# Barycentric coordinate neighbourhoods in Riemannian manifolds

Ramsay Dyer* Gert Vegter† Mathijs Wintraecken ‡


## Abstract

We quantify conditions that ensure that a signed measure on a Riemannian manifold has a well defined centre of mass. We then use this result to quantify the extent of a neighbourhood on which the Riemannian barycentric coordinates of a set of $n+1$ points on an $n$-manifold provide a true coordinate chart, i.e., the barycentric coordinates provide a diffeomorphism between a neighbourhood of a Euclidean simplex, and a neighbourhood containing the points on the manifold.


## 1 Introduction and Preliminaries

In this work $M$ is a smooth ($C^\infty$) Riemannian manifold (without boundary) of dimension $n$. A function defined on a closed set $A \subset M$ is *smooth* if it can be extended to a smooth function on an open neighbourhood of $A$. The geodesic distance between $x, y \in M$ is denoted $d_M(x, y)$, and $B_M(c, r) = \{x \in M \mid d_M(c, x) < r\}$ is the open geodesic ball of radius $r$ centred at $c \in M$. The topological closure of a set $B \subset M$ is denoted $\overline{B}$. A geodesic is *minimising* if it is a shortest path between any two of its points. The injectivity radius $\iota_M$ of $M$ is the supremum of the distances $r$ such that any two points $x, y \in M$ with $d_M(x, y) < r$ have a unique minimising geodesic connecting them.

### 1.1 Convexity

A set $B \subseteq M$ is *convex* if for all $x, y \in B$ there is a unique minimising geodesic in $M$ connecting $x$ to $y$ within $B$, and this geodesic is the only one connecting $x$ and $y$ contained in $B$. If $\Lambda_u$ is an upper bound on the sectional curvatures of $M$, then for any point $c \in M$ the ball $\overline{B}_M(c, r)$ is convex if $r < \rho_0$ with

$$\rho_0 = \min\left\{\frac{\iota_M}{2}, \frac{\pi}{2\sqrt{\Lambda_u}}\right\}, \tag{1}$$

where we set $\frac{\pi}{2\sqrt{\Lambda_u}} = \infty$ if $\Lambda_u \leq 0$ (see [1, Thm. IX.6.1]). A function $g\colon A \to \mathbb{R}$, with $A \subseteq M$ is *convex* if for any $p \in A$ and geodesic $\gamma$ with $\gamma(0) = p$, we have $\frac{d^2}{dt^2} g(\gamma(t))|_{t=0} \geq 0$. We say $g$ is *strictly convex* if this inequality is strict.

### 1.2 Riemannian centre of mass

Let $\mu$ be a (unsigned) measure whose support is contained within a convex geodesic ball $B_\rho \subseteq M$ with radius $\rho$, and assume $\mu(B_\rho) = 1$. Consider the energy function $\mathcal{E}_\mu\colon \overline{B}_\rho \to \mathbb{R}$ defined by

$$\mathcal{E}_\mu(x) = \frac{1}{2}\int d_M(x, y)^2 \, d\mu(y), \tag{2}$$

where $d\mu(y)$ indicates that the integration is with respect to the variable $y$, and the domain of integration is understood to be $M$, or equivalently $B_\rho$, since it contains the support of $\mu$. Karcher [4, Theorem 1.2] showed that $\mathcal{E}_\mu$ has a unique minimum on $B_\rho$, under the following condition: If there are positive sectional curvatures in $B_\rho$, the radius satisfies $\rho \leq \frac{\pi}{4\sqrt{\kappa}}$, where $\kappa$ is an upper bound on the sectional curvatures in $B_\rho$. This unique minimum is called the *Riemannian centre of mass* of $\mu$.

In the case of a discrete measure we write

$$\mathcal{E}_\lambda(x) = \frac{1}{2}\sum_i \lambda_i d_M(x, p_i)^2,$$

and identify the $\lambda_i$ with barycentric coordinates and call the points $p_i$ vertices, see Section 3 and [2].

### 1.3 Contribution and previous work

This abstract is based in part on Section 3.4 of the PhD-thesis [7], where the proof of Theorem 1 can be found. This is an alternate proof of the existence and uniqueness of Riemannian centres of mass. Our proof is based on the Toponogov comparison theorem, which compares geodesic triangles on manifolds to geodesic triangles on spaces of constant curvature. It is elementary in the sense that, if we take the Toponogov comparison theorem for granted, it follows by applying Taylor's theorem to the cosine rules for spaces of constant curvature, see [7, Section 3.4.3]. Our result can accommodate mass distributions that include negative weights.

Sander [6] has recently published such a result. One of the motivations for our new proof is that it allows us to quantify the extent of barycentric coordinate neighbourhoods on manifolds, which is not easily obtained from Sander's result.

Our work was inspired by questions regarding the interconnection of Riemannian simplices, that are important in manifold meshing. Here negative barycentric

*INRIA Sophia-Antipolis yasmar@gmail.com

†Rijksuniversiteit Groningen, g.vegter@rug.nl

‡INRIA Sophia-Antipolis, supported by FET-Open grant 255827 and the Ghudi project, m.h.m.j.wintraecken@gmail.com

coordinates simplify matters significantly. We also envision applications in the study of intrinsic polytopes and (mathematical) physics.

There are two traditional proofs for the existence and uniqueness of Riemannian centres of mass with positive weights. A first one by Karcher [4] and a later one by Kendall [5]. Karcher used comparison theorems of Rauch type to prove that the energy function (2) has a unique minimum. Comparison theorems of Rauch type compare the behaviour of vector fields on spaces of arbitrary curvature to the behaviour of vector fields on spaces of constant curvature. Our conditions coincide for positive weights. The proof by Kendall [5] is based on the concept of 'convex geometry' used in the context of Dirichlet problems. The definition of 'convex geometry' does not resemble the usual definition of convexity, see [7, Section 3.4.1] for a discussion.

### 1.4 Signed measures

In this work we are only concerned with finite (i.e., bounded) Borel measures. We will be considering signed measures $\mu$ on $M$. The Jordan decomposition theorem [3, §29], states that there are unique unsigned measures $\mu_+$ and $\mu_-$ such that $\mu = \mu_+ - \mu_-$, and for our purposes we can take this as a definition of a signed measure. If $\nu$ is an unsigned measure on $M$, then the *support* ($\mathrm{supp}(\nu)$) of $\nu$ is the set of points in $M$ for which every open neighbourhood has positive measure. The support of a signed measure $\mu$ is defined by $\mathrm{supp}(\mu) = \mathrm{supp}(\mu_+) \cup \mathrm{supp}(\mu_-)$.

