$ .<sup>1</sup> ◀

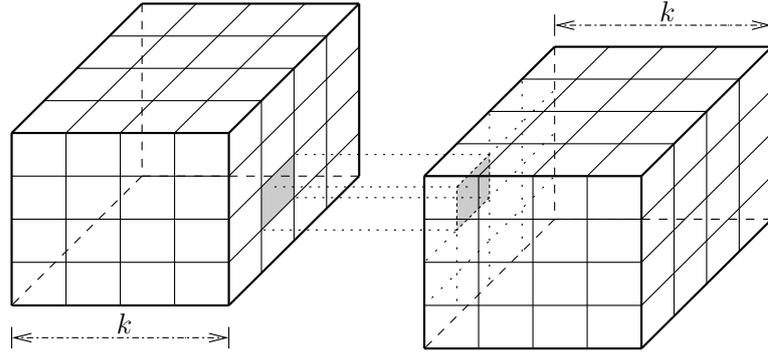
► **Theorem 3.** *Any pair of polyominoes of total size  $2n$  has  $O(n^2)$  compositions.*

**Proof.** Any two polyominoes  $P_1, P_2$  of total size  $2n$  have  $O(n^2)$  compositions. Indeed, let  $n_1 = |P_1|$  and  $n_2 = |P_2|$ , where  $n_1 + n_2 = 2n$ . Then, every cell of  $P_1$  can touch every cell of  $P_2$  in  $\leq 4$  ways, yielding  $4n_1n_2 \leq 4n^2$  as an upper bound on the number of compositions. ◀

<sup>1</sup> In fact, we believe that with a more complex argument, we can improve the lower bound to  $\Omega\left(\frac{n^2}{2^{8 \cdot \sqrt{\log_2 n}}}\right)$ .



■ **Figure 2** A pair of 2-level combs.



■ **Figure 3** Compositions two “megacubes.”

### 3 Higher Dimensions

#### 3.1 Minimum Number of Compositions

► **Theorem 4.** *All pairs of  $d$ -D polycubes of total size  $2n$  have  $\Omega(n^{1-1/d}/d)$  compositions.*

**Proof.** The proof is similar to that of Thm. 1. Consider a pair of polycubes  $P_1, P_2$  of total size  $2n$ . Assume, w.l.o.g., that  $|P_1| \geq n$ . The size of the boundary of  $P_1$  is  $\Omega(n^{1-1/d})$ . Let  $i$  ( $1 \leq i \leq d$ ) be the dimension along which the volume of the projection of  $P_1$  is the largest. This projection is made of  $b = \Omega(n^{1-1/d}/d)$  faces orthogonal to  $x_i$ . Then,  $P_2$  may touch  $P_1$  in at least  $b$  ways: Put  $P_2$  “below”  $P_1$  (along  $x_i$ ), so that one specific ( $x_i$ -parallel) “column” of  $P_2$  is aligned with the  $k$ th column of  $P_1$  ( $1 \leq k \leq b$ ), and shift  $P_2$  towards  $P_1$  until they touch. ◀

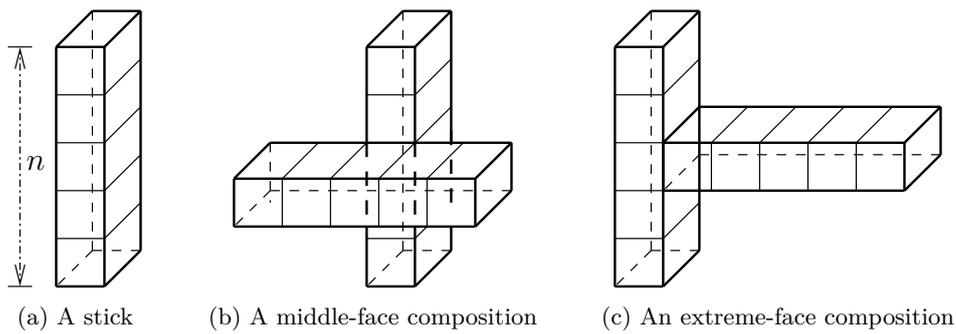
► **Theorem 5.** *There exist pairs of  $n$ -cell  $d$ -D polycubes, having  $O(2^d dn^{1-1/d})$  compositions.*

**Proof.** Fig. 3(a) shows a composition of two copies of an  $n$ -cell  $d$ -dimensional “megacube”  $P$ , whose sidelength is  $k = n^{1/d}$ . Two copies of  $P$  can slide towards each other in  $2d$  directions until they touch. There are no other compositions since no megacube can penetrate into the other. Once we decide which “mega faces” touch each other, this can be done in  $(2k-1)^{d-1}$  ways. Indeed, in each of the  $d-1$  dimensions orthogonal to the sliding direction, there are  $2k-1$  possible offsets of one megacube relative to the other. The total number of compositions in this example is  $(2d)(2k-1)^{d-1} = 2d(2n^{1/d} - 1)^{d-1} = \Theta(2^d dn^{1-1/d})$ . ◀

#### 3.2 Maximum Number of Compositions

► **Theorem 6.** *All pairs of  $d$ -dimensional polycubes of total size  $2n$  have  $O(dn^2)$  compositions. For  $d \geq 3$ , the upper bound is tight.*

**Proof.** Similarly to 2D, any pair of polycubes  $P_1, P_2$  of total size  $2n$  have  $O(dn^2)$  compositions. Indeed, let  $|P_1|=n_1$  and  $|P_2|=n_2$ , where  $n_1+n_2=2n$ . Every cell of  $P_1$  can touch every cell of  $P_2$  in at most  $2d$  ways, resulting in at most  $2dn_1n_2 \leq 2dn^2$  compositions. The upper bound is attainable. Consider two nonparallel “sticks,” each of size  $n$ , see Fig. 4(a). Each stick has two extreme  $(d-1)$ -D faces and  $2(d-1)n$  middle faces of dimension  $d-1$ . First, the number of compositions that involve only the intermediate faces is  $2(d-2)n^2$  (Fig. 4(b)). Second, the number of compositions that involve an extreme face of one stick and an intermediate face of the other stick is  $4n$  (Fig. 4(c)). All compositions of the two types are different by construction. Overall, the two sticks have  $2(d-2)n^2 + 4n = \Theta(dn^2)$  compositions. ◀



■ **Figure 4** Compositions of two “sticks.”

Note the difference between two and higher dimensions. In  $d > 2$ , the compositions are restricted by the dimensions along which the sticks in the proof of Thm. 6 are aligned, but the other dimensions allow every pair of cells, one of each polycube, to have distinct compositions. This is not the case in 2D, hence the proof of Thm. 2 is much more complex.

---

#### References

- 1 G. BAREQUET, G. ROTE, AND M. SHALAH, An improved upper bound on the growth constant of polyiamonds, *Proc. 10th European Conf. on Combinatorics, Graph Theory, and Applications*, Bratislava, Slovakia, *Acta Mathematica Universitatis Comenianae*, 88, 429–436, August 2019.
- 2 S.R. BROADBENT, J.M. HAMMERSLEY, Percolation processes: I. Crystals and mazes, *Proc. Cambridge Philosophical Society*, **53**, 629–641, 1957.
- 3 S.W. GOLOMB, *Polyominoes*, Princeton University Press, Princeton, NJ, 2nd ed., 1994.

# Practical volume estimation of polytopes by billiard trajectories and a new annealing schedule

**Apostolos Chalkis**

Department of Informatics & Telecommunications  
National & Kapodistrian University of Athens, Greece  
[achalkis@di.uoa.gr](mailto:achalkis@di.uoa.gr)

**Ioannis Z. Emiris**

Department of Informatics & Telecommunications  
National & Kapodistrian University of Athens, and  
Athena Research Innovation Center, Greece  
[emiris@di.uoa.gr](mailto:emiris@di.uoa.gr)

**Vissarion Fisikopoulos**

Department of Informatics & Telecommunications  
National & Kapodistrian University of Athens, Greece  
[vfisikop@di.uoa.gr](mailto:vfisikop@di.uoa.gr)

---

## Abstract

We study the problem of estimating the volume of convex polytopes. Our algorithm is based on Multiphase Monte Carlo (MMC) methods and our main contributions include: (i) a new uniform sampler employing Billiard Walk (BW) for the first time in volume computation, (ii) a new simulated annealing schedule, generalizing existing MMC by making use of adaptive convex bodies which fit to the input, thus drastically reducing the required number of iterations. Extensive experiments show that our algorithm requires fewer oracle calls and arithmetic operations in total, when compared to existing practical algorithms for volume approximation. We offer an open-source, optimized C++ implementation, and analyze its performance. Our code tackles problems intractable so far, offering the first implementation to scale up to dimension  $d = 100$  for V-polytopes and zonotopes, and for  $d$  in the thousands for H-polytopes in  $\leq 2$  hour.

**2012 ACM Subject Classification** Design and analysis of algorithms:  
Computational geometry, Random walks and Markov chains

**Keywords and phrases** Polytope volume, sampling, zonotope, V-polytope, simulated annealing, Billiard walk, algorithm engineering, software, experiments

## 1 Introduction

Volume computation is a fundamental problem with many applications. It is #P-hard for explicit polytopes [5, 9], and APX-hard [7] for convex bodies in the oracle model. Therefore, a significant effort has been devoted to randomized approximation algorithms, starting with the celebrated result in [6] with complexity  $O^*(d^{23})$  oracle calls, where  $O^*(\cdot)$  suppresses polylog factors and dependence on error parameters, and  $d$  is the dimension. Improved algorithms reduced the exponent to 5 [10] and further results [3, 11] reduced it to 3. Regarding implementations, the method of [10] led to the first practical implementation in [8] for high dimensions, followed by another practical implementation [4] based on [3, 11]. However, both implementations can handle only H-polytopes.

A typical randomized algorithm uses a Multiphase Monte Carlo (MMC) technique, which reduces volume approximation of convex  $P$  to computing a telescoping product of ratios of integrals. Then each ratio is estimated by means of random walks sampling from a proper multivariate distribution. MMC in [10] specifies a sequence of convex bodies  $P_m \subseteq \dots \subseteq P_0 = P$ , assuming  $rB_d \subseteq P \subseteq RB_d$ , where  $B_d$  is the unit ball for  $m > 1$  and real

$r, R$ . One defines a sequence of convex bodies  $P_i = (2^{(m-i)/d}rB_d) \cap P$ ,  $i = 0, \dots, m$ . Then,

$$\text{vol}(P) = \text{vol}(P_m) \frac{\text{vol}(P_{m-1})}{\text{vol}(P_m)} \dots \frac{\text{vol}(P_0)}{\text{vol}(P_1)}, \quad m = \lceil d \lg(R/r) \rceil, P_0 = RB_d \cap P. \quad (1)$$

To compute  $\text{vol}(P_m) = \text{vol}(rB_d)$  there is a closed-form expression. Each ratio  $r_i = \text{vol}(P_{i+1})/\text{vol}(P_i)$  in Eq. (1) can be estimated within arbitrary small error  $\epsilon_i$  by sampling uniformly distributed points from  $P_i$  and accept / reject points in  $P_{i+1}$ . The issue is to minimize  $m$  while each ratio remains bounded by a constant, and to use a random walk that converges after a minimum number of steps to the uniform distribution. The first would permit a larger approximation error per ratio without compromising overall error, while it would require a smaller uniform sample to estimate each ratio. The second would reduce the cost per point. The most studied representations of convex polytopes in the literature are H-polytopes (sets of linear inequalities), V-polytopes (convex hulls of pointsets), and zonotopes or Z-polytopes (Minkowski sums of segments). A full version of the paper is given in [2]. However, the current abstract presents new results in progress.

## 2 Algorithm

Our algorithm relies on MMC of Eq. (1) and introduces further algorithmic innovations. We define a new simulated annealing that specifies the  $P_i$ 's by exploiting the statistical properties of the telescoping ratios to drastically reduce the number of phases. In particular, we bound each ratio  $r_i = \text{vol}(P_{i+1})/\text{vol}(P_i)$  to a given interval  $[r, r + \delta]$  with high probability. Moreover, our MMC generalizes balls, used in [10] and previous papers, by taking as input any convex body  $C$  and constructing the sequence by only scaling  $C$ .

We prove that the number  $m$  of bodies defined in MMC is inversely proportional to  $\text{vol}(P_m)$ , i.e. the volume of the body with minimum volume in MMC. The bound we derive on  $m$  is not surprising, as it does not improve worst-case complexity [3], if  $C$  is a ball, but offers crucial advantages in practice. First, the hidden constant is small. More importantly, if  $C$  fits to  $P$ , then  $\text{vol}(P_m)$  increases and  $m$  decreases. The latter property allows us to skip rounding for Z-polytopes which is costly as it requires uniform sampling: in particular, we let  $C$  be a centrally symmetric H-polytope which fits to  $P$ . Our approach is to construct  $C$  fast and reduce  $m$  and the total runtime more than a rounding preprocessing would do in practice. We also show that, for constant  $d$ , and  $k$  (number of segments) increasing,  $m$  decreases to 1, when we use ball in MMC without rounding, since our algorithm constructs an enclosing ball of  $P$ . Intuitively, this can be explained by the main result in [1]. Summarizing, in our experiments, for Z-polytopes, the number of phases is  $m \leq 3$  for any order (i.e.  $k/d$ ), without rounding for  $d \leq 100$ . For H-polytopes we use the rounding method in [8] and for V-polytopes an improved version of the same method, exploiting the vertices of  $P$ .

For uniform sampling, Billiard walk (BW) [12] defines a linear trajectory starting at the current point, using boundary reflections. No theoretical mixing time exists. We experimentally derive that, with an appropriate choice of parameters, BW behaves like an almost perfect uniform sampler even if the walk length is 1. Interestingly, for this walk length, it generates just  $O^*(1)$  points per phase, and provides the desired accuracy, when we set an upper bound of  $O(d)$  to the number of reflections per generated point. We experimentally analyze complexity: our algorithm performs fewer oracle calls and fewer arithmetic operations than current practical algorithms (Table 1). This leads to the first implementation that handles efficiently 3 polytope representations and scales up to  $d = 1000$  for H-polytopes and  $d = 100$  for V- and Z-polytopes. Our software contributions build upon and enhance

|            | unit cube / randH |           | rand V-    |          | rand Z-    |          |
|------------|-------------------|-----------|------------|----------|------------|----------|
|            | steps             | cost/step | BOC        | cost/BOC | BOC        | cost/BOC |
| [8]        | $O^*(d^3)$        | $O(m)$    | $O^*(d^3)$ | 2 LP     | $O^*(d^3)$ | 2 LP     |
| [4]        | $O^*(d^{2.5})$    | $O(m)$    | --         | 2 LP     | --         | 2 LP     |
| this paper | $O^*(d^2)$        | $O(dm)$   | $O^*(d)$   | 1 LP     | $o^*(d)$   | 1 LP     |

■ **Table 1** For unit cube and random-H: #steps (points generated), cost/step,  $m = \#$ facets. For random V- and Z-polytopes: #boundary oracle calls (BOC), cost/BOC for each algorithm.

`volesti`<sup>1</sup> a C++ open source library for high dimensional sampling and volume computation with an R interface. This enables us to analyze the complexity of our algorithm.

---

## References

- 1 J. Bourgain and J. Lindenstrauss. Approximating the ball by a Minkowski sum of segments with equal length. *Discr. & Comput. Geom.*, 9:131–144, 1993.
- 2 A. Chalkis, I.Z. Emiris, and V. Fisikopoulos. Practical volume estimation by a new annealing schedule for cooling convex bodies. *CoRR*, abs/1905.05494, 2019.
- 3 B. Cousins and S. Vempala. Bypassing KLS: Gaussian cooling and an  $O^*(n^3)$  volume algorithm. In *Proc. ACM STOC*, pages 539–548, 2015.
- 4 B. Cousins and S. Vempala. A practical volume algorithm. *Mathematical Programming Computation*, 8, 2016.
- 5 M. Dyer and A. Frieze. On the complexity of computing the volume of a polyhedron. *SIAM J. Computing*, 17(5):967–974, 1988.
- 6 M. Dyer, A. Frieze, and R. Kannan. A random polynomial-time algorithm for approximating the volume of convex bodies. *J. ACM*, 38(1):1–17, 1991.
- 7 G. Elekes. A geometric inequality and the complexity of computing volume. *Discr. Comput. Geom.*, 1(4):289–292, 1986.
- 8 I.Z. Emiris and V. Fisikopoulos. Practical polytope volume approximation. *ACM Trans. Math. Soft.*, 44(4):38:1–38:21, 2018. Prelim. version: Proc. SoCG 2014.
- 9 E. Gover and N. Krikorian. Determinants and the volumes of parallelotopes and zonotopes. *Linear Algebra & Appl.*, 413:28–40, 2010.
- 10 L. Lovász, R. Kannan, and M. Simonovits. Random walks and an  $O^*(n^5)$  volume algorithm for convex bodies. *Random Structures and Algorithms*, 11:1–50, 1997.
- 11 L. Lovász and S. Vempala. Simulated annealing in convex bodies and an  $O^*(n^4)$  volume algorithms. *J. Computer & System Sciences*, 72:392–417, 2006.
- 12 B.T. Polyak and E.N. Gryazina. Billiard walk - a new sampling algorithm for control and optimization. *IFAC Proceedings Volumes*, 47(3):6123 – 6128, 2014. 19th IFAC World Congress.

---

<sup>1</sup> [https://github.com/GeomScale/volume\\_approximation](https://github.com/GeomScale/volume_approximation)

# Samples Using Geometric Counting Lower Bounds

Sepideh Aghamolaei

Sharif University of Technology, Tehran, Iran  
aghamolaei@ce.sharif.edu

Mohammad Ghodsi

Sharif University of Technology, Tehran, Iran  
ghodsi@sharif.edu

---

## Abstract

Using the sampling probability of Szemerédi–Trotter theorem, we give a sampling algorithm for counting the number of incidences between  $m$  lines and  $n$  points on those lines, with high probability. We give a similar proof and sampling algorithm for  $k$ -center based on the dominating set.

**2012 ACM Subject Classification** Theory of computation → Generating random combinatorial structures; Theory of computation → Computational geometry

**Keywords and phrases** Incidences, Sampling, Dominating Set,  $k$ -center

## 1 Introduction

We give sampling algorithms based on the probabilistic proofs of the incidence counting problem and  $k$ -center. Given  $m$  distinct lines and  $n$  distinct points on them, the number of incidences  $I(n, m)$  is the number of (point, line) pairs such that the point lies on the line. For a graph  $G$ , we denote this value with  $\phi(G)$ . We give a  $O(1)$ -approximation of  $\phi(G)$  in  $O(\frac{n^2}{m} \log(nm))$ , improving upon the  $O((m^2 + n) \log(n + m))$  time line sweeping algorithm.

► **Theorem 1** (Szemerédi–Trotter [4, 5]).  $I(n, m) \leq O(n^{2/3}m^{2/3} + n + m)$ .

**Proof Sketch.** A simple probabilistic proof [3] uses the sampling method of Algorithm 1. Build the graph  $G = (P, E)$ , where  $E$  is all pairs of consecutive points on a line in  $L$ . So,  $|E| = \phi(G) - |L|$ . The crossing inequality states  $\text{cr}(G) \geq \frac{|E|^3}{64|P|^2}$  for  $|E| > 4|P|$ , where  $\text{cr}(G)$

■ **Algorithm 1** Sampling for counting the number of incidences of points  $P$  and lines  $L$

- 
- 1:  $P_S =$  Sample each point of  $P$  with probability  $p = \frac{4n}{m}$ .
  - 2:  $L_S =$  Keep a line of  $L$  if two points on this line are in  $P_S$ .
  - 3: Report the number of incidences between  $P_S$  and  $L_S$ .
- 

is the number of crossings of  $G$ . The probability that maximizes the number of crossings is:  $\text{cr}(G)p^4 \geq |E|p^2 - |P|p \Rightarrow p = \frac{4|P|}{|E|}$ . So,  $p = \frac{4n}{\phi(G)-m} \geq \frac{4n}{m}$  gives the expected bound. ◀

A *geometric dual* converts a line  $l$  to a point  $l^*$  and a point  $p$  to a line  $p^*$  [1]:

$$p = (p_x, p_y) \rightarrow p^* : y = p_x x - p_y, \quad l : y = mx + c \rightarrow l^* = (m, -c).$$

This transformation preserves point-line intersections.

The *dominating set* of a graph is the smallest subset  $I$  of vertices such that each vertex is either in  $I$  or adjacent to a vertex in  $I$ .

Given a set of  $n$  points in a metric space with distance function  $d$  and an integer  $k$ , the  $k$ -center problem asks for a subset of  $k$  input points called centers such that the distance from each input point to their nearest center is at most  $r$ , and  $r$  is minimized. Equivalently, the  $k$ -center problem is the dominating set of the disk graph of radius  $r$ , where the disk graph is the intersection graph of disks of radius  $r$  centered at input points. Each edge of an intersection graph corresponds to an intersection and each vertex corresponds to a shape.

## 2 Bounding the Number of Incidences Using Duality

In Theorem 1,  $p = \frac{4n}{m} \leq 1$ . For  $\frac{4n}{m} > 1$ , we use the duals of  $P$  and  $L$  using Lemmas 2 and 3.

► **Lemma 2.**  $I(n, m) = I(m, n)$  and  $\phi(P, L) = \phi(L^*, P^*)$ , where  $*$  shows the geometric dual.

**Proof.** Take the geometric dual of the points and the lines. The incidence counting problem of  $n$  points and  $m$  lines after taking its dual becomes the incidence counting problem of  $m$  points and  $n$  lines. Since point-line intersections are preserved, their count is also preserved. ◀

► **Lemma 3.** In a drawing  $G$  of  $m$  lines and  $n$  points on them,  $cr(G) \geq \frac{n^3}{64m^2}$  for  $4n > m$ .

**Proof.** After taking the dual of the input points and lines, the resulting graph  $G'$  satisfies the condition  $|E| \geq 4|V|$ , so the crossing inequality holds for  $G'$  as well. ◀

■ **Algorithm 2** Sampling for counting the number of incidences of points  $P_X$  and lines  $L_X$

- 
- 1:  $T = \text{Add } 2 \text{ random points on each line } L_X$ .
  - 2:  $P'_X = P_X \cup T$
  - 3: Compute  $\phi(P'_X, L_X)$  using Algorithm 1.
  - 4: Return  $\phi'(P_X, L_X) = \phi(P'_X, L_X) - 2|L_X|p$ .
- 

Algorithm 2 guarantees the 2nd step of Algorithm 1 does not remove incidences of lines with a single point on them. The probability of these new random points colliding or falling on an intersection is 0, and the number of changes to  $\phi(G)$  is  $2mp$ , w.h.p..

► **Theorem 4.**  $\max_X \phi(P_X, L_X)$  over  $\theta(\log(nm))$  random samples  $X$  using Algorithm 2 is a  $O(1)$ -approximation for the number of incidences, w.h.p. in  $O(\min(\frac{n^4}{m^2}, \frac{m^4}{n^2}) \log(nm))$  time.

**Proof.** Using  $p = \frac{4n}{m}$ , the expected number of vertices of the sample is  $np = \frac{4n^2}{m}$  and the expected number of edges is  $mp^2 = \frac{16n^2}{m}$ . The size of the sample is therefore  $np + mp^2 = O(\frac{n^2}{m})$  and the time complexity of counting the number of incidences is  $O(\frac{n^4}{m^2})$ .

Since  $G_S$  is a subgraph of  $G$ , we have  $\phi(G_S) \leq \phi(G)$ . Using Chebyshev's inequality and  $\text{Var}(\phi(G_S)) = \phi(G) \cdot p(1-p)$ , we get  $pr(|\phi(G_S) - \phi(G)p^4| \geq \alpha \cdot \phi(G)p^4) \leq \frac{\phi(G)^2 p^2 (1-p)^2}{\alpha^2 \phi(G)^2 p^8} = \frac{(1-p)^2}{\alpha^2 p^6}$ . So, the probability of a  $c$ -approximation is at least  $1 - \frac{(1-p)^2 p^2}{c^2} \geq \frac{15}{16}$ , for  $c = \alpha p^4$ .  $\max_S \phi(G_S)$  for  $\log \phi(G) = O(\log nm)$  independent samples  $S$  is a  $c$ -approximation for  $\phi(G)$  w.h.p.. The total time complexity is  $O(\frac{n^4}{m^2} \log(nm))$ ; Taking the minimum time complexity of this instance and its dual gives the better bound  $O(\min(\frac{n^4}{m^2} \log(nm), \frac{m^4}{n^2} \log(nm)))$ . ◀

## 3 Dominating Set and $k$ -Center

► **Theorem 5.** A random sample of points with probability  $p = \frac{1}{2k}$  gives a covering with radius  $c$  times the  $k$ -center, w.h.p., for  $k = \Omega(1)$ .

**Proof.** Consider the graph  $G = (V, E)$ , with  $n$  vertices,  $m$  edges, and  $\deg_i$  denoting the degree of each vertex  $v_i$ . From basic graph theory,  $\sum_{i=1}^n \deg_i = 2m$ . A dominating set  $I$  of  $G$  satisfies  $\sum_{i \in I} \deg_i \geq n - |I|$ , since each vertex not in  $I$  is adjacent to at least one vertex in  $I$ . Also,  $\sum_{i \in S} \deg_i \leq 2m$ , for all  $S \subset V$ , including  $S = I$ . Sample the vertices of  $G$  with probability  $p$ , and use the bounds on the sample:

$$\sum_{i \in I} p \sum_{j \in \text{adj}(i)} p \geq (n - |I|)p \Leftrightarrow \sum_{i \in I} p^2 \deg_i \geq (n - |I|)p \Leftrightarrow p \sum_{i \in I} \deg_i \geq n - |I| \Leftrightarrow p \geq \frac{n - |I|}{\sum_{i \in I} \deg_i}.$$

For the disk graph of radius  $r$  of a  $k$ -center instance with radius  $r$ , the set of centers is a dominating set  $|I| \leq k$ . If  $k = |I|$ , all the vertices are covered by  $I$ , and  $\sum_{i \in I} \deg_i \leq |I| \cdot n$ . Assuming  $k \leq n/2$ , then  $\frac{n-|I|}{\sum_{i \in I} \deg_i} \geq \frac{n-|I|}{|I|n} = \frac{1}{|I|} - \frac{1}{n} \geq \frac{1}{2k}$ . This gives the bound  $p \geq \frac{1}{2k}$  on the probability. Using this probability, the size of the random sample is  $\frac{n}{2k}$ . For  $k = O(1)$ , this sample has the same size as the input, so this sample does not improve the time complexity of the algorithm. For  $k = \Omega(1)$ , we compute the probability of finding a good sample. Using Markov's inequality,  $pr(|I_s| \geq a|I|p) \leq \frac{1}{a}$ . Substituting  $p = \frac{1}{2k}$  and  $|I| = k$ , the inequality becomes  $pr(|I_s| \geq a) \leq \frac{1}{2a}$ . For  $a = \frac{ck}{2}$ , the probability of finding a  $c$ -approximation with one sample is  $1 - \frac{1}{ck}$ .

So, it is possible to compute a  $c$ -approximation sample of size  $O(\frac{n}{k})$  for dominating sets of size  $k = \Omega(1)$ . The same bound applies to  $k$ -center. ◀

A pseudo-approximation for  $k$ -center [2] with  $k = O(\sqrt{n})$  can be achieved using Theorem 5.

---

## References

- 1 Mark De Berg, Marc Van Kreveld, Mark Overmars, and Otfried Schwarzkopf. Computational geometry. In *Computational geometry*, pages 1–17. Springer, 1997.
- 2 Shi Li and Ola Svensson. Approximating  $k$ -median via pseudo-approximation. *SIAM Journal on Computing*, 45(2):530–547, 2016.
- 3 László A Székely. Crossing numbers and hard Erdős problems in discrete geometry. *Combinatorics, Probability and Computing*, 6(3):353–358, 1997.
- 4 Endre Szemerédi and William T. Trotter. Extremal problems in discrete geometry. *Combinatorica*, 3(3-4):381–392, 1983.
- 5 Endre Szemerédi and William T Trotter Jr. A combinatorial distinction between the Euclidean and projective planes. *European Journal of Combinatorics*, 4(4):385–394, 1983.

# Recovering the homology of immersed manifolds

Raphaël Tinarrage

Datashape, Inria Paris-Saclay, France  
raphael.tinarrage@inria.fr

---

## Abstract

Given a sample of an abstract manifold immersed in some Euclidean space, we describe a way to recover the singular homology of the original manifold. It consists in estimating its tangent bundle—seen as subset of another Euclidean space—in a measure theoretic point of view, and in applying measure-based filtrations for persistent homology. The construction we propose is consistent and stable, and does not involve the knowledge of the dimension of the manifold. In order to obtain quantitative results, we introduce the normal reach, which is a notion of reach suitable for an immersed manifold.