## 2 Center of mass of signed measures

We consider a signed measure $\mu$ with support contained in a convex ball $B_\rho \subset M$. In this section we give conditions on $\mu$ and $\rho$ that ensure that $\mathcal{E}_\mu$ has a unique minimum.

**Theorem 1** *Let $M$ be a manifold whose sectional curvatures $K$ are bounded by $\Lambda_\ell \leq K \leq \Lambda_u$, and let $\mu$ be a signed measure on $M$, whose support is contained in a geodesic ball $B_M(c,r)$. Suppose $B_\rho = B_M(c,\rho)$, with $r < \rho < \rho_0$ as defined in* (1).

*Then $\mathcal{E}_\mu \colon \overline{B}_\rho \to \mathbb{R}$ has a unique minimum in $B_\rho$ if $(\rho - r)\mu_+(M) - (\rho + r)\mu_-(M) > 0$, and*

$$\frac{\vartheta_u}{\tan\vartheta_u}\mu_+(M) - \mu_-(M) > 0 \quad \text{if } 0 \leq \Lambda_\ell \leq \Lambda_u,$$

$$\frac{\vartheta_u}{\tan\vartheta_u}\mu_+(M) - \frac{\vartheta_\ell}{\tanh\vartheta_\ell}\mu_-(M) > 0 \quad \text{if } \Lambda_\ell \leq 0 \leq \Lambda_u,$$

$$\mu_+(M) - \frac{\vartheta_\ell}{\tanh\vartheta_\ell}\mu_-(M) > 0 \quad \text{if } \Lambda_\ell \leq \Lambda_u \leq 0,$$

*where $\vartheta_\ell = 2\rho\sqrt{|\Lambda_\ell|}$, and $\vartheta_u = 2\rho\sqrt{|\Lambda_u|}$.*

Which, under stronger conditions, simplifies to:

**Corollary 2** *If we strengthen the assumptions of Theorem 1 to $|K| \leq \Lambda$, and assume $\mu(M) = 1$, the function $\mathcal{E}_\mu$ has a unique minimum in $B_\rho$ if $\rho > (1 + 2\mu_-(M))r$, and $\rho < \frac{\pi}{4\sqrt{\Lambda}}\left(1 + 3\mu_-(M)\right)^{-1}$.*

## 3 Barycentric coordinate neighbourhoods

Let $M$, in this section, be an $n$-dimensional Riemannian manifold with sectional curvatures $K$ bounded by $|K| \leq \Lambda$, and let $\sigma \subset M$ be a set of $n+1$ points such that $d_M(p,q) < L$ for any $p, q \in \sigma$. Using $\sigma$ we can define an Riemannian simplex $\sigma_M$ by

$$\mathcal{B}_{\sigma^n} : \sigma \to M : \lambda \mapsto \operatorname*{argmin}_{x \in \overline{B}_\rho} \mathcal{E}_\lambda(x),$$

where $\mathbf{\Delta}^n$ is the standard $n$-simplex and $\lambda$ denotes its barycentric coordinates. In [2, Prop. 29] we have given conditions that ensure that the barycentric coordinate map is non-degenerate, combining this with the result from the previous section we can extend the barycentric coordinates to a (large) neighbourhood of the simplex, to be precise:

**Theorem 3 (Barycentric coordinates)** *Given a scale parameter $s \geq 1$, if $\sigma$ defines a Euclidean simplex $\tilde{\sigma} \subset \mathbb{E}^n$ with the same edge lengths -that is the geodesic distances between the vertices of $\sigma$-, and with thickness $t$ satisfying $t^2 \geq 25s^2L\sqrt{\Lambda}$, then the barycentric coordinate map $b$: $B_{\mathbb{E}^n}(\tilde{v}, sL) \to B_\rho$, where $\tilde{v}$ is any vertex of $\tilde{\sigma}$, is an embedding. The ball $B_\rho$ is centred on the vertex of $\sigma$ corresponding to $\tilde{v}$, and the radius $\rho$ can be chosen in the non-empty interval given by*

$$\left(1 + \frac{2s}{t}\right)L < \rho < \frac{\pi}{4\sqrt{\Lambda}}\left(1 + 3\frac{s}{t}\right)^{-1}.$$

## References


[1] I. Chavel. *Riemannian Geometry: A Modern Introduction*. Cambridge University Press, 2006.

[2] R. Dyer, G. Vegter, and M. Wintraecken. Riemannian simplices and triangulations. *Geometriae Dedicata*, 179(1):91–138, 2015.

[3] P. R. Halmos. *Measure theory*. Springer, 1974.

[4] H. Karcher. Riemannian center of mass and mollifier smoothing. *Communications on Pure and Applied Mathematics*, 30:509–541, 1977.

[5] W. Kendall. Probability, convexity, and harmonic maps with small image I: Uniqueness and fine existence. *Procedings of the London Mathematical society*, s3-61 (Issue 2):371–406, 1990.

[6] O. Sander. Geodesic finite elements of higher order. *IMA Journal of Numerical Analysis*, 36(1):238–266, 2016.

[7] M. Wintraecken. *Ambient and intrinsic triangulations and topological methods in cosmology*. PhD thesis, Rijksuniversiteit Groningen, 2015.

# Spectral Properties of Distance Matrices of High Dimensional Mixtures

Jean-Daniel Boissonnat * David Cohen-Steiner † Alba Chiara De Vitis ‡

**Abstract**

We use spectral analysis of distance matrices of high dimensional mixtures to learn a mixture of distributions in $\mathbb{R}^n$. Our approach focuses on high-dimensions and uses concentration of measure. It applies to any distribution with concentration properties.

## 1 Introduction

A mixture of $k$ distributions $F_1, ..., F_k$ with mixing weights $w_1, ..., w_k$, where $\sum_i w_i = 1$, is the distribution in which each sample is drawn from $F_i$ with probability $w_i$. Learning the mixture consists in identifying the parameters of the distributions from the sample (e.g. the mean value $\mu_i$ and the covariance matrix in the Gaussian case). Several approaches have been proposed to solve this problem, most notably classification methods, based on clustering data points, and moment based methods, whose aim is to find parameters so that the mixture distribution has moments approximately matching the observed empirical moments. The latter class is strongly tied to the Gaussian case and usually leads to very hard computational problems.

## 2 Previous work

Our method focuses on the classification approach, which led to many important theoretical results, most notably for the class of isotropic Gaussians. The first approach by Dasgupta used random projections onto a low dimensional space [4]. He proved that a mixture of $k$ isotropic Gaussians in $\mathbb{R}^n$, with weights $\Omega(\frac{1}{k})$ and variances within a bounded range, can be correctly learned under the separation condition (where the $\Omega^*$ notation suppresses the terms depending logarithmically on $n$),

$$|\mu_i - \mu_j| \geq (\sigma_i + \sigma_j)\Omega^*(n^{\frac{1}{2}})$$

where $\mu_i$ and $\sigma_i^2$ are respectively the mean and the variance of the components.

The result was later improved by Dasgupta and Shulman [5] using a variant of the EM algorithm, for isotropic Gaussians, and by Arora and Kannan [2], using a distance-based algorithm, for general Gaussians, leading to separation bound $\Omega^*(n^{\frac{1}{4}})$.

Vempala and Wang [12] improved on this result, for isotropic Gaussians, avoiding the dependency on the dimension. They used spectral projection to obtain a dimension-free separation bound :

$$|\mu_i - \mu_j| \geq (\sigma_i + \sigma_j)\Omega^*(k^{\frac{1}{4}}).$$

Since $k$ is usually much smaller than $n$, this is a substantial improvement. Kannan et al. [7] extended this approach to a mixture of general Gaussians obtaining a separation condition polynomial in $k$. Achlioptas and McSherry further improved on the polynomial in $k$ in [1].

Finally, a method called *Isotropic PCA* has been proposed by Brubaker and Vempala [3]. Their method uses conditions much weaker then the previous ones and improves on known results for general Gaussians.

## 3 Main result

We present a new approach for clustering mixtures of distributions in high dimensional Euclidean spaces using spectral properties of the distance matrix of the data. We exploit the concentration of measure phenomena in a new way in order to analyze mixtures of distributions having concentration properties. This class includes many high dimensional distributions, *e.g.* general Gaussians with bounded anisotropy and, by the thin-shell conjecture, uniform distributions on convex isotropic bodies.

For a metric measure space $(X, d, \mu)$, concentration of measure refers to the tendency of Lipschitz functions to concentrate around their median [11],[10],[9],[6],[8]. A function $f$ has $\sigma$-concentration on $X$ if, for some real non negative parameters $C$ and $p$

$$\mu_X\{x : |f(x) - M(f)| \geq \epsilon\} \leq Ce^{-\frac{\epsilon^p}{\sigma^p}}.$$

A space $(X, d, \mu)$ has $\sigma$-concentration if all 1-Lipschitz functions have $\sigma$-concentration. Levy's lemma gives concentration rates for the $n$-dimensional sphere and the $n$-dimensional Gaussian space.

Our main technical result is the description of the fine structure of the normalized distance matrix $D(X_N)$, for a sample $X_N$ from the mixture. By concentration of distance functions, $D(X_N)$ is known to have a $k \times k$ block structure, with entries almost constant in each block. We further characterize each block entry using concentration properties of auxiliarily defined maps. In

*INRIA Sophia-Antipolis, jean-daniel.boissonnat@inria.fr
†INRIA Sophia-Antipolis, david.cohen-steiner@inria.fr
‡INRIA Sophia-Antipolis, alba.de-vitis@inria.fr

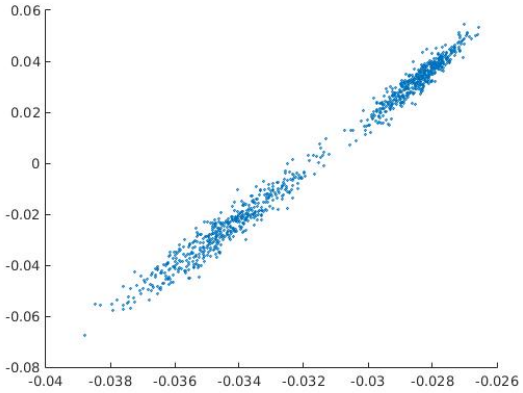



Figure 1: Projection of a sample from a mixture of two isotropic Gaussians, with same centre and $\sigma_1 = 10$ and $\sigma_2 = 15$ for $N = 1000$ $d = 100$, on the first two top singular vectors space of the distance matrix.

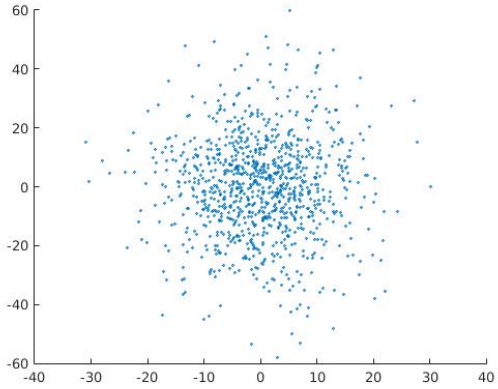



Figure 2: Projection of the sample from Fig.1 on the first two principal directions of the dataset.

particular, we prove that each block is almost rank one up to a small error. For the specific case of a mixture of isotropic Guassians in $\mathbb{R}^n$ , the entries of each block in $D(X_N)$ are constant within a $\sigma$ range and we show that each block is rank one up to $O\left(\frac{\sigma}{\sqrt{n}}\right)$. Then, using this structure, we are able to provide bounds on the singular values of $D(X_N)$.

Based on this result, we produce a simple algorithm to cluster the data. Finally, we provide theoretical guarantees for the classification of the points, assuming a gap condition on the singular spectrum of the distance matrix. While these theoretical guarantees, as is, are not comparable in general with the state of the art, the approach contains new ingredients and may lead to new improvements.

In fact, for some specific cases, we outperform existing spectral methods - that use PCA on the dataset - and Isotropic PCA [3] - which requires that there exists a hyperplane separating the components. Our method is, for example, able to separate a mixture of two Gaussians with different variances that have the same centre (*Figures 1*, *2*). The performance in this setting improves as the ratio between the variances increases.

While remaining comparable to Arora and Kannan [2], that improve on existing methods for a mixture of Gaussians with different variances, our approach is different and extends spectral analysis to distance based methods.

## Acknowledgment

This work is partially supported by the Advanced Grant of the European Research Council GUDHI (Geometric Understanding in Higher Dimensions).

## References

[1] D. Achlioptas and F. McSherry. On spectral learning of mixture of distributions. *In Proc. of COLT*, 2005.

[2] S. Arora and R. Kannan. Learning mixture of separated non-spherical gaussians. *The Annals of Applied Probability 2005, Vol. 15, Institute of Mathematical Statistics*, 2005.