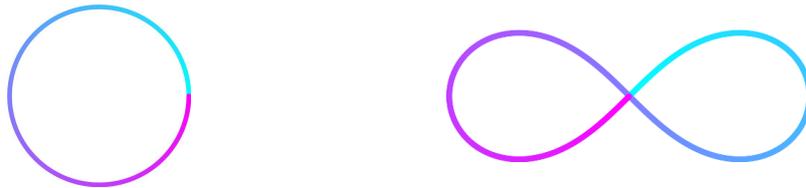
**2012 ACM Subject Classification** Mathematical Foundations

**Keywords and phrases** Topological Data Analysis, Persistent homology, Immersed manifold, Tangent bundle, Wasserstein distance

**Related Version** The complete version of the paper can be found at <https://arxiv.org/abs/1912.03033>

**Acknowledgements** The author want to thank Frédéric Chazal, Marc Glisse and Théo Lacombe for fruitful discussions and corrections.

Let  $\mathcal{M}_0$  be a compact  $\mathcal{C}^2$ -manifold of dimension  $d$ , and  $\mu_0$  a Radon probability measure on  $\mathcal{M}_0$  with support  $\text{supp}(\mu_0) = \mathcal{M}_0$ . Let  $E = \mathbb{R}^n$  be the Euclidean space and  $u: \mathcal{M}_0 \rightarrow E$  be an immersion. We assume that the immersion is such that self-intersection points correspond to different tangent spaces. In other words, for every  $x_0, y_0 \in \mathcal{M}_0$  such that  $x_0 \neq y_0$  and  $u(x_0) = u(y_0)$ , the tangent spaces  $d_{x_0}u(T_{x_0}\mathcal{M}_0)$  and  $d_{y_0}u(T_{y_0}\mathcal{M}_0)$  are different. Define the image of the immersion  $\mathcal{M} = u(\mathcal{M}_0)$  and the pushforward measure  $\mu = u_*\mu_0$ . We suppose that we are observing the measure  $\mu$ , or a close measure  $\nu$ . Our goal is to infer the singular homology of  $\mathcal{M}_0$  (with coefficients in  $\mathbb{Z}_2$  for instance) from  $\nu$ .

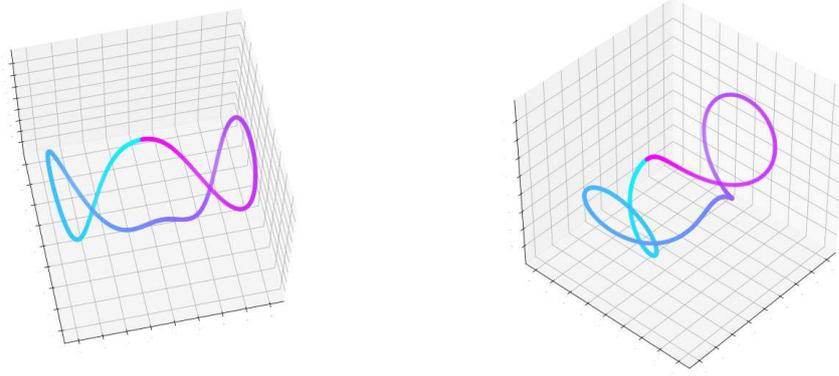


■ **Figure 1** Left: The abstract manifold  $\mathcal{M}_0$ , diffeomorphic to a circle. Right: The immersion  $\mathcal{M} \subset \mathbb{R}^2$ , known as the lemniscate of Bernoulli.

As shown in Figure 1, the immersion may self-intersect, hence the singular homology of  $\mathcal{M}_0$  and  $\mathcal{M}$  may differ. To get back to  $\mathcal{M}_0$ , we proceed as follows: let  $\mathcal{M}(E)$  be the vector space of  $n \times n$  matrices, and  $\check{u}: \mathcal{M}_0 \rightarrow E \times \mathcal{M}(E)$  the application

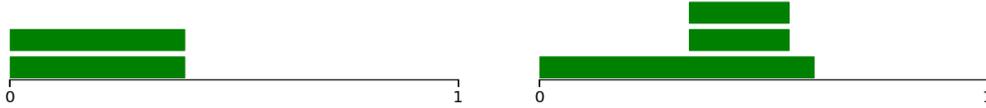
$$\check{u}: x_0 \mapsto \left( u(x_0), \frac{1}{d+2} p_{T_x \mathcal{M}} \right),$$

where  $p_{T_x \mathcal{M}}$  is the matrix representative of the orthogonal projection on the tangent space  $T_x \mathcal{M} \subset E$ . Define  $\check{\mathcal{M}} = \check{u}(\mathcal{M}_0)$ . The set  $\check{\mathcal{M}}$  is a submanifold of  $E \times \mathcal{M}(E)$ , diffeomorphic to  $\mathcal{M}_0$ . It is called the lift of  $\mathcal{M}_0$ .



■ **Figure 2** Two views of the submanifold  $\check{\mathcal{M}} \subset \mathbb{R}^2 \times \mathcal{M}(\mathbb{R}^2)$ , projected in a 3-dimensional subspace via PCA. Observe that it does not self-intersect

Suppose that one is able to estimate  $\check{\mathcal{M}}$  from  $\nu$ . Then one could consider the persistent homology of a filtration based on  $\check{\mathcal{M}}$ —say the Čech complex of  $\check{\mathcal{M}}$  in the ambient space  $E \times \mathcal{M}(E)$  for instance—and hope to read the singular homology of  $\mathcal{M}_0$  in the corresponding persistent barcode.



■ **Figure 3** Left: Persistence barcode of the 1-homology of the Čech filtration of  $\mathcal{M}$  in the ambient space  $\mathbb{R}^2$ . One reads the 1-homology of the lemniscate. Right: Persistence barcode of the 1-homology of the Čech filtration of  $\check{\mathcal{M}}$  in the ambient space  $\mathbb{R}^2 \times \mathcal{M}(\mathbb{R}^2)$ . One reads the 1-homology of a circle.

Instead of estimating the lifted submanifold  $\check{\mathcal{M}}$ , we propose to estimate the *exact lifted measure*  $\check{\mu}_0$ , defined as  $\check{\mu}_0 = \check{u}_* \mu_0$ . It is a measure on  $E \times \mathcal{M}(E)$ , with support  $\check{\mathcal{M}}$ . Using measure-based filtrations—such as the DTM-filtrations—one can also hope to recover the singular homology of  $\mathcal{M}_0$ .

It is worth noting that  $\check{\mathcal{M}}$  can be naturally seen as a submanifold of  $E \times \mathcal{G}_d(E)$ , where  $\mathcal{G}_d(E)$  denotes the Grassmannian of  $d$ -dimensional linear subspaces of  $E$ . From this point of view,  $\check{\mu}_0$  can be seen as a measure on  $E \times \mathcal{G}_d(E)$ , i.e., a varifold. However, for computational reasons, we choose to work in the matrix space  $\mathcal{M}(E)$  instead of  $\mathcal{G}_d(E)$ .

Here is an alternative definition of the exact lifted measure  $\check{\mu}_0$ : for any  $\phi: E \times \mathcal{M}(E) \rightarrow \mathbb{R}$  with compact support,

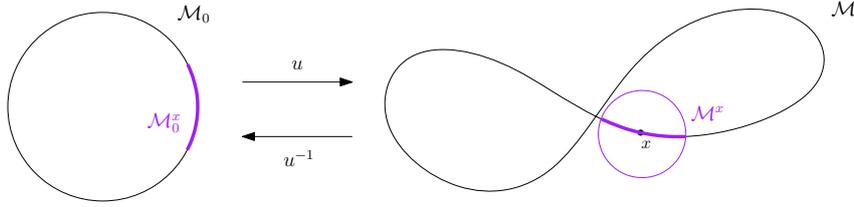
$$\int \phi(x, A) d\check{\mu}_0(x, A) = \int \phi\left(u(x_0), \frac{1}{d+2} p_{T_x \mathcal{M}}\right) d\mu_0(x_0).$$

Getting back to the actual observed measure  $\nu$ , we propose to estimate  $\check{\mu}_0$  with the *lifted measure*  $\check{\nu}$ , defined as follows: for any  $\phi: E \times \mathcal{M}(E) \rightarrow \mathbb{R}$  with compact support,

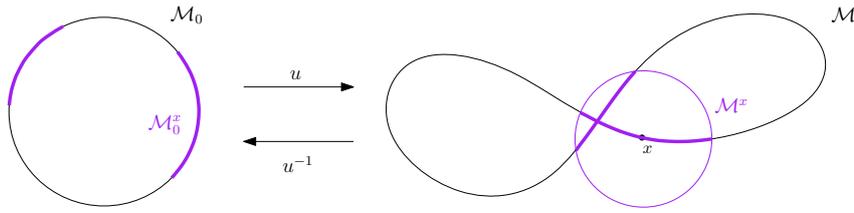
$$\int \phi(x, A) d\check{\nu}(x, A) = \int \phi\left(x, \bar{\Sigma}_\nu(x)\right) d\nu(x),$$

where  $\bar{\Sigma}_\nu(x)$  is the *normalized local covariance matrix* of  $\nu$  at  $x$ . It depends on a radius parameter  $r > 0$ , not made explicit in the notation. We prove that  $\bar{\Sigma}_\nu(x)$  can be used to estimate the tangent spaces  $\frac{1}{d+2} p_{T_x \mathcal{M}}$  of  $\mathcal{M}$ .

Such a tangent space is well estimated if  $x$  is far from an intersection. We quantify this property by introducing the normal reach of  $\mathcal{M}$  at  $x$ . The normal reach  $\lambda(x)$  is a nonnegative real number, and can be seen as a local version of the usual notion of reach. It allows to get back to  $\mathcal{M}_0$  from  $\mathcal{M}$  (see Figures 4 and 5).



■ **Figure 4** The set  $u^{-1}(\mathcal{M} \cap \bar{\mathcal{B}}(x, r))$ , with  $r < \lambda(x)$ , is connected.



■ **Figure 5** The set  $u^{-1}(\mathcal{M} \cap \bar{\mathcal{B}}(x, r))$ , with  $r \geq \lambda(x)$ , may not be connected.

Using the normal reach  $\lambda(x)$ , a bound  $\rho$  on the curvature of  $\mathcal{M}_0$ , and a well-chosen parameter  $r > 0$ , we can state an estimation result: the normalized local covariance matrix is close to the tangent space, with respect to the Frobenius norm.

► **Proposition 1.** *Let  $x_0 \in \mathcal{M}_0$ , and denote  $x = u(x_0)$ . Choose  $r < \lambda(x) \wedge \frac{1}{2\rho}$ . Then*

$$\left\| \bar{\Sigma}_\mu(x) - \frac{1}{d+2} T_x \mathcal{M} \right\|_{\text{F}} \leq cr,$$

where  $c > 0$  is a constant.

Moreover,  $\bar{\Sigma}_\nu(x)$  is stable with respect to  $\nu$  in Wasserstein distance: for every other measure  $\mu$ , and under some regularity assumptions on the measures, we have

$$\left\| \bar{\Sigma}_\mu(x) - \bar{\Sigma}_\nu(x) \right\|_{\text{F}} \leq c \left( \frac{W_1(\mu, \nu)}{r^{d+1}} \right)^{\frac{1}{2}},$$

where  $W_1(\mu, \nu)$  denotes the 1-Wasserstein distance, and  $c > 0$  is a constant.

If  $r$  is chosen greater than the normal reach  $\lambda(x)$ , this estimation may be biased closed to  $x$ , as shown in Figure 6. However, we prove a global estimation result, of the following form: the exact lifted measure  $\check{\mu}_0$  and the lifted measure  $\check{\nu}$  are close in the Wasserstein metric, as long as  $\mu$  and  $\nu$  are. As a consequence, the persistence diagrams of the DTM-filtrations based on  $\check{\mu}_0$  and  $\check{\nu}$  are close in bottleneck distance.

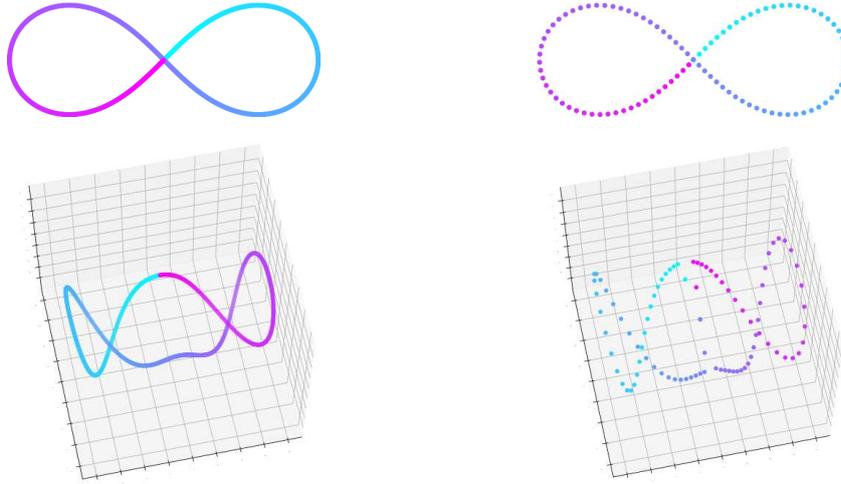
## Recovering the homology of immersed manifolds

► **Corollary 2.** *Select the parameters  $m \in (0, 1)$ ,  $r > 0$  and  $\gamma > 0$ . Assume that  $\mathcal{M}_0$  and  $\mu_0$  satisfy some regularity hypotheses. Let  $\nu$  be any probability measure on  $E$ . Suppose that  $r$  and the Wasserstein distance  $W_2(\mu, \nu)$  are small enough. Then the persistence diagrams of the DTM-filtration on  $\check{\mu}_0$  and on  $\check{\nu}$  are  $\epsilon$ -close in bottleneck distance, with*

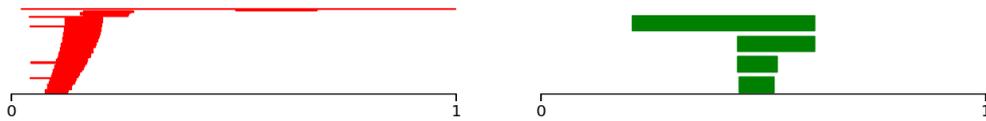
$$\epsilon = c(1 + \gamma c')^{\frac{1}{2}} m^{-\frac{1}{2}} r^{\frac{1}{4}} + 2c'' m^{\frac{1}{d}},$$

where  $c, c', c'' > 0$  are constants.