[3] S. Brubaker and S.Vempala. Isotropic PCA and affine-invariant clustering. *Building Bridges (Ed.s M. Grotchel and G.O.H.Katona), Bolyai Society Mathematical Studies, Proc. of FOCS 2008.*

[4] S. Dasgupta. Learning mixtures of gaussians. *In Proceedings of the 40th Annual IEEE Symposium on Foundations of Computer Science, FOCS 99*, 1999.

[5] S. Dasgupta and L. Schulman. A two-round variant of em for gaussian mixtures. *Uncertainty in Artificial Intelligence*, 2000.

[6] M. Gromov. *Metric structures for Riemannian and non-Riemannian spaces*. Birkhäuser Verlag Basel, 1999.

[7] R. Kannan, S. Vempala, and H. Salmasian. The spectral method for general mixture models. *Proc. of the 18th Conference on Learning Theory, 2005.SIAM J. Computing,*, 2008.

[8] M. Ledoux. *The Concentration of Measure Phenomenon*. AMS Mathematical Surveys & Monographs, Providence, 2001.

[9] P. Lévy and F. Pellegrino. *Problèmes concrets d'analyse fonctionnelle.* Collection de monographies sur la théorie des fonctions Gauthier-Villars Paris, 1951.

[10] V. Milman. Asymptotic properties of functions of several variables that are defined on homogeneous spaces. *Soviet Math. Dokl. 12*, pages 1277–1281, 1971.

[11] V. Milman. A certain property of functions defined on infinite-dimensional manifolds. *Dokl. Akad. Nauk SSSR 200*, pages 781–784, 1971.

[12] S. Vempala and G. Wang. A spectral algorithm for learning mixture models. *Journal of Computer and System Sciences.*

# Recognition of the Spherical Laguerre Voronoi Diagram*

Supanut Chaidee* Kokichi Sugihara†


## Abstract

We construct an algorithm for judging whether or not a given tessellation on a sphere is a spherical Laguerre Voronoi diagram. This algorithm is based on the properties of polyhedra corresponding to the spherical Laguerre Voronoi diagram and their transformation in projective spaces.


## 1 Introduction

The inverse Voronoi diagram problem is to judge whether a given tessellation is a Voronoi diagram and, if not, find the best fit Voronoi diagram. This problem is useful for analyzing polygon-like patterns found in the real world. We focus on the Laguerre Voronoi diagram introduced by [4, 1], the weighted-Voronoi diagram whose edges are straight lines.

Not many studies focus on the inverse Laguerre Voronoi diagram problem. The Laguerre recognition problem in the plane was studied by Duan et al. [3], and the Laguerre approximation problem using planar tomographic image was done by [5].

The concept of the Laguerre Voronoi diagram was extended to the diagram on the sphere called the spherical Laguerre Voronoi diagram (SLVD) [6]. Let $U$ be a unit sphere of $\mathbb{R}^3$ centered at the origin. The spherical circle centered at $p_i \in U$ is defined by $\tilde{c}_i = \{p \in U \mid \tilde{d}(p_i, p) = r_i\}$, where $\tilde{d}(p_i, p)$ is the geodesic distance between $p_i$ and $p$, and $r_i$ is interpreted as the weight of point $p_i$. We define the Laguerre Proximity by $\tilde{d}_L(p, \tilde{c}_i) = \cos\tilde{d}(p, p_i)/\cos r_i$. The SLVD $\mathcal{L}$ is formed by a set of circles $\tilde{G} = \{\tilde{c}_1, ..., \tilde{c}_n\}$ on $U$ with bisector of $\tilde{c}_i, \tilde{c}_j$ defined by $B_L(\tilde{c}_i, \tilde{c}_j) = \{p \in U \mid \tilde{d}_L(p, \tilde{c}_i) = \tilde{d}_L(p, \tilde{c}_j)\}$. The SLVD is constructed by the intersection of all halfspaces $H(\tilde{c}_i)$, where $H(\tilde{c}_i)$ is a halfspace of the plane $\pi(\tilde{c}_i)$ passing through the circle $\tilde{c}_i$ including the origin [6].

For the inverse problem of SLVD, recently, we considered SLVD approximation to tessellations where the generators of tessellation are given [2]. The study motivated us to investigate the structure of polyhedra corresponding to SLVD.

In this study, we define the transformation in projective space $P^3(\mathbb{R})$ between two polyhedra which have the same projection. Assume that we have a given 3-regular spherical tessellation $\mathcal{T} = \{T_1, ..., T_n\}$ consisting of $n$ cells, where $n \geq 3$ and $T_i$ is a convex spherical polygon. We give an algorithm for solving the inverse problem, in which we construct a polyhedron and check its consistency.

## 2 Polyhedron Transformation

Proposition 1 is directly implied from the SLVD construction [6].

**Proposition 1** *$\mathcal{L}$ is a SLVD if and only if there is a convex polyhedron $\mathcal{P}$ containing the center of the sphere whose central projection coincides with $\mathcal{L}$.*

To find the class of the polyhedra which have the same projection, we define a map $f : P^3(\mathbb{R}) \to P^3(\mathbb{R})$ by $f(\mathbf{v}) = A \cdot \mathbf{v}$, where $\mathbf{v}$ is a homogeneous coordinate representation of a point, and

$$A = \begin{pmatrix} \alpha & \beta & \gamma & \delta \\ 0 & \eta & 0 & 0 \\ 0 & 0 & \eta & 0 \\ 0 & 0 & 0 & \eta \end{pmatrix}, \ \alpha, \beta, \gamma, \delta, \eta \in \mathbb{R}, \ \alpha, \eta \neq 0.$$

This map follows the properties that $f(O) = O$, and $\mathbf{v}$ and $f(\mathbf{v})$ align on the same line passing through $O$, which implies that the transformed polyhedron $f(\mathcal{P})$ preserves the projection. We use the projective transformation because the class of transformation we need is not included in the affine transformation.

## 3 Main Algorithm

Suppose that $\mathcal{T}$ is a given spherical tessellation and $\mathcal{V}$ is the tessellation vertex set. Let $\widehat{e}_{i,j}$ be the tessellation edge separating cells $i$ and $j$, $P_{i,j}$ be the plane passing through the edge $\widehat{e}_{i,j}$, $v_{i,j,k}$ the tessellation vertex corresponding to cells $i, j, k$, and $P_i := \pi(\tilde{c}_i)$. The line of intersection of planes $P_i$ and $P_j$ is written as $\ell_{i,j}$. The line $\ell_{i,j}$ is also contained in the plane $P_{i,j}$, and hence the projection of $\ell_{i,j}$ onto the sphere coincides with the bisector. We construct a polyhedron whose projection coincides with the given tessellation by the

*This research was partly supported by the Grant-in-Aid for Basic Research No. 24360039 and Exploratory Research No.15K12067 of MEXT. S.C. is granted by MIMS Ph.D. program of Meiji University, and Royal Thai Government scholarship.

†Graduate School of Advanced Mathematical Sciences, Meiji University, Japan, {schaidee, kokichis}@meiji.ac.jp

following procedures.

**Algorithm 1 (Initial 3 planes construction)**
**Input:** a tessellation vertex $v_{i,j,k}$, edges $\widehat{e}_{i,j}$, $\widehat{e}_{j,k}$, $\widehat{e}_{i,k}$
**Output:** planes $P_i$, $P_j$, $P_k$
**Procedure:** Choose $\tilde{c}_i$ and construct the $P_i := \pi(\tilde{c}_i)$ and planes $P_{i,j}$, $P_{i,k}$, $P_{j,k}$. Find $\ell_{i,j}$ from the intersection of $P_i$ and $P_{i,j}$, and $\ell_{i,k}$ from the intersection of $P_i$ and $P_{i,k}$. Choose a point $q_j$ in polygon $j$ in such a way that $P_j$ passes through $\ell_{i,j}$ and $q_j$. Finally, find the intersection $\ell_{j,k}$ of the $P_j$ and $P_{j,k}$ and construct $P_k$ passing through the $\ell_{i,k}$ and $\ell_{j,k}$.

The next algorithm gives a method for the generation of $n$ planes.

**Algorithm 2 (Generation of $n$ planes)**
**Input:** the vertices set $\mathcal{V}$ with tessellation edges
**Output:** planes $P_1$,..., $P_n$
**Procedure:** All vertices start unmarked. Choose an arbitrary vertex $v_{i,j,k} \in \mathcal{V}$ and employ Algorithm 1 to construct $P_i, P_j, P_k$, and mark $v_{i,j,k}$. Then choose an unmarked vertex whose two planes were already constructed. Construct the third plane of that vertex using the constructed two planes in the same way as described in Algorithm 1, and change the status to a marked vertex. Repeat this process until all planes are constructed.

The following theorem states about the property of the polyhedron construction.

**Theorem 2** *There are four degrees of freedom of the construction of planes composing a polyhedron $\mathcal{P}$ with respect to the given SLVD.*

Three degrees of freedom in Theorem 2 are fixed by choosing the first circle $\tilde{c}_i$, and the other degree of freedom is fixed by choosing the adjacent plane sharing the line $\ell_{i,j}$. Hence, there are at least four degress of freedom in the choice of a polyhedron whose projection coincides with a given SLVD. This is also the upper bound although we omit the proof. This means that if $\mathcal{T}$ is SLVD, we start the processes from arbitrary first and second planes which satisfy Algorithm 1 to get the polyhedron whose projection coincides with $\mathcal{T}$, which means that Algorithm 2 gives the unique polyhedron up to the choice of 4 degrees of freedom. We implemented Algorithms 1, 2 using Wolfram Mathematica®10.3; examples of output are shown in Figure 1.

The SLVD judgement can be done using the uniqueness of the polyhedron by the following Algorithm.

**Algorithm 3 (Consistency Check)**
**Input:** the output of Algorithm 2
**Output:** judgment "true" or "false"
**Procedure:** For each unmarked vertex $v_{l,p,q}$, check whether or not the plane $P_l$ constructed in Algorithm 2 is the same as the plane $P'_l$ constructed by planes $P_p, P_q$. If it is not, terminate and regard that $\mathcal{T}$ is not SLVD. Otherwise, move to the next unmarked vertex. If all the vertices are marked, report "true" and terminate.

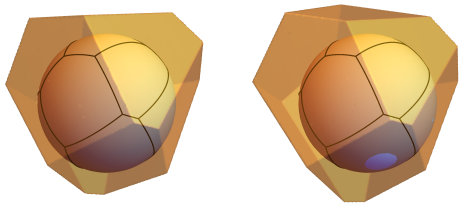


Figure 1: The figures show the different polyhedra whose projections coincide with the given SLVD.

The polyhedron constructed by Algorithm 2 does not guarantee that the planes intersect the sphere. However, if we shrink the polyhedron in such a way that all planes intersect the sphere, Proposition 1 can be applied directly, and we can conclude as follows. If $\mathcal{T}$ is the SLVD, the central projection of the polyhedron constructed in Algorithm 2 onto the sphere coincides with $\mathcal{T}$. Otherwise, the projection of the polyhedron constructed by Algorithm 2 gives us a SLVD which is different to $\mathcal{T}$.

Using the priority queue for choosing the next unmarked vertex in Algorithm 2, we get the theorem.

**Theorem 3** *For $n$ cells tessellation, the complexity of SLVD recognition problem is $O(n \log n)$.*

## 4 Future Works

The properties of the polyhedron mentioned in the recognition processes can be applied to the SLVD approximation problem which is our current study. The halfspaces composing a polyhedron constructed in Algorithm 2 can be adjusted in such a way whose projection is close to the given tessellation using optimization.

## References


[1] Aurenhammer, F.: Power Diagram: Properties, Algorithms, and Applications, SIAM J. Comput., 16, 78–96 (1987)

[2] Chaidee, S., and Sugihara, K.: Fitting Spherical Laguerre Voronoi Diagrams to Real-World Tessellations Using Planar Photographic Images, accepted in LNCS.

[3] Duan, Q., Kroese, D. P., Brereton, T., Spettl, A., Schmidt, V.: Inverting Laguerre Tessellations, Comput. J. 57, 1431–1440 (2014)

[4] Imai, H., Iri, M., Murota, K.: Voronoi Diagram in the Laguerre Geometry and its Applications. SIAM J. Comput. 14, 93–105 (1985)

[5] Spettl, A., Brereton, T., Duan, Q., Werz, T., Krill, III C. E.,Kroese, D. P. and Schmidt, V.: Fitting Laguerre tessellation approximations to tomographic image data, Philosophical Magazine, DOI: 10.1080/14786435.2015.1125540 (2016)

[6] Sugihara, K.: Laguerre Voronoi Diagram on the Sphere. J. Geom. Graph. 6, 69–81 (2002)

# Computing the Expected Area of an Induced Triangle

Vissarion Fisikopoulos* Frank Staals§ Constantinos Tsirogiannis§

## 1 Introduction

Consider the following problem: given a set $P$ of $n$ points in the plane, compute the expected area of a triangle *induced by* $P$, that is, a triangle whose vertices are selected uniformly at random from the points in $P$. This problem is a special case of computing the expected area of the convex hull of $k$ points, selected uniformly at random from $P$. These problems are important in computing the *functional diversity* in Ecology [4]. In this setting, each point represents some characteristics of a species, and the expected area of the convex hull provides an estimate of the diversity of the species, given that only $k$ species exist in a geographic region.