To conclude, the persistent homology of  $\check{\mu}_0$ —and its support  $\check{\mathcal{M}}$ , diffeomorphic to  $\mathcal{M}_0$ —can be recovered from  $\nu$  via its lift  $\check{\nu}$ .



■ **Figure 6** Left: The sets  $\text{supp}(\mu) = \mathcal{M}$  and  $\text{supp}(\check{\mu}_0) = \check{\mathcal{M}}$ , where  $\mu$  is the uniform measure on the lemniscate. Right: The sets  $\text{supp}(\nu)$  and  $\text{supp}(\check{\nu})$ , where  $\nu$  is the empirical measure on a 100-sample of the lemniscate. Parameters  $\gamma = 2$  and  $r = 0,1$ .



■ **Figure 7** Persistence barcodes of the 0-homology (left) and 1-homology (right) of the DTM-filtration of  $\check{\nu}$ . Observe that the 1-homology of the circle appears as a large feature of the barcode. Parameters  $\gamma = 2$ ,  $r = 0,1$  and  $m = 0,01$ .

# Understanding the topology and geometry of the persistence diagrams space using optimal transport

Vincent Divol

Datashape, Inria Saclay, France

<https://vincentdivol.github.io/>

vincent.divol@inria.fr

Théo Lacombe

Datashape, Inria Saclay, France

<https://tlacombe.github.io/>

theo.lacombe@inria.fr

---

## Abstract

---

Despite the obvious similarities between the metrics used in topological data analysis and those of optimal transport, an explicit optimal transport based formalism to study persistence diagrams and similar topological descriptors has yet to come. By considering the space of persistence diagrams as a measure space, and by observing that its metrics can be expressed as optimal partial transport problems, we introduce and study a generalization of persistence diagrams, namely Radon measures supported on the upper half plane. Such measures naturally appear in topological data analysis when considering continuous representations of persistence diagrams (e.g. persistence surfaces) but also as limits for laws of large numbers on persistence diagrams or as expectations of probability distributions on the persistence diagrams space. We stress that the standard definition of metrics between persistence diagrams, namely as a combinatorial optimization problem, is not suited to handle such general measures that could have a continuous support. Therefore, the first step of our work consists in extending these combinatorial metrics to the space of all non-negative Radon measures (even of infinite mass) in a consistent way. Building a true formal connection between the (metric) space of persistence diagram and the optimal transport theory has major benefits: from a theoretical perspective, it allows us to prove topological and geometric properties of this new space, which will also hold for the closed subspace of persistence diagrams, giving a better understanding of these commonly used topological descriptors. In terms of applications, we derive new statistical results and algorithms inspired from the optimal transport literature which allow to address difficult problems when considering persistence diagrams, such as barycenter estimation or quantization.

First, we provide a characterization of convergence in the space of persistence diagram (with respect to the diagram metrics) in terms of *vague* convergence of measures (convergence in duality with continuous, compactly supported functions). It formally reads

$$\text{OT}_p(\mu_n, \mu) \rightarrow 0 \Leftrightarrow \begin{cases} \mu_n \xrightarrow{v} \mu \\ \text{Pers}_p(\mu_n) \rightarrow \text{Pers}_p(\mu), \end{cases} \quad (1)$$

where  $(\mu_n)_n, \mu$  are any Radon measures supported on the upper half-plane (in particular, persistence diagrams),  $\xrightarrow{v}$  denotes the *vague* convergence of Radon measures, and  $\text{Pers}_p(\mu)$  is *total persistence* of  $\mu$ , that is the distance from a measure  $\mu$  to the empty diagram. Here,  $p \in [1, +\infty)$ ; a similar results hold in the case  $p = +\infty$  (encompassing the so-called bottleneck distance). Rephrasing convergence of persistence diagrams (for their standard metrics  $\text{OT}_p$ ) as a convergence of measures provides a powerful tool to study convergence and continuity in the space of persistence diagrams; in particular it allows us to give an exhaustive characterization of continuous *linear representations* of persistence diagrams, a common tool used when incorporating persistence diagrams in machine learning pipelines.

Second, this formalism allows us to prove new results regarding persistence diagrams in a random setting, as it enables to manipulate some counterpart of persistence diagrams with continuous support instead of discrete support, such as *expected persistence diagrams* and to prove convergence

## 2 Understanding the PD space using OT

rates and stability results in this context. In particular, we prove that if one consider two laws  $\xi, \xi'$  supported on  $\mathbb{R}^d$ , and denote by  $\text{Dgm}(\mathbb{X}_n)$  (resp.  $\text{Dgm}(\mathbb{X}'_n)$ ) the (random) persistence diagram obtained by building the Čech filtration on a  $n$ -sample  $\mathbb{X}_n = (x_1 \dots x_n)$  of law  $\xi$  (resp.  $\mathbb{X}'_n$  of law  $\xi'$ ), and by  $\mathbb{E}[\text{Dgm}(\mathbb{X}_n)]$  (resp.  $\mathbb{E}[\text{Dgm}(\mathbb{X}'_n)]$ ) the corresponding expected persistence diagrams), one has

$$\text{OT}_\infty(\mathbb{E}[\text{Dgm}(\mathbb{X}_n)], \mathbb{E}[\text{Dgm}(\mathbb{X}'_n)]) \leq W_\infty(\xi, \xi'), \quad (2)$$

where  $\text{OT}_\infty$  denotes the extension of the bottleneck distance to general Radon measures, and  $W_\infty$  denotes the so-called Wasserstein distance between probability measures. Finally, an optimal transport-based formalism has huge strengths in applications. Indeed, modern tools routinely used in computational optimal transport can be straightforwardly adapted to persistence diagrams metrics, allowing to estimate efficiently distances, barycenters, or quantizations of persistence diagrams. During the presentation, we will in particular showcase some modern algorithms (in quantization and barycenter estimation notably) that can be adapted from optimal transport literature in a faithful manner thanks to our formalism.

This presentation will be mostly based on the work done in [1].

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

**Keywords and phrases** Topological Data Analysis, Geometry, Statistics, Optimal Transport

---

### References

- 1 Vincent Divol and Théo Lacombe. Understanding the topology and the geometry of the persistence diagram space via optimal partial transport. <http://arxiv.org/abs/1901.03048>.

# Optimizing Embeddings using Persistence

**Jasna Urbančič**

Queen Mary University of London, United Kingdom  
j.urbanic@qmul.ac.uk

**Primož Škraba**

Queen Mary University of London, United Kingdom  
p.skraba@qmul.ac.uk

---

## Abstract

---

This work looks to optimize Takens-type embeddings of time series using persistent (co)homology. The motivation is to assume that an input time series exhibits periodic, quasi-periodic, or recurrent behaviour, then use continuous optimization to find good embeddings. Here we provide a practical approach to finding good embeddings with indications as to possible theoretical questions and directions which arise.

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

**Keywords and phrases** Embedding, Persistent Homology, Optimization

## 1 Introduction

This work combines techniques from computational topology and geometry to develop tools for finding recurrent behaviour in data, primarily time series data. While this has garnered recent attention, we focus on a different aspect of the approach. Our starting point is the so-called *time-delay embedding*. Originally proposed by Takens [18] for studying chaotic attractors, the technique constructs an embedding by mapping each point in the time series to a higher dimensional point by introducing a set of parameters appropriately called delays. The geometry and topology of the embedding carries a great deal of information about the global structure of the observed system. Indeed, Takens showed that a generic embedding will not only preserve the underlying space of a system, but also the dynamics. For example, point clouds in a shape of a circle represent periodic processes, high-dimensional tori are linked to quasi-periodicity and fractal geometry is evidence of chaotic behaviour [12, 18]. Combining this with persistent (co)homology [2] is a natural idea as it can be used to capture the shape of the embedding. Applying persistence diagrams to time-delay embeddings is a relatively well-established idea [3, 14, 10, 13].

A critical issue is the sensitivity of the embedding to the choice of parameters. For example, taking a sine function, a delay of a quarter of the period (in this case the optimal choice) embeds the sine function into a circle on the plane, whereas the choice of a sub-optimal delay produces a thin ellipse. There has been a substantial amount of work on for choosing these parameters, but most are based on the underlying dynamics [16, 5].

This work turns the question on its head: how can we find “good” embeddings, assuming the input (time series) exhibits recurrent behaviour? Our approach builds on a relatively recent approach which has primarily been used for combining persistence diagrams as a layer in deep networks [15, 1, 6, 7, 8]. One of the key results is that it is possible to define and derive a gradient almost everywhere, which makes continuous optimization possible. We aim to find parameter choices that will result in good embeddings in practice. Given a good embedding it is often possible to reconstruct the recurrent signal using circular coordinates [2, 19]. We believe this could be extended to more general settings as in [11].

## 2 Optimizing the Embedding

Here we outline our general approach: given a time series  $s(t)$ . For the sake of exposition, we assume the time series is uniform sampled at  $\Delta t$  intervals and is scaled so that  $|s(t)| \leq 1$ . In the classical approach, given a **delay vector**,  $\omega = [\omega_1, \omega_2, \dots, \omega_d]$ , the embedding is given by  $p_\omega(t) = [s(t), s(t + \omega_1), s(t + \omega_2), \dots, s(t + \omega_d)]$ . We modify this, convolving with a Gaussian function  $\varphi(\mu, \sigma)$  centered around each delay, where  $(\mu, \sigma)$  denote the parameters of the Gaussian function. The primary rationale for convolving with Gaussians rather than simply taking delta functions is that since in practice we have discrete samples than the underlying continuous signal, this gives a continuous curve without explicitly computing a piecewise-linear interpolation. Secondly, convolving with Gaussians gives the curve nicer geometric properties, i.e. the resulting curve is  $C^\infty$ . Appropriately, we redefine the delay vector  $\omega = [\mu_1, \mu_2, \dots, \mu_d, \sigma_1, \sigma_2, \dots, \sigma_d]$ . The  $i$ -th coordinate of the embedding is then given by  $p_{\omega(i)}(t) = \varphi(\mu_i, \sigma_i) * s(t)$ . We observe that each  $\omega$  defines an embedding of the time samples into  $\mathbb{R}^d$  as a point cloud which we denote by  $\mathcal{P}_\omega$ . Our goal is to optimize this embedding via persistence diagrams.

Given the embedding, we first compute the persistence diagram  $\text{Dgm}(\mathcal{P}_\omega)$ . We refer the reader to [4] for background on persistence. Persistent homology requires a filtration function and a natural choice in this setting is the sub-level sets of the distance function, for  $x \in \mathbb{R}^d$   $d(x, \mathcal{P}_\omega)$ . For low dimensional settings (2d or 3d), we use  $\alpha$ -shape filtration, but stress that the approach can be applied with any distance based filtration, i.e. Čech or Vietoris-Rips filtrations<sup>1</sup>. To optimize the embedding, we require the computation of the gradient. This is possible on persistence diagrams as has been shown in a series of work [15, 1, 9, 7, 6, 8]. We remind the reader that persistence diagrams are a multiset of points  $(b_i, d_i)$ , where the coordinates are referred to as *birth* and *death times*. Assuming sufficient genericity, given a functional on the persistence diagram, one can define a chain rule to compute a gradient. Here we focus on the functional for a single recurrent pattern – we treat dimension zero separately, as all points in dimension zero are born simultaneously. The functional for dimension zero aims to push all but one persistence points towards the diagonal, i.e.  $(0,0)$ . This also ensures that the embedding does not simply increase the scale. For higher dimensions, all but a fixed number of the most persistent points are moved to towards diagonal, e.g.  $d_i - b_i \rightarrow 0$ . The deaths of the most persistent points are moved towards one. Assume that the points in the  $k$ -dimensional diagram (denoted  $\text{Dgm}_k$ ) are ordered by persistence and we would like to keep the  $m$  most persistent points. The corresponding functional is

$$\mathcal{F} = \sum_{\substack{i \in \text{Dgm}_0 \\ d_i \neq \infty}} d_i^2 + \sum_{\substack{i \in \text{Dgm}_k \\ i=1}}^{m_k} (1 - d_i)^2 + \sum_{\substack{i \in \text{Dgm}_k \\ i=m_k+1}}^n (d_i - b_i)^2. \quad (1)$$

The first term of the functional ensures the space remains connected as quickly as possible (minimizing the persistence in dimension 0), while the second and third terms handle higher dimensions. The second term aims to move the deaths of a fixed number,  $m_k$ , most persistent points towards 1, whereas the third term moves the remainder of the points in dimension  $k$  towards the diagonal. Hence we can apply gradient descent using the following equation:  $\omega_j = \omega_j - \Delta t \frac{\partial \mathcal{F}}{\partial \omega_j}$ . For recurrent behaviour we will often choose  $m_1 = 1$  and minimize all other dimensions.

<sup>1</sup> Other distance-based filtrations such as Witness complexes or Graph-induced complexes are possible with additional modifications to the described derivations

### 3 Example

For illustration, we show the results for periodic signals. In Figure 1, we show  $s_s(t) = \sin(t) + \varepsilon\mathcal{U}(-1, 1)$  and triangular signal  $s_\Delta(t) = \int_0^t \text{sgn}(\sin(\frac{u}{2}))du + \varepsilon\mathcal{U}(-1, 1)$ , where the term  $\mathcal{U}(-1, 1)$  represents uniformly distributed noise with level  $\varepsilon$ . We show the results with  $\varepsilon = 0.2$  in Figure 1. We let gradient descent run for at most 50 iterations.

Surprisingly, for all tested noise levels, the optimization converged quickly and to roughly same values for delay after 50 iterations. To estimate how close to the correct values these approximations are we have to remember that in case of sine signal, the embedding with delay of  $\frac{\pi}{2}$  is optimal. A similar reasoning applies to the triangle signal case. Gradient descent for sine converges to 1.567 when the level of noise is low, and to 1.571 when the level of noise is high, which is close to  $\frac{\pi}{2}$  (allowing for noise).

Finally, we show the reconstructed signal using the circular coordinates approach described in [2, 19]. To construct the circular coordinates, we use persistent cohomology and the associated 1-dimensional cocycle representative. We do not correct for phase shift in these experiments as this is work in progress but will be addressed in later work.