We present a simple exact algorithm for the problem that computes the expected triangle area in $O(n^2 \log n)$ time, and extends to the case of computing the area of the convex hull of a size $k$ subset. Additionally, we present a $(1 \pm \varepsilon)$-approximation algorithm for the case in which the ratio $\rho$ between the furthest pair distance and the closest pair distance of the points in $P$ is bounded. With high probability (whp.) our algorithm computes an answer in the range $[(1-\varepsilon)A^*, (1+\varepsilon)A]$, where $A$ is the true expected triangle area, in $O(\frac{1}{\varepsilon^{8/3}} \rho^4 n^{5/3} \log^{4/3} n)$ expected time.

**Notation.** Let $\Delta$ denote the random variable corresponding to a triangle induced by $P$, and let $\mathcal{A}(Q)$ denote the area of a region $Q \subset \mathbb{R}^2$. We are thus interested in computing $\mathbb{E}[\mathcal{A}(\Delta)]$. We denote the probability of an event $X$ by $\mathbb{P}[X]$. Assume w.l.o.g. that the origin $o$ lies outside of the convex hull $\mathcal{CH}(P)$ of $P$, and assume that $P \cup \{o\}$ is in general position, i.e. no three points lie on a line.

## 2 An Exact Algorithm

For a simple polygon $Q = v_0, .., v_n$ whose vertices are given in counterclockwise (ccw) order the well-known shoelace formula gives us that $\mathcal{A}(Q) = \frac{1}{2} \sum_{i=0}^{n} \mathcal{A}'(\overrightarrow{v_i v_{i+1 \bmod n}})$, where $\mathcal{A}'(\overrightarrow{pq}) = \det\left(\begin{smallmatrix} p_x & q_x \\ p_y & q_y \end{smallmatrix}\right)$ denotes the area of the triangle defined by the origin and the directed line segment from $p$ to $q$. See Fig. 1 for an illustration.

Let $E_1$, $E_2$, and $E_3$ be random indicator variables corresponding to the edges of $\Delta$ in ccw order. We then have $\mathbb{E}[\mathcal{A}(\Delta)] =$

$$\mathbb{E}\left[\sum_{i=1}^{3} \mathcal{A}'(E_i)\right] = \sum_{i=1}^{3} \mathbb{E}[\mathcal{A}'(E_i)] = \sum_{i=1}^{3} \sum_{a} a\mathbb{P}[\mathcal{A}'(E_i) = a].$$

*Département d'Informatique, Université Libre de Bruxelles, fisikop@gmail.com

§MADALGO, Aarhus University, [f.staals|constant]@cs.au.dk

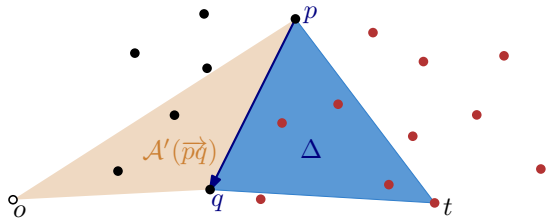



Figure 1: $\mathcal{A}(\Delta)$ is the sum of three "signed" areas, one of which is shown in orange. The number of red points is $n_{pq}$.

We now observe that all areas $a$ are realized by an ordered pair of points $(p, q)$, and thus $\sum_{i=1}^{m} \sum_{a} a\mathbb{P}[\mathcal{A}'(E_i) = a] =$

$$\sum_{i=1}^{3} \sum_{p,q \in P} \mathcal{A}'(\overrightarrow{pq}) \mathbb{P}[E_i = \overrightarrow{pq}] = \sum_{p,q \in P} \mathcal{A}'(\overrightarrow{pq}) \sum_{i=1}^{3} \mathbb{P}[E_i = \overrightarrow{pq}].$$

An edge $\overrightarrow{pq}$ cannot be both the $i^{\text{th}}$ and the $j^{\text{th}}$ edge of $\Delta$ (for $i \neq j$), and thus, $\sum_{i=1}^{3} \mathbb{P}[E_i = \overrightarrow{pq}]$ equals the probability that $\overrightarrow{pq}$ is a ccw edge in $\Delta$. For $\overrightarrow{pq}$ to be a ccw edge in $\Delta$, the remaining vertex $t$ of $\Delta$ should lie to the left of (the oriented line containing) $\overrightarrow{pq}$, and thus $\mathbb{P}[\overrightarrow{pq}$ is a ccw edge in $\Delta] = n_{pq} / \binom{n}{3}$, where $n_{pq}$ is the number of points to (the oriented line containing) $\overrightarrow{pq}$. This is illustrated in Fig. 1. Thus, we have

$$\begin{aligned} \mathbb{E}[\mathcal{A}(\Delta)] &= \sum_{p,q \in P} \mathcal{A}'(\overrightarrow{pq}) \mathbb{P}[\overrightarrow{pq} \text{ is a ccw edge in } \Delta] \\ &= \frac{1}{\binom{n}{3}} \sum_{p,q \in P} \mathcal{A}'(\overrightarrow{pq}) n_{pq}. \end{aligned} \tag{1}$$

As $\mathcal{A}'(\overrightarrow{pq})$ can be computed in $O(1)$ time for every pair $p, q$, all we need to do is compute all values $n_{pq}$. We can easily do this in $O(n^2 \log n)$ time, by fixing each point $p$ and sorting the remaining points around $p$. We conclude:

**Theorem 1** *We can compute* $\mathbb{E}[A(\Delta)]$ *in* $O(n^2 \log n)$ *time.*

This approach directly extends to computing $\mathbb{E}[\mathcal{A}(\mathcal{CH}(S))]$ of a randomly selected subset $S \subseteq P$ of size $k$.

## 3 A $(1 \pm \varepsilon)$-Approximation

We describe a $(1 \pm \varepsilon)$-approximation algorithm for evaluating Eq. 1, and thus for computing $\mathbb{E}[\mathcal{A}(\Delta)]$. The basic idea is to decompose the $\binom{n}{2}$ pairs of points into few pairs of sets $\{a\}, B$, such that all points $b \in B$ have roughly the same triangle area $\mathcal{A}'(\overrightarrow{ab})$, and to approximate the sum of the $n_{ab}$ values for $b \in B$.