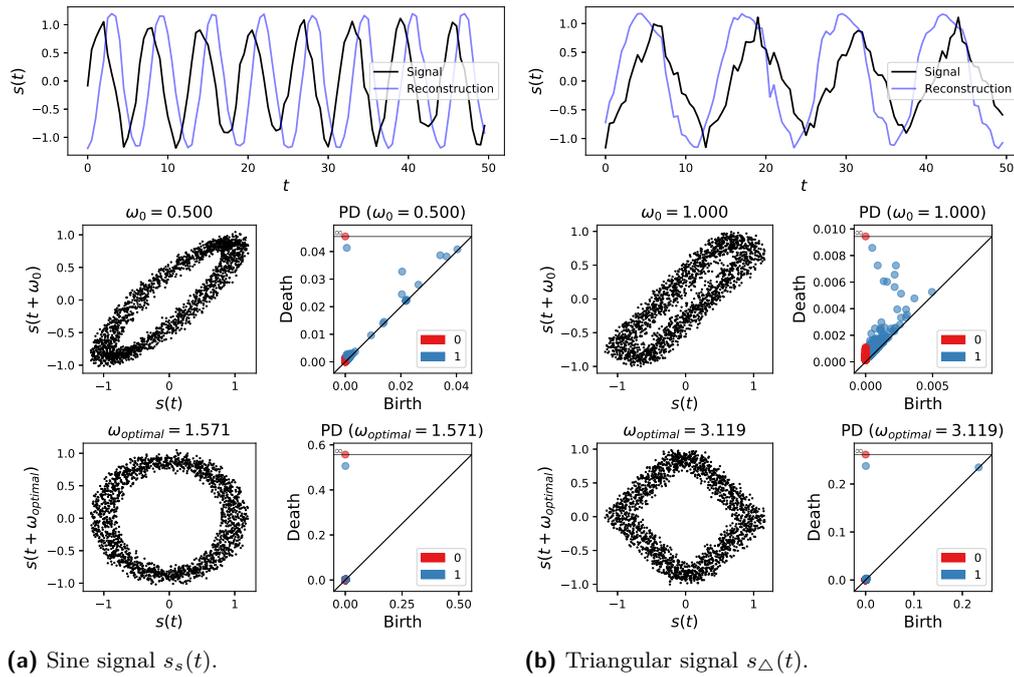
### 4 Discussion

Due to space constraints, we do not include additional experiments – including on quasi-periodic signals. While the primary motivation is to construct embeddings which are good enough, this raises interesting new questions in understanding the space of embeddings from a computational geometry and topology perspective. It is known that in sufficiently high dimension, there is an isotopy between embedding [17]. In the future, we plan to investigate whether guarantees can be given under additional assumptions on the input signal, both recurrence and periodicity.

---

#### References

- 1 Chao Chen, Xiuyan Ni, Qinxun Bai, and Yusu Wang. A topological regularizer for classifiers via persistent homology. *arXiv preprint arXiv:1806.10714*, 2018.
- 2 Vin De Silva, Dmitriy Morozov, and Mikael Vejdemo-Johansson. Persistent cohomology and circular coordinates. *Discrete & Computational Geometry*, 45(4):737–759, 2011.
- 3 Vin de Silva, Primoz Skraba, and Mikael Vejdemo-Johansson. Topological analysis of recurrent systems. In *Workshop on Algebraic Topology and Machine Learning, NIPS*, 2012.
- 4 Herbert Edelsbrunner and J Harer. *Computational topology: an introduction*. 2010. OCLC: 1073022164.
- 5 Armin Eftekhari, Han Lun Yap, Michael B Wakin, and Christopher J Rozell. Stabilizing embedology: Geometry-preserving delay-coordinate maps. *Physical Review E*, 97(2):022222, 2018.
- 6 Rickard Brüel Gabriëlsson, Vignesh Ganapathi-Subramanian, Primoz Skraba, and Leonidas J Guibas. Topology-aware surface reconstruction for point clouds. *arXiv preprint arXiv:1811.12543*, 2018.
- 7 Rickard Brüel Gabriëlsson, Bradley J Nelson, Anjan Dwaraknath, Primoz Skraba, Leonidas J Guibas, and Gunnar E Carlsson. A topology layer for machine learning. *ArXiv, abs/1905.12200*, 2019.
- 8 Jacob Leygonie, Steve Oudot, and Ulrike Tillmann. A framework for differential calculus on persistence barcodes. *arXiv preprint arXiv:1910.00960*, 2019.
- 9 Michael Moor, Max Horn, Bastian Rieck, and Karsten Borgwardt. Topological autoencoders. *arXiv preprint arXiv:1906.00722*, 2019.



■ **Figure 1** Top: Two signals  $s_s(t)$  and  $s_{\Delta}(t)$  with  $\varepsilon = 0.2$ . Middle: Embedding of  $s(t)$ 's into two dimensions with initial delay  $\omega_0$  (left) and persistence diagram (PD) for the embedding on the left (right). Bottom: Embedding of  $s(t)$ 's after 50 iterations of gradient descent with delay  $\omega_{optimal}$  (left) and persistence diagram for the embedding on the left (right).

- 10 Jose A Perea. Persistent homology of toroidal sliding window embeddings. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6435–6439. IEEE, 2016.
- 11 Jose A Perea. Multiscale projective coordinates via persistent cohomology of sparse filtrations. *Discrete & Computational Geometry*, 59(1):175–225, 2018.
- 12 Jose A. Perea. Topological Time Series Analysis. *arXiv:1812.05143 [cs, math]*, November 2018. arXiv: 1812.05143. URL: <http://arxiv.org/abs/1812.05143>.
- 13 Jose A Perea. Topological time series analysis. *Notices of the American Mathematical Society*, 66(5), 2019.
- 14 Jose A Perea and John Harer. Sliding windows and persistence: An application of topological methods to signal analysis. *Foundations of Computational Mathematics*, 15(3):799–838, 2015.
- 15 Adrien Poulenard, Primoz Skraba, and Maks Ovsjanikov. Topological Function Optimization for Continuous Shape Matching. *Computer Graphics Forum*, 37(5):13–25, August 2018. URL: <http://doi.wiley.com/10.1111/cgf.13487>, doi:10.1111/cgf.13487.
- 16 Tim Sauer, James A Yorke, and Martin Casdagli. Embedology. *Journal of statistical Physics*, 65(3-4):579–616, 1991.
- 17 Arkadiy B Skopenkov. Embedding and knotting of manifolds in euclidean spaces. *London Mathematical Society Lecture Note Series*, 347:248, 2008.
- 18 Floris Takens. Detecting strange attractors in turbulence. In David Rand and Lai-Sang Young, editors, *Dynamical Systems and Turbulence, Warwick 1980*, volume 898, pages 366–381. Springer Berlin Heidelberg, Berlin, Heidelberg, 1981. Series Title: Lecture Notes in Mathematics. URL: <http://link.springer.com/10.1007/BFb0091924>, doi:10.1007/BFb0091924.
- 19 Mikael Vejdemo-Johansson, Florian T Pokorny, Primoz Skraba, and Danica Kragic. Cohomological learning of periodic motion. *Applicable algebra in engineering, communication and computing*, 26(1-2):5–26, 2015.

# Finding Surfaces in 2-Dimensional Simplicial Complexes with Bounded Treewidth 1-Skeletons

Mitchell Black<sup>1</sup>

Department of Computer Science  
Oregon State University, Corvallis, Oregon, USA  
blackmit@oregonstate.edu

Amir Nayyeri<sup>1</sup>

Department of Computer Science  
Oregon State University, Corvallis, Oregon, USA  
nayyeria@oregonstate.edu

---

## Abstract

---

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

**Keywords and phrases** Treewidth, Computational Topology, Combinatorial Surfaces

## 1 Introduction

The problem of finding a surface embedded in a topological space is a common problem in topology. A 2-manifold is non-orientable if and only if it contains a subspace homeomorphic to the Möbius band. The Unknot Recognition problem can be solved by deciding whether a simple cycle in a particular 3-manifold bounds a disk [1]. We are interested in a combinatorial variant of this problem of finding a subcomplex of a simplicial complex homeomorphic to a given surface. We consider the following problem.

### 2-DIM-SURFACE

*Input:* A 2-dimensional simplicial complex  $K$ , a subcomplex  $\beta \subset K$  of disjoint union of simple cycles, and a compact surface  $\Sigma$  specified by its genus  $g$  and orientability

*Question:* Is there a subcomplex  $S \subset K$  homeomorphic to  $\Sigma$  with boundary  $\beta$ ?

This problem is a generalization of the problem 2-DIM-SPHERE [2] of finding a subcomplex homeomorphic to the 2-sphere; in this case, the boundary  $\beta = \emptyset$  and the surface  $\Sigma$  has genus 0 and is orientable. 2-DIM-SPHERE is NP-Hard [5]. 2-DIM-SPHERE is also W[1]-Hard when parameterized by the number of 2-simplices of the sphere [2]. Both of these hardness results apply to our problem as well.

We present an algorithm for 2-DIM-SURFACE that is parameterized by the treewidth  $k$  of the 1-skeleton of  $K$  and the genus  $g$  of  $\Sigma$ .

► **Theorem 1.** *Given  $K$ ,  $\beta$ , and  $\Sigma$  with genus  $g$ , there is an  $(k + g)^{O(k^2)}n$  time algorithm to solve 2-DIM-SURFACE, where  $k$  is the treewidth of the 1-skeleton of  $K$ .*

Our algorithm implies that finding a surface with constant genus within a simplicial complex is fixed parameter tractable with respect to the treewidth of the complex. Moreover, as  $g = \Omega(n^2)$ , it implies a polynomial time algorithm for finding all surfaces within a simplicial complex of constant treewidth.

---

<sup>1</sup> This material is based upon work supported by the National Science Foundation under Grant Nos. CCF-1617951 and CCF-1816442.

## 2 Overview

### Tree Decompositions

Our algorithm uses a nice tree decomposition of the 1-skeleton of a 2-dimensional simplicial complex  $K$  to find surfaces with given boundary in  $K$ . A tree decomposition of a graph is a rooted tree where each node of the tree is associated with a bag of vertices of the graph, satisfying certain axioms. A nice tree decomposition [3] is a type of tree decomposition that minimizes the difference between adjacent bags which makes it easier to design dynamic programs. A nice tree decomposition has four types of nodes: introduce, forget, join, and leaves. The bags of introduce and forget nodes differ from the bags of their children by a single vertex. One key property of a nice tree decomposition is that on any path from the root to a leaf the bags containing a given simplex form a subpath of this path; the endpoints of this path are said to introduce and forget this simplex.

### Candidate Solutions

Given a tree decomposition of the 1-skeleton of  $K$  and a node  $t$  in the tree decomposition, let  $K_t$  be the subcomplex of  $K$  induced by the union of all bags associated with nodes in the subtree rooted at  $t$ . Given a solution surface  $S \subset K$ , we consider the partial solution  $S_t := S \cap K_t$ . We store all subcomplexes of  $K_t$  that could possibly be extended to a surface for each node  $t$ ; we call such a subcomplex a candidate solution. The following lemma is a key observation for identifying candidate solutions.

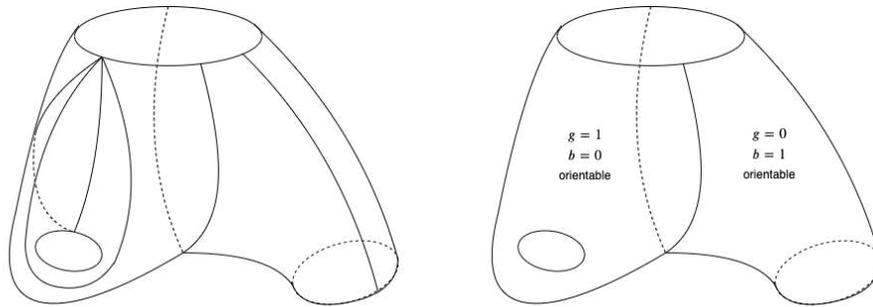
► **Lemma 2.** *Let  $S$  be a surface,  $t$  a node in a tree decomposition with bag  $X_t$ , and  $v \in K_t \setminus X_t$  a vertex. The link of  $v$  in  $S_t$  equals the link of  $v$  in  $S$ .*

Any vertex  $v$  already introduced and forgotten at the current bag must have the same link in a candidate solution as it would in an actual solution. Each time we forget a vertex  $v$ , we can therefore discard any candidate solution where the link of  $v$  in the candidate solution could not be the link of  $v$  in an actual solution. This allows us to (1) prune candidate solutions and (2) express candidate solutions solely in terms of the vertices in the bag associated with the current node. When we forget a vertex  $v$ , we want to remove any reference to this vertex in a candidate solution while still retaining the topological information of the candidate solution. We do this by merging all faces incident to  $v$  into a single face. The resulting topological space is no longer a simplicial complex as a face can have more than three vertices on its boundary. We thus introduce a new data structure, the *annotated cell complex*, to represent candidate solutions.

### Annotated Cell Complexes

A cell complex [4] is a set of edges, a set of faces, and a boundary map between faces and cyclic sequences of edges. A cell complex gives a description of a surface as a set of disks identified along shared edges in their boundaries. An individual disk can have its own surface information - such as genus, number of boundary components, and orientability - if multiple edges on the boundary of the disk are identified. We make the observation that the surface information of a individual face is independent of the actual edges on the boundary of the disk. We therefore generalize a cell complex to an annotated cell complex to allow ourselves to represent a candidate solution using fewer edges. Each face in an annotated cell complex is annotated with a genus, number of boundary components, and a boolean to denote whether or not it is orientable. By forgetting the exact edges in a cell complex and only remembering

topological features, these annotations allow us to express a cell complex with  $O(n)$  edges using  $O(k)$  space. The faces in annotated cell complex can be thought of as a collection of arbitrary surfaces - not just disks as in a cell complex - identified along shared edges in their boundaries. A candidate solution at a node in the tree decomposition is represented by an annotated cell complex with edges taken between vertices of the bag. Therefore, a candidate solution is composed of  $O(k^2)$  edges and faces, and for each face there are  $O(g)$  possibilities for its genus and orientability. Overall, the table entry for each node can have  $(k + g)^{O(k^2)}$  candidate solutions, and the total running time is  $(k + g)^{O(k^2)}n$ .



■ **Figure 1** Some handles, crosscaps, and boundaries in a cell complex are the result of edges on the boundary of the same face being identified. We can reduce the number of edges in our cell complex by simply storing the number of each of these features on a face.

---

## References

- 1 Ian Agol, Joel Hass, and William P. Thurston. The computational complexity of knot genus and spanning area, 2002.
- 2 Benjamin Burton, Sergio Cabello, Stefan Kratsch, and William Pettersson. The Parameterized Complexity of Finding a 2-Sphere in a Simplicial Complex. In Heribert Vollmer and Brigitte Vallée, editors, *34th Symposium on Theoretical Aspects of Computer Science (STACS 2017)*, volume 66 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 18:1–18:14, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.
- 3 Marek Cygan, Fedor V. Fomin, Łukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer International Publishing, 2015.
- 4 Jean Gallier. The classification theorem for compact surfaces and a detour on fractals, 2008.
- 5 Sergei Ivanov (<https://mathoverflow.net/users/4354/sergei-ivanov>). computational complexity. MathOverflow. URL:<https://mathoverflow.net/q/118428> (version: 2013-01-09).

# Topology Aware Morphological Operations

**Erin W. Chambers**

St. Louis University  
erin.chambers@slu.edu

**Ellen Gasparovic**

Union College  
gasparoe@union.edu

**Tao Ju**

Washington University in St. Louis  
taoju@wustl.edu

**David Letscher**

St. Louis University  
david.letscher@slu.edu

**Hannah Schreiber** 

St. Louis University  
hannah.schreiber@slu.edu

**Dan Zeng**

Washington University in St. Louis  
danzeng@wustl.edu

---

## Abstract

---

This work in progress aims to give a solution of the homological simplification problem in  $\mathbb{R}^2$  that is also geometrically nice.

**2012 ACM Subject Classification** Theory of computation  $\rightarrow$  Computational geometry; Computing methodologies  $\rightarrow$  Shape analysis; Mathematics of computing  $\rightarrow$  Topology

**Keywords and phrases** Medial axis, Homological Simplification, Morphological Operations

**Funding** The first and third through sixth authors' research is partially supported by NSF grant NBI-1759807. The first author is also supported by NSF grants CCF-1907612 and CCF-1614562.

When analyzing graphical data, we are often constrained to work with noisy data because it is generated from real world elements, for example using MRI or CT scans. A popular way to counteract the noise and simplify the shape is to use *morphological operations* such as *opening* and *closing*; see for example [5] and [4]. These methods have the advantage of yielding a “nice looking” result because they take into account the geometry of the shape. On the other hand, they completely ignore the topology of the shape and can actually introduce new topological features instead of simply reducing the existing number.

In this work we propose a version of the opening operation in  $\mathbb{R}^2$  that takes into account topology. More precisely, we give a solution to the *homological simplification problem* in  $\mathbb{R}^2$ , which has the additional advantage of preserving a geometrically nice appearance. The homological simplification problem asks: given two shapes  $S \subseteq N$ , can one construct an intermediate shape  $R$  such that  $S \subseteq R \subseteq N$  and the homology  $H(R)$  is isomorphic to the homology induced by the inclusion map from  $H(S)$  to  $H(N)$ ? In two dimensions, there is always a solution to the homological simplification problem, but not in higher dimensions. In fact, it is NP-hard to calculate one (if it exists) even in three dimensions [1].

For this purpose, we will use the notion of the *medial axis* [2] to find geometrically satisfactory ways to fix the topology of the shape. The medial axis of a shape  $S$  is the set of all points having more than one closest point on the boundary of  $S$ . This tool gives us a

good guideline to follow to reconnect components that were disconnected by the opening operation in a “geometrically nice” way.

**Problem statement.** Given a piece-wise smooth bounded shape  $X$  in  $\mathbb{R}^2$  and a fixed radius  $\epsilon$  yielding the thinned shape  $X_{-\epsilon} = \{x \in X \mid B(x, \epsilon) \subset X\}$ , compute a shape  $Y$  such that  $X_{-\epsilon} \subseteq Y \subseteq X$  and such that the homology  $H(Y)$  is isomorphic to  $H(X_{-\epsilon} \hookrightarrow X)$ , i.e.,  $Y$  has the homology of  $X_{-\epsilon}$  except for the homology features that die entering  $X$ .

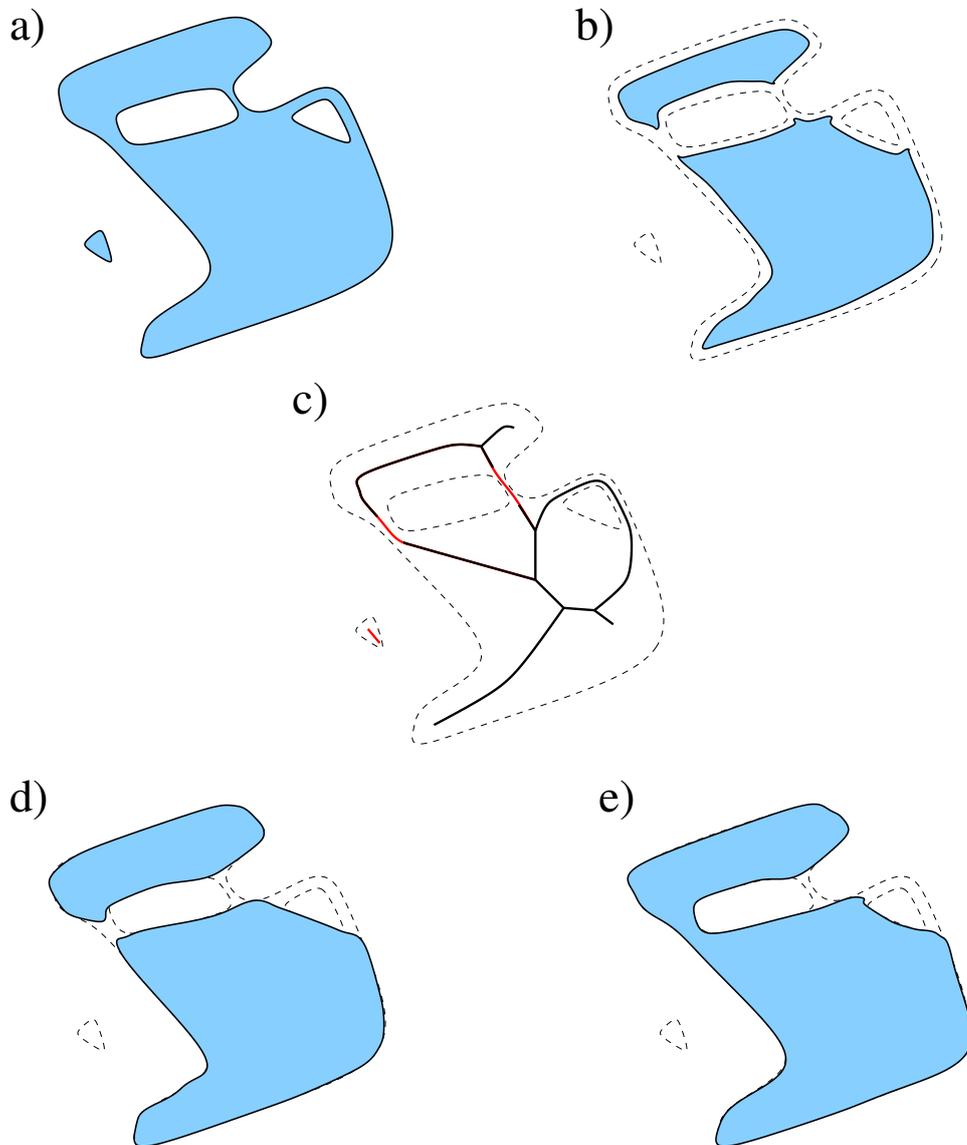
**Approach.** Let  $M(X)$  and  $M(X_{-\epsilon})$  be the medial axis of  $X$  and the thinned shape, respectively. Because  $X$  is piece-wise smooth and bounded in  $\mathbb{R}^2$ ,  $M(X)$  is a finite graph [3]. Recall that morphological opening is defined as  $Open(X) = \cup_{x \in M(X_{-\epsilon})} B(x, r(x))$ , where  $r(x) = d(x, \partial X)$  is local feature size. This is built by eroding the shape by  $\epsilon$  (see Figure 1b), but including balls of radius  $r(x)$  for each point in the erosion (see Figure 1d). Unfortunately,  $Open(X)$  may have the wrong topology; see Figure 1a and 1d. Let  $C = M(X) - M(X_{-\epsilon})$ ; see Figure 1c. Our approach is to add a subset of  $C$  to  $M(X_{-\epsilon})$  to obtain a new graph  $M^*$ , and then build a solution to the homological simplification problem  $Y$  from this medial axis. More precisely, we will define  $Y = \cup_{x \in M^*} B(x, r(x))$ , which is a superset of  $Open(X)$  and a subset of  $X$  as desired. However, we must choose the correct subset of  $C$  to add, since  $M(X_{-\epsilon}) \cup C = M(X)$  may contain loops or additional components which result in not having the correct topology, i.e. its homology group is not isomorphic to  $H(X_{-\epsilon} \hookrightarrow X)$ .

We consider starting with  $M(X_{-\epsilon})$ , and order edges in  $C$  by computing the levelset filtration from  $X_\delta$  for  $-\epsilon \leq \delta \leq 0$ . Then, we take the medial axis of each intermediate levelset, adding an edge from  $C$  as soon as it entirely appears in the medial axis of the levelset  $X_\delta$ . This gives a filtration of the graph  $C$ ; we initialize a queue  $Q$  with the edges of  $C$  in this order. We build  $Y$ 's medial axis  $M$  in a loop, starting with  $M = M(X_{-\epsilon})$ , and considering an edge from  $Q$ . As we add an edge, there are four possible cases:

- The edge could not cause a topological change; in this case we add the edge to  $M$  and continue.
- An edge of  $C$  could add a new component to the graph; in this case, we put the edge back at the end of the queue.
- An edge of  $C$  could form a new loop; in this case we discard the edge without adding to  $M$  or re-inserting in  $Q$ .
- An edge of  $C$  could merge two components in  $M$ ; in this case we add the edge to  $M$ .

We continue until the queue is empty, or until each remaining edge in  $Q$  is one that would create a new component. This ordering gives us the “thickest” possible shape, i.e., the minimum of  $r$  is maximized. We are essentially greedily adding the thickest edges of  $C$  until we have the correct homology. At the end, we then prune edges from  $C$  in  $M$  if their removal does not change the number of connected components; this results in our proposed medial axis,  $M^*$ , and we form  $Y = \cup_{x \in M^*} B(x, r(x))$ . See Figure 1e for an example.

The aim of this work is to show that  $Y$  as described above solves the simplification problem in  $\mathbb{R}^2$ . To do so, we first will prove that the opening does not introduce any new 1-dimensional features in  $X_{-\epsilon}$ , even though new 0-dimensional features can appear. It then suffices to show that the induced map from  $H(X_{-\epsilon})$  to  $H(Y)$  is surjective (i.e., no new feature appears in  $Y$ ) and that the induced map from  $H(Y)$  to  $H(X)$  is injective (i.e., no feature of  $Y$  dies in  $X$ ). Furthermore, we characterize some of the geometric properties of the simplified shapes.



■ **Figure 1** a) The original shape  $X$ , b) the eroded shape  $X_{-\epsilon}$ , c) the medial axis of the shape (with  $M(X_{-\epsilon})$  shown darker and  $C$  lighter), d) the result of opening,  $Open(X)$ , which does not have the correct homology, e) our topologically correct opening  $Y$ .

► Remark 1. The problem can also be seen from a dual point of view: in morphology, there is an operation dual to opening, called *closing*, which takes  $X_{+\epsilon} = \{B(x, \epsilon) \mid x \in X\}$ . The homological simplification problem here considers the inclusion of  $X$  in the thickened  $X_{+\epsilon}$ , and we can use a similar process (although using the complement of the shapes and their inclusions) to get a geometrically nice solution.

**Conclusion.** In  $\mathbb{R}^2$ , we have given a method to simplify a given shape without the disadvantage of creating additional topology and at the same time preserving the advantage of morphological operations resulting in correct-looking shapes. While this method works in  $\mathbb{R}^2$ , there exist counterexamples in  $\mathbb{R}^3$  when trying to generalize the construction. We hope to generalize this construction in higher dimensions when there exists a solution to the homological simplification problem.

---

**References**

---

- 1 Dominique Attali, Ulrich Bauer, Olivier Devillers, Marc Glisse, and André Lieutier. Homological Reconstruction and Simplification in  $\mathbb{R}^3$ . *Computational Geometry*, 48(8):606 – 621, 2015.
- 2 Harold F. Blum. A Transformation for Extracting new Descriptors of Form. *Models for the perception of speech and visual form*, pages 362–380, 1964.
- 3 James Damon and Ellen Gasparovic. *Medial/skeletal linking structures for multi-region configurations*, volume 250. American Mathematical Society, 2017.
- 4 Laurent Najman and Hugues Talbot. *Mathematical Morphology: from Theory to Applications*. ISTE-Wiley, June 2010. ISBN: 9781848212152 (520 pp.).
- 5 Jean Serra. *Image Analysis and Mathematical Morphology*. Academic Press, Inc., USA, 1983.

# Which Integrable Projection is Which?

**Josh Vekhter**

UT Austin, USA <http://www.cs.utexas.edu/~josh/>  
josh@cs.utexas.edu

**Etienne Vouga**

UT Austin, USA <https://www.cs.utexas.edu/users/evouga/>  
evouga@cs.utexas.edu

---

## Abstract

---

Given a piecewise constant vector field  $\mathbf{v}$  on a manifold  $\mathcal{M}$ , we begin with the question of when it can be expressed as the gradient of some scalar field  $s$ , in which case we say that this field is *discretely integrable*. When  $\mathbf{v}$  is not integrable, one can then begin considering projections from  $\mathbf{v}$  to the space of integrable fields. In this document we note that there are a number of related notions of integrable projection defined in the literature. We present a framework in which we generalize three of these notions to simplicial complexes in arbitrary dimension, and which shows promise as a computational tool for solving otherwise combinatorially hard problems in computational topology.

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

**Keywords and phrases** Discrete Differential Geometry, Mathematical Foundations of Topological Combinatorics, Basketweaving

**Acknowledgements** We want to thank Yousuf Soliman for suggesting substitution to get between Equations 3 and 4, and Paul Zhang, David Palmer, and Justin Solomon for helpful discussions regarding applications of integrability to problems in mesh generation.