### 3.1 Approximating the Areas

A well-separated pair decomposition (WSPD), with separation $s = 4/\delta$, of $P$ is a partition of the $\binom{n}{2}$ pairs of points into $m = O(s^2 n \log n)$ pairs of *well-separated* sets $(\{a_i\}, B_i)$, i.e. if $B_i$ fits into a disk $\mathcal{D}(B_i)$ of radius $r$, the distance between $a_i$ and any point $b \in B_i$ is at least $(s+1)r$ [2]. It follows that for any two points $p, q \in \mathcal{D}(B_i)$, the distance $\|a_i p\|$ is a $(1 \pm \delta)$-approximation of the distance $\|a_i q\|$.

Assume w.l.o.g. that the distance from the origin $o$ to any point in $P$ is at least the diameter $d$ of $P$. It then follows that for any set $B_i$, the pair $(\{o\}, B_i)$ is well-separated.

**Lemma 2** *Let $(\{a\}, B)$ be a well-separated pair with separation $s = 4/\delta$, where $\delta = \frac{\varepsilon c^2}{40 d^2}$, $c$ is the distance between the closest pair of points in $P$, and $d$ is the distance between the furthest pair of points in $P$, and finally let $A_{aB} = \mathcal{A}'(ap)$ for the point $p \in \mathcal{D}(B)$ furthest from the line containing $\overline{ao}$. For every $b \in B$, $A_{aB}$ is a $(1+\varepsilon)$-approximation of $\mathcal{A}'(\overrightarrow{ab})$.*

**Lemma 3** *We can compute an oracle that gives an $(1+\varepsilon)$-approximation of $\mathcal{A}'(pq)$, for any $p, q \in P$, in $O(1)$ time, using $O((\rho^4/\varepsilon^2) n \log n)$ preprocessing time, where $\rho$ is the ratio between the furthest and the closest pair of points in $P$.*

### 3.2 Approximating the Number of Points

Fix a point $a \in P$ and a subset $B \subseteq P$ of size $z \geq 2$. We present a $(1 \pm \varepsilon)$-approximation for $F^*_a(B) = F^*(B) = \sum_{b \in B} n_b$, where $n_b = n_{ab}$. Our algorithm will compute $F(B) = (z/|B'|) \sum_{b \in B'} n'_b$, where $B' \subseteq B$ is a sample of the points in $B$, and $n'_b$ is a $(1 \pm \delta)$-approximation of $n_b$. Let $E = |F^*(B) - F(B)|$ denote the error in our approximation.

Given a line(segment) $s$, we denote the half planes bounded by the line containing $s$ by $s^-$ and $s^+$. Let $\mathcal{H} = \{\overline{ab}^+ \mid b \in B\}$ denote the set of half planes defined by $a$ and $B$. For a given point $p \in P$, let $R_p = \{h \mid h \in \mathcal{H} \wedge h \ni p\}$ denote half planes containing $p$, and let $m_p$ denote the number of such half planes. We are thus interested in computing $F^*(\mathcal{H}) = \sum_{h \in \mathcal{H}} n_h = \sum_{p \in P} m_p = G^*_{\mathcal{H}}(P)$. To this end, our algorithm distinguishes two cases, depending on $z$.

**Case $\mathcal{H}$ is small.** When $z \leq t$, for some, to be determined $t$, we simply query each plane. Using a $(1 \pm \varepsilon)$-approximate half-plane counting algorithm [1] we immediately get $E \leq \sum_{h \in H} \varepsilon n_p = \varepsilon F^*(\mathcal{H})$.

**Case $\mathcal{H}$ is large.** When $z > t$ we take a (uniformly drawn) random sample $H$ of the half-planes, and query only the half-planes in $H$. More precisely, we compute $F(\mathcal{H}) = \overline{F}(H) = (z/|H|) \sum_{h \in H} n'_h$, where $n'_h$ denotes a $(1 \pm \delta)$-approximation of the number of points from $P$ on half-plane $h$. If we take a sample of size $O(r^2 \log r)$, then whp. $H$ is an $(1/r)$-approximation for the range space $(\mathcal{H}, \mathcal{R})$, where $\mathcal{R} = \{R_p \mid p \in P\}$ [3]. That is, for all ranges $R \in \mathcal{R}$ we have that

$$\left| \frac{|R|}{|\mathcal{H}|} - \frac{|R \cap H|}{|H|} \right| \leq (1/r).$$

This allows us to show that the absolute error E is at most $nz/r + nz\delta$. We now choose $(1/r) = \delta = (z\varepsilon)/8n$, which gives us $E \leq \varepsilon z^2/4$. By ordering the points defining the planes in $\mathcal{H}$ appropriately, we get $F^*(\mathcal{H}) \geq z(z-1)/2 \geq z^2/4$. Thus, $F(\mathcal{H})$ is a $(1 \pm \varepsilon)$-approximation for $F^*$.

**Running time.** We choose the threshold $t$ to minimize the running time. If $\mathcal{H}$ is small the running time to handle the pair $(\{a\}, B)$ is $O(z \log n)$. If $\mathcal{H}$ is large the running time is $O(r^2 \log r \log n) = O(\frac{n^2}{z^2 \varepsilon^2} \log^2 n)$. These two quantities balance out for $t = z = (n/\varepsilon)^{2/3} \log^{1/3} n$. We conclude:

**Lemma 4** *After $O(n \log n)$ expected time preprocessing, we can whp. compute a $(1 \pm \varepsilon)$ approximation of $F^*_a(B)$, for any $\{a\} \cup B \subseteq P$, in $O((n/\varepsilon)^{2/3} \log^{1/3} n)$ expected time.*

### 3.3 Combining the Approximations

Straightforward calculations show that if we combine the results from Lemmas 3 and 4, choosing both approximation errors to be $\varepsilon/3$, we get a $(1 \pm \varepsilon)$-approximation.

**Theorem 5** *Whp. we can compute a $(1 \pm \varepsilon)$-approximation of $\mathbb{E}[\mathcal{A}(\Delta)]$ in $O(\frac{1}{\varepsilon^{8/3}} \rho^4 n^{5/3} \log^{4/3} n)$ expected time.*

## 4 Future Work

We would like to improve, or remove, the dependency on $\rho$ in our approximation algorithm. A possible approach to do so would be to replace the WSPD by a different decomposition of the pairs of points that allows a better approximation of the triangle areas. We conjecture that computing $\mathbb{E}[\mathcal{A}(\Delta)]$ exactly is 3SUM-hard. Proving this is another avenue for future work. Finally, we would like to investigate a $(1 \pm \varepsilon)$-approximation algorithm for the general problem of computing $\mathbb{E}[\mathcal{A}(\mathcal{CH}(S))]$ for a fixed size sample $S$.