## 1 Background

A fundamental operation in many computer graphics and engineering applications, ranging from surface parameterization to design of stripe patterns, weaves, and trusses [8, 21, 1], is the projection of a vector field onto the subspace of curl-free fields: given a compact manifold  $\mathcal{M}$  and a vector field  $\mathbf{v}$  on  $\mathcal{M}$ , many applications demand one to recover the vector field  $\mathbf{w}$  on  $\mathcal{M}$  such that  $\nabla \times \mathbf{w} = 0$  and  $\|\mathbf{v} - \mathbf{w}\|$  is as small as possible [3].

The classic solution to this problem is given by the Helmholtz-Hodge decomposition [7]. Unfortunately, many applications require additional constraints on  $\mathbf{w}$ , in addition to being curl-free. For instance, one might wish to recover integrable fields which satisfy certain boundary conditions [19], a setting where even in 2D and 3D the correct discrete decomposition into orthogonal spaces [4] is still an area of active research [13, 24, 17]. Similarly, for computational fabrication applications it may be desirable to find integrable  $\mathbf{w}$  that differs only in magnitude, but not direction, from  $\mathbf{v}$  (so that  $\mathbf{w}(p) = s(p)\mathbf{v}(p)$  for some scalar field  $s : \mathcal{M} \rightarrow \mathbb{R}$ ).

Orthogonally, in settings like surface parameterization, one may wish to recover integrable fields which are everywhere approximately unit normed, in order to recover parameterizations where all cells have roughly equal side lengths [22, 18]. One open challenge in higher dimensional parameterization problems is the automatic placement and recovery of singular structures, a problem studied in this 2013 SoCG contribution [6]. Here one question is whether the computational approach to surface parameterization which has had much success in 2D [5, 20] is up to the task in 3D, where singular topologies can be much more complex [10].

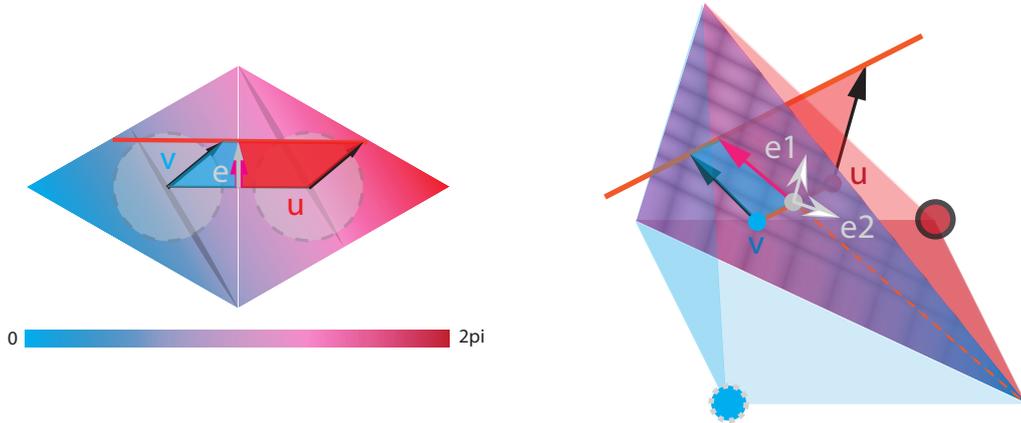
## 2 Defining Discretely Integrable Fields

Here we introduce a definition of *discrete integrability* suitable for modeling piecewise constant fields on simplicial approximations of  $k$ -manifolds embedded in  $\mathbb{R}^n$ . This generalizes Definition 3 in [15]. There, the authors introduced a measure of discrete rotation (illustrated in our Fig 1, (left)). In Theorem 1 they prove a necessary and sufficient condition for global vector field integrability in  $\mathbb{R}^2$  over simply connected domains, and an analogous theorem holds in our higher dimensional setting.

Let  $DM$  be a simplicial  $k$ -complex approximating a manifold  $\mathcal{M}$ . Concretely, consider the list of  $k$ -simplices  $S^k$ . We say that this complex has the structure of *mesh*, approximating a manifold without boundary, if and only for each simplex  $s_i \in S^k$  there exists a bounding set  $S_i^k \subset S^k$  with  $k+1$  elements, so that for each  $s_j \in S_i^k$ , the intersection  $s_i \cap s_j$  has dimension  $k-1$ . All such pairs of simplices  $s_i, s_j$  with non-trivial intersection can be said to *share a facet*. In our setting, we associate a vector  $v_i \in \mathbb{R}^n$  to each  $S_i \in S^k$ .

► **Definition 1.** *Two vectors  $v_i, v_j$  associated to on simplices  $s_i, s_j$  that share a common facet are discretely integrable if they share a common projection. i.e. let  $e_1, \dots, e_{m-1}$  span  $s_i \cap s_j$ . Then discrete integrability amounts to the condition that  $v_i \cdot e_k = v_j \cdot e_k$  for all  $k$ .*

Definition 1 is illustrated in 2D and 3D in figure 1.



► **Figure 1** In these figures, the blue/red color gradient encodes a scalar field compatible with discrete gradient field  $[v, u]$ . On (left) we illustrate that the integrability condition in 2D amounts to  $v \cdot e = u \cdot e$  (as in [15]). On (right) we move to 3D, noting that there are now 2 linear constraints ( $v \cdot e_1 = u \cdot e_1$  and  $v \cdot e_2 = u \cdot e_2$ ) per shared face needed to enforce integrability.

## 3 Algorithmic Applications

Definition 1 suggests a linear operator  $C : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^{k-1}$  such that if  $C(v, w) = 0$  for all adjacent field pairs  $(v, w) \in \mathbf{w}$ , then  $\mathbf{w}$  is integrable. This provides a convex expression for recovering the integrable component  $\mathbf{w} + \delta$  of an input vector field  $\mathbf{w}$ :

$$\pi_{\text{hodge}}(\mathbf{w}) = \arg \min_{\delta} \frac{1}{2} \|\delta\|^2 \quad \text{s.t.} \quad C(\mathbf{w} + \delta) = 0 \quad (1)$$

However, suppose we want to preserve the behavior of streamlines in our projection  $\mathbf{w}$ . This can be expressed as Eq 2, now instead of finding the closest integrable field, we want

the one which is closest among all possible rescalings  $s\mathbf{w}$  of the input field  $\mathbf{w}$ . Turns out that this can be formulated as a convex problem, as demonstrated for fields on 2D surfaces in Appendix C of [21]:

$$\pi_{\text{rescale}}(\mathcal{M}, \mathbf{w}, \lambda) = \arg \min_{s, \delta} \frac{1}{2} \|\delta\|^2 + \frac{\lambda}{2} \|\nabla s\|^2 \quad \text{s.t.} \quad \begin{aligned} \|s\|^2 + \|\delta\|^2 &= 1 \\ C(s\mathbf{w} + \delta) &= 0, \end{aligned} \quad (2)$$

Integrable projection can also be used to recover good solutions to non-convex problems. Suppose for instance that we want to recover fields which are everywhere approximately geodesic [16]. This is equivalent to asking for fields which are everywhere integrable and approximately unit, and smooth solutions can be recovered efficiently in practice (see Algorithm 1 in [21]). A related algorithm for fields in  $R^3$  was presented in [23], where the authors present a  $\Gamma$ -convergence proof of its convergence in the sharp  $\lambda \rightarrow 0$  limit.

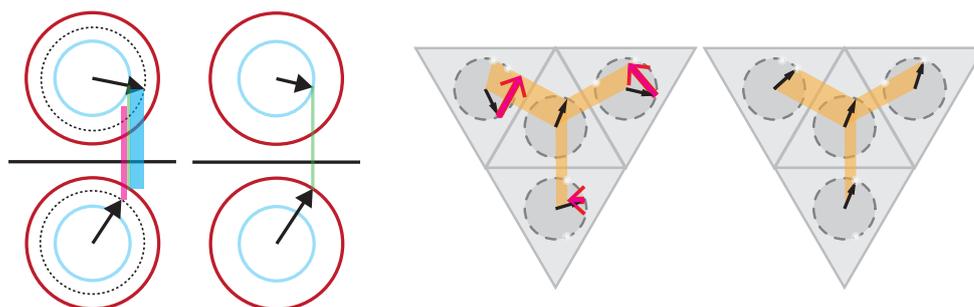
$$\pi_{\text{geodesic}}(\mathcal{M}, \lambda) = \arg \min_{\mathbf{w}, \delta} \frac{1}{2} \|\delta\|^2 + \frac{\lambda}{2} \|\nabla(\mathbf{w} + \delta)\|^2 \quad \text{s.t.} \quad \begin{aligned} C(\mathbf{w} + \delta) &= 0 \\ \|\mathbf{w}_i\| &= 1, \end{aligned} \quad (3)$$

Incidentally, this gives solutions similar to minimizers the Aviles-Giga functional [2], over manifolds in all dimensions. To see this, let  $\nabla \mathbf{u} := \mathbf{w} + \delta$ . We've constrained  $\mathbf{w} + \delta$  to be integrable, and thus we can assume that the scalar field  $\mathbf{u}$  exists. From here we can almost write Eq. 3 as Eq. 4, which would match the form presented in [9] (because at optimality  $\delta$  and  $\mathbf{w}$  must be colinear):

$$\pi_{\text{aviles-giga}}(\mathcal{M}, \lambda) = \arg \min_{\mathbf{u}} \frac{1}{2} (\|\nabla \mathbf{u}\|^2 - 1)^2 + \frac{\lambda}{2} \|\nabla(\nabla \mathbf{u})\|^2 \quad (4)$$

Minimizers of such functionals can form non-trivial topologies [11, 14] which may be of independent interest to the SoCG community.

Going forward, one problem of interest is integrable projection in the presence of nonlinear constraints beyond the pointwise unit norm in Eq 3. One candidate question is projection to integrable frame fields, using representations like the ones studied in [12].



■ **Figure 2** Minimizers of Equation 2 (*left*) recover fields close to a fixed direction field, but allow the norm to vary arbitrarily. Minimizers of Equations 3/4 (*right*) are everywhere close to unit.

---

## References

- 1 Rahul Arora, Alec Jacobson, Timothy R. Langlois, Yijiang Huang, Caitlin Mueller, Wojciech Matusik, Ariel Shamir, Karan Singh, and David I. W. Levin. Volumetric michell trusses for

- parametric design fabrication. In *Proceedings of the ACM Symposium on Computational Fabrication*, SCF '19, New York, NY, USA, 2019. Association for Computing Machinery. doi:10.1145/3328939.3328999.
- 2 Patricio Aviles and Yoshikazu Giga. A mathematical problem related to the physical theory of liquid crystal configurations. In *Miniconference on geometry/partial differential equations, 2*, pages 1–16, Canberra AUS, 1987. Centre for Mathematical Analysis, The Australian National University. URL: <https://projecteuclid.org/euclid.pcma/1416336633>.
  - 3 H. Bhatia, G. Norgard, V. Pascucci, and P. Bremer. The helmholtz-hodge decomposition—a survey. *IEEE Transactions on Visualization and Computer Graphics*, 19(8):1386–1404, 2013.
  - 4 Jason Cantarella, Dennis DeTurck, and Herman Gluck. Vector calculus and the topology of domains in 3-space. *The American Mathematical Monthly*, 109(5):409–442, 2002. URL: <http://www.jstor.org/stable/2695643>.
  - 5 Fernando de Goes, Mathieu Desbrun, and Yiying Tong. Vector field processing on triangle meshes. In *ACM SIGGRAPH 2016 Courses*, SIGGRAPH '16, pages 27:1–27:49, New York, NY, USA, 2016. ACM. URL: <http://doi.acm.org/10.1145/2897826.2927303>, doi:10.1145/2897826.2927303.
  - 6 Jeff Erickson. Efficiently hex-meshing things with topology. In *Proceedings of the Twenty-Ninth Annual Symposium on Computational Geometry*, SoCG '13, page 37–46, New York, NY, USA, 2013. Association for Computing Machinery. doi:10.1145/2462356.2462403.
  - 7 H. Helmholtz. Über integrale der hydrodynamischen gleichungen, welche den wirbelbewegungen entsprechen. *Journal für die reine und angewandte Mathematik*, 1858(55), 1858.
  - 8 Felix Knöppel, Keenan Crane, Ulrich Pinkall, and Peter Schröder. Stripe patterns on surfaces. *ACM Trans. Graph.*, 34:1–11, 2015.
  - 9 Robert Kohn. Energy-driven pattern formation. *Proceedings of the International Congress of Mathematicians, Vol. 1, 2006-01-01, ISBN 978-3-03719-022-7, pags. 359-383*, 1, 01 2006.
  - 10 Heng Liu, Paul Zhang, Edward Chien, Justin Solomon, and David Bommes. Singularity-constrained octahedral fields for hexahedral meshing. *ACM Trans. Graph.*, 37(4), July 2018. doi:10.1145/3197517.3201344.
  - 11 Thomas Machon, Hillel Aharoni, Yichen Hu, and Randall Kamien. Aspects of defect topology in smectic liquid crystals. *Communications in Mathematical Physics*, 02 2019. doi:10.1007/s00220-019-03366-y.
  - 12 David N Palmer, David Bommes, and Justin Solomon. Algebraic representations for volumetric frame fields. *ArXiv*, abs/1908.05411, 2019.
  - 13 Konstantin Poelke and Konrad Polthier. Boundary-aware hodge decompositions for piecewise constant vector fields. *Computer-Aided Design*, 78:126 – 136, 2016. SPM 2016. URL: <http://www.sciencedirect.com/science/article/pii/S0010448516300240>, doi:<https://doi.org/10.1016/j.cad.2016.05.004>.
  - 14 Joseph Pollard, Gregor Posnjak, Simon Čopar, Igor Mušević, and Gareth Alexander. Point defects, topological chirality, and singularity theory in cholesteric liquid-crystal droplets. *Physical Review X*, 9, 04 2019. doi:10.1103/PhysRevX.9.021004.
  - 15 Konrad Polthier and Eike Preuß. Identifying vector field singularities using a discrete hodge decomposition. In Hans-Christian Hege and Konrad Polthier, editors, *Visualization and Mathematics III*, pages 113–134, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.
  - 16 Helmut Pottmann, Qixing Huang, Bailin Deng, Alexander Schiftner, Martin Kilian, Leonidas Guibas, and Johannes Wallner. Geodesic patterns. *ACM Trans. Graph.*, 29(4):43:1–43:10, July 2010. URL: <http://doi.acm.org/10.1145/1778765.1778780>.
  - 17 Faniry Razafindrazaka, Konstantin Poelke, Konrad Polthier, and Leonid Goubergrits. A consistent discrete 3d hodge-type decomposition: implementation and practical evaluation, 11 2019.
  - 18 Andrew O. Sageman-Furnas, Albert Chern, Mirela Ben-Chen, and Amir Vaxman. Chebyshev nets from commuting polyvector fields. *ACM Trans. Graph.*, 38(6), November 2019. doi:10.1145/3355089.3356564.

- 19 Guenter Schwarz. Hodge decomposition - a method for solving boundary value problems. 1995.
- 20 Amir Vaxman, Marcel Campen, Olga Diamanti, David Bommes, Klaus Hildebrandt, Mirela Ben-Chen Technion, and Daniele Panozzo. Directional field synthesis, design, and processing. In *ACM SIGGRAPH 2017 Courses*, SIGGRAPH '17, New York, NY, USA, 2017. Association for Computing Machinery. doi:10.1145/3084873.3084921.
- 21 Josh Vekhter, Jiacheng Zhuo, Luisa F Gil Fandino, Qixing Huang, and Etienne Vouga. Weaving geodesic foliations. *ACM Trans. Graph.*, 38(4), July 2019. doi:10.1145/3306346.3323043.
- 22 Ryan Viertel and Braxton Osting. An approach to quad meshing based on cross valued maps and the ginzburg-landau theory.
- 23 Shawn Walker and Wujun Zhang. A finite element method for nematic liquid crystals with variable degree of orientation. *SIAM Journal on Numerical Analysis*, 55, 12 2015. doi:10.1137/15M103844X.
- 24 Rundong Zhao, Mathieu Desbrun, Guo-Wei Wei, and Yiyong Tong. 3d hodge decompositions of edge- and face-based vector fields. *ACM Trans. Graph.*, 38(6), November 2019. doi:10.1145/3355089.3356546.

# Computing relevant subtrajectory bundles faster

Erfan Hosseini Sereshgi 

Department of Computer Science, Tulane University, United States  
shosseinisereshgi@tulane.edu

Carola Wenk 

Department of Computer Science, Tulane University, United States  
cwenk@tulane.edu

**2012 ACM Subject Classification** Information systems → Geographic information systems;  
Theory of computation → Computational geometry

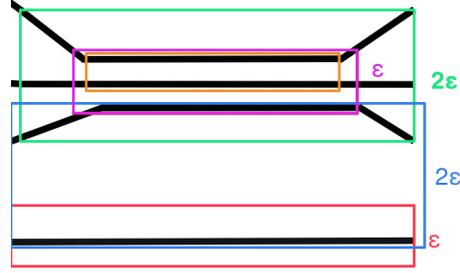
**Keywords and phrases** Trajectories, map construction, clustering, geometric algorithms

**Acknowledgements** This research was partially supported by the National Science Foundation under grant CCF 1637576. We also thank Kevin Buchin for fruitful discussions.

## 1 Introduction

The ubiquitous use of smartphones and other GPS-enabled devices leads to vast amounts of tracking data being collected on a regular basis. Such data captures, for example, the movement of cars driving on road networks, people hiking in remote areas, or animals in their habitat. One of the plentiful uses of this data is for map construction [1] or map updates, especially in areas with frequent road changes. But generally, processing large GPS trajectory data requires clustering contiguous portions of the trajectories together; this is referred to as subtrajectory clustering. There are many underlying challenges, such as determining and constructing the road that a subtrajectory bundle corresponds to. But a core challenge is to determine the exact portions of the trajectories that are similar. It has been shown that subtrajectory clustering with respect to the Fréchet distance is NP-complete [4], however the authors gave a polynomial-time 2-approximation algorithm that is based on sweeping the free space diagram (which is usually used for deciding the Fréchet distance [2]) to identify sets of subtrajectories within Fréchet distance  $\varepsilon$ , for given  $\varepsilon > 0$ .

Let  $\tau$  be a set of input trajectories. A subtrajectory bundle is a set of subtrajectories from the trajectories in  $\tau$ . There are three parameters for such bundles: the *length* ( $l$ ) of the longest trajectory in the bundle, the *size* ( $k$ ) denoting the number of trajectories in the bundle, and the *spatial proximity* ( $\varepsilon$ ) defined as the the maximum distance between any two (sub)trajectories in the bundle. See Figure 1. We call a subtrajectory bundle with these parameters a  $(k, l, \varepsilon)$ -bundle. Since this is a multi-criteria optimization problem, there are many potential bundles of interest and it is unclear which ones to choose. Buchin *et al.* [3] proposed an approach for identifying *relevant bundles* of subtrajectories, and they used those bundles for map construction. A subtrajectory bundle is *relevant*, if it is maximal, stable, and long. Here, *long* refers to maximizing the length of the subtrajectories. A bundle is *maximal* with respect to containment; a bundle  $B_1$  contains a bundle  $B_2$  if each subtrajectory in  $B_1$  contains a subtrajectory in  $B_2$ . The *lifespan*  $[\varepsilon_1, \varepsilon_2]$  of a bundle consists of all those  $\varepsilon$  for which this bundle contains the same subtrajectories ( $k$  stays the same and  $l$  only adjusts with  $\varepsilon$ ). A bundle is *stable* if  $\varepsilon_2 - \varepsilon_1 \geq \varepsilon_1$ . See Figure 1. We present an algorithm to compute stable bundles which improves the runtime of the original algorithm [3] by a factor of  $\min(\varepsilon_{max}/\log_2 \varepsilon_{max}, |\tau|)$  with  $\varepsilon_{max}$  being the maximum value for  $\varepsilon$ .



■ **Figure 1** Bundles with different parameters:  $k=1$  (red),  $k=2$  (blue, orange),  $k=3$  (pink, green). All bundles except the orange one are maximal. The green bundle is stable, the blue bundle is not.

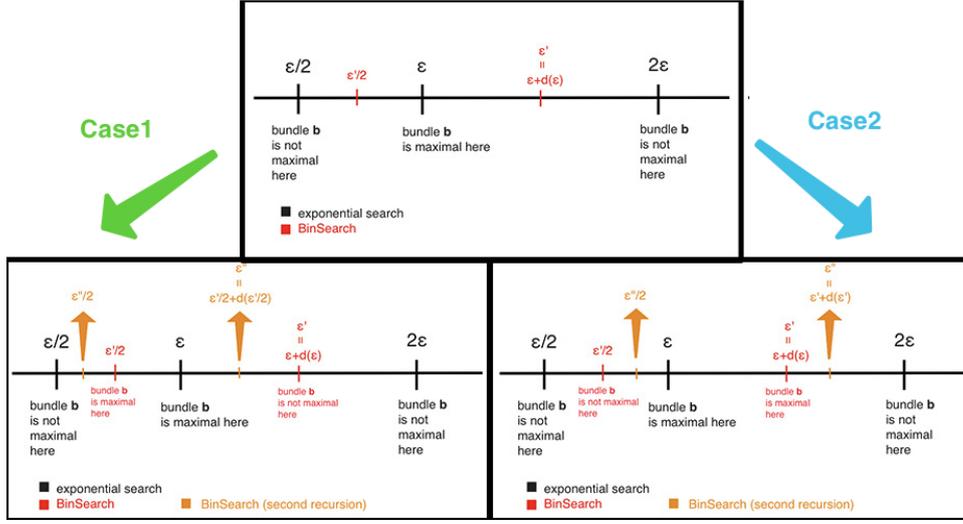
## 2 Algorithm

The original algorithm [3] computes relevant bundles in two steps. First it computes a superset of bundles by running the 2-approximation Fréchet clustering algorithm of [4] for all  $k$  and  $\varepsilon$  within a given range, maximizing  $l$  for each cluster. In a subsequent filtering step it extracts maximal and stable bundles. Here,  $k \in \{3, \dots, |\tau|\}$  and  $\varepsilon \in \mathcal{E} = \{5i \mid 1 \leq i, 5i \leq \varepsilon_{max}\}$ , where values of  $\varepsilon$  are interpreted in meters,  $|\tau|$  is the number of trajectories, and  $\varepsilon_{max}$  is an input parameter. A drawback of this approach is the exhaustive search among the large number of  $\varepsilon \in \mathcal{E}$ . Our approach reduces this number by combining exponential and binary search to avoid computing bundles that are not stable in the first place.

Our algorithm computes a set  $S$  of stable maximal bundles. We iterate in an exponential search over  $\varepsilon \leq \varepsilon_{max}$ , starting with  $\varepsilon = 5$  and doubling  $\varepsilon$  in each iteration. For fixed  $\varepsilon$  we do:

1. We compute the set  $B(\varepsilon)$  of all maximal longest bundles over all  $k \in \{3, \dots, |\tau|\}$ , as in [3].
2. For each  $b \in B(\varepsilon)$  we check if  $b \in B(2\varepsilon)$ ; then we know  $b$  is stable and we add it to  $S$ .
3. Now, for each  $b \in B(\varepsilon) \setminus S$ , we do the following: We know that  $b \in B(\varepsilon)$  but  $b \notin B(2\varepsilon)$ . So we perform a variant of binary search to find an  $\varepsilon'$  such that  $[\varepsilon'/2, \varepsilon'] \subseteq [\varepsilon_1, \varepsilon_2]$ , where  $[\varepsilon_1, \varepsilon_2]$  is the lifespan of  $b$ . Namely, we seek  $\varepsilon'$  so that  $b$  is a longest bundle for  $k = b.size()$  and  $\varepsilon'$  as well as for  $k$  and  $\varepsilon'/2$ . We say that  $b$  is stable for  $\varepsilon'$ . Let  $d(\varepsilon) = (\varepsilon - \varepsilon/2)/2$ . We start the recursive search with  $\varepsilon' := \varepsilon + d(\varepsilon)$ . First we compute the set  $L_k(\varepsilon')$  of all longest bundles for  $k$  and  $\varepsilon'$ , and the set  $L_k(\varepsilon'/2)$  using the algorithm of [4]. If  $b \in L_k(\varepsilon'/2) \cap L_k(\varepsilon')$ , then we know  $b$  is stable for  $\varepsilon'$  and add it to  $S$ . And if  $b \notin L_k(\varepsilon'/2)$  and  $b \notin L_k(\varepsilon')$  then we know  $b$  is not stable. The maximality of  $b$  can be checked by computing  $L_{k+1}(\varepsilon')$ . Since  $b$  is already maximal in  $B(\varepsilon)$  we do not have to compute  $L_{k+1}(\varepsilon'/2)$ . We continue the recursive search in the following two cases: (1) If  $b \in L_k(\varepsilon'/2)$  but  $b \notin L_k(\varepsilon')$ , then we recurse on the left with  $\varepsilon' := \varepsilon'/2 + d(\varepsilon'/2)$ . (2) If  $b \in L_k(\varepsilon')$  but  $b \notin L_k(\varepsilon'/2)$ , then we recurse on the right with  $\varepsilon' := \varepsilon' + d(\varepsilon')$ .

See Figure 2 for an illustration of the cases of the recursive search. The combination of exponential and binary search works since the bundles are monotone in  $\varepsilon$  in the sense that if a bundle  $b \in L_k(\varepsilon)$  then  $\exists b' \in L_k(\varepsilon')$  that contains  $b$  for all  $\varepsilon' > \varepsilon$ . Such a bundle  $b'$  can be identified in linear time. One can run the binary search until the desired precision of  $\varepsilon$  is achieved. But if we stop it at  $\varepsilon$  values that are multiples of five, then our algorithm returns the same bundles as the original algorithm.



■ **Figure 2** The initial call to binary search and the two recursive cases.

### 3 Runtime

Let  $\mathcal{E} = \{5^i \mid 1 \leq i, 5^i \leq \epsilon_{max}\}$  be the set of  $\epsilon$  values that the algorithm in [3] uses to compute the bundles. Similarly we define the set  $\mathcal{E}'$  which contains all  $\epsilon$  values that we iterate over in our algorithm. From the exponential search that we employ follows that  $\mathcal{E}' = \{5 \cdot 2^i \mid i = 1 \dots \log_2 \epsilon_{max}\}$ . Thus,  $|\mathcal{E}| = \frac{\epsilon_{max}}{5}$  and  $|\mathcal{E}'| = \log_2 \frac{\epsilon_{max}}{5}$ .

Computing  $L_k(\epsilon)$  takes  $O(|\tau|^2 M^2)$  time [4], where  $M$  is the maximum number of edges per trajectory, over all trajectories. Computing all maximal longest bundles  $B(\epsilon)$  for all  $k$  adds an additional  $|\tau|$  factor, which takes  $O(|\tau|^3 M^2)$  time.

The original algorithm computes  $B(\epsilon)$  for all  $k$  and for all  $\epsilon \in \mathcal{E}$  yields a runtime of  $O(\epsilon_{max} |\tau|^3 M^2)$ . The stable bundles are computed in a postprocessing step. Our algorithm only computes  $B(\epsilon)$  for all  $\epsilon \in \mathcal{E}'$ , and in the worst-case it computes  $L_k(\epsilon)$  twice for all  $\epsilon \in \mathcal{E} \setminus \mathcal{E}'$ . Thus the total runtime is  $O(\log_2 \epsilon_{max} |\tau|^3 M^2 + \epsilon_{max} |\tau|^2 M^2) = O(\max(|\tau| \log_2 \epsilon_{max}, \epsilon_{max}) |\tau|^2 M^2)$ . Compared to the original algorithm this is an improvement of a factor of  $\min(\epsilon_{max} / \log_2 \epsilon_{max}, |\tau|)$ . We are currently working on further improving the runtime by avoiding recomputing  $L_k(\epsilon)$  from scratch for each value of  $\epsilon$ , but rather trace the bundles for varying values of  $\epsilon$ .

### References

- 1 Mahmuda Ahmed, Sophia Karagiorgou, Dieter Pfoser, and Carola Wenk. Map construction algorithms. 2015.
- 2 Helmut Alt and Michael Godau. Computing the Fréchet distance between two polygonal curves. *International Journal of Computational Geometry & Applications*, 5:75–91, 1995.
- 3 Kevin Buchin, Maike Buchin, David Duran, Brittany Terese Fasy, Roel Jacobs, Vera Sacristan, Rodrigo I. Silveira, Frank Staals, and Carola Wenk. Clustering trajectories for map construction. In *Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, Redondo Beach, CA, USA, 2017.
- 4 Kevin Buchin, Maike Buchin, Joachim Gudmundsson, Maarten Löffler, and Jun Luo. Detecting commuting patterns by clustering subtrajectories. *Int. J. Comput. Geometry App.*, 21(3):253–282, 2011.