## References


[1] P. Afshani and T. M. Chan. On Approximate Range Counting and Depth. *Discrete & Comp. Geom.*, 42(1):3–21, 2009.

[2] P. B. Callahan and S. R. Kosaraju. A decomposition of multidimensional point sets with applications to k-nearest-neighbors and n-body potential fields. *J. ACM*, 42(1):67–90, Jan. 1995.

[3] D. Haussler and E. Welzl. $\varepsilon$-nets and simplex range queries. *Discrete & Comp. Geom.*, 2(2):127–151, 1987.

[4] M. A. Mouchet, S. Villéger, N. W. Mason, and D. Mouillot. Functional diversity measures: an overview of their redundancy and their ability to discriminate community assembly rules. *Functional Ecology*, 24(4):867–876, 2010.

# Transforming Hierarchical Trees on Metric Spaces*

Mahmoodreza Jahanseir† Donald R. Sheehy‡

**Abstract**

We show how a simple metric hierarchical tree called a cover tree transforms into a more complex one called a net-tree in linear time. We also propose two linear time algorithms to make a trade-off between depth and the degree of nodes in cover trees.

## 1 Introduction

Cover trees are a popular data structure for (approximate) nearest neighbor search on metric spaces of low intrinsic dimension [1]. They are superficially similar to the net-trees of Har-Peled & Mendel [2] as both structures may be interpreted as generalizations of compressed quadtrees beyond the Euclidean setting. Cover trees are the simplest of these data structures, requiring linear space, independent of the ambient or intrinsic dimension of the data set. An efficient implementation of cover trees is available. On the other hand, net-trees have better theoretical guarantees for preprocessing and query times. However, to achieve subquadratic preprocessing time, they use complex techniques and large constants which are not efficient in practice.

We show how a slight modification to the definition of a cover tree allows it to satisfy the stronger conditions of a net-tree. Leveraging this structural result, we give a simple algorithm to turn a given cover tree into a net-tree in linear time. We also propose two linear time algorithm to make a cover tree coarser or finer, which establish a trade-off between the height of the tree and the degrees of nodes.

## 2 Definitions

The input is a set of $n$ points $P$ in a metric space. The *closed metric ball* centered at $p$ with radius $r$ is denoted $\mathrm{B}(p,r) := \{q \in P \mid \mathbf{d}(p,q) \leq r\}$. The *doubling constant* $\rho$ of $P$ is the minimum $\rho \in \mathbb{R}$ such that every ball $\mathrm{B}(p,r)$ can be covered by $\rho$ balls of radius $r/2$. We assume $\rho$ is constant.

Cover trees and net-trees are both examples of hierarchical trees. Fig 1 shows an example of hierarchical trees. Note that in this figure the tree is not either a cover tree or a net-tree, because there are not any restrictions on the distance between points. In these trees, the input points are leaves and each point $p$ can be associated with many internal nodes. Each node is uniquely identified by its associated point and an integer called its *level*. Leaves are in level $-\infty$ and the root is in $+\infty$. The node in level $\ell$ associated with a point $p$ is denoted $p^\ell$. Let $\mathrm{par}(p^\ell)$ be the parent of a node $p^\ell \in T$. Also, let $\mathrm{ch}(p^\ell)$ be all children of node $p^\ell$. Each non-leaf node has a child with the same associated point. Similar to compressed quadtrees, a node skips a level iff it is the only child of its parent and it has only one child. Let $L_\ell$ be the points associated with nodes in level at least $\ell$. Let $P_{p^\ell}$ denote leaves of the subtree rooted at $p^\ell$. The levels of the tree represent the metric space at different scales. The constant $\tau > 1$, called the *scale factor* of the tree determines the change in scale between levels.

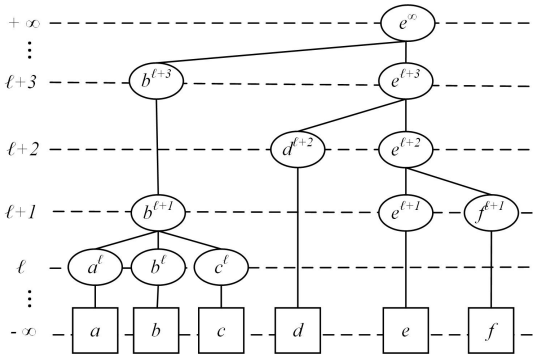



Figure 1: A hierarchical tree on $P = \{a, b, c, d, e, f\}$. Squares and ovals illustrate points and nodes.

**Cover Trees.** A cover tree is defined by the following properties. (*Packing*) For all distinct $p, q \in L_\ell$, $\mathbf{d}(p,q) > c_p\tau^\ell$. (*Covering*) For each $r^h \in \mathrm{ch}(p^\ell)$, $\mathbf{d}(p,r) \leq c_c\tau^\ell$. We call $c_p$ and $c_c$ the *packing constant* and the *covering constant*, respectively, and $c_c \geq c_p > 0$. We represent all cover trees with the same scale factor, packing constant, and covering constant with $\mathrm{CT}(\tau, c_p, c_c)$. Note that the cover tree definition by Beygelzimer et al. [1] create a tree in $\mathrm{CT}(2,1,1)$.

**Net-trees.** A *net-tree* is a hierarchical tree. For each node $p^\ell$ in a net-tree, the following invariants hold. (*Packing*) $\mathrm{B}(p, c_p\tau^\ell) \bigcap P \subset P_{p^\ell}$. (*Covering*) $P_{p^\ell} \subset \mathrm{B}(p, c_c\tau^\ell)$. Here, $c_p$ and $c_c$ are defined similar to cover trees. Also, we identify a set of net-trees with $\mathrm{NT}(\tau, c_p, c_c)$. The net-tree definition in [2] constructs a

*Partially supported by the National Science Foundation under grant numbers CCF-1464379 and CCF-1525978

†University of Connecticut reza@engr.uconn.edu

‡University of Connecticut don.r.sheehy@gmail.com

tree in NT(11, $\frac{\tau-5}{2(\tau-1)}$, $\frac{2\tau}{\tau-1}$). It is not hard to show that a net-tree satisfies the cover tree properties.

A net-tree can be augmented to maintain a list of nearby nodes with no additional cost. For each node $p^\ell$, $\text{Rel}(p^\ell)$ is the set of all nodes $x^h \in T$ with $y^g = \text{par}(x^h)$, such that $h \le \ell < g$ and $\mathbf{d}(p,x) \le c_r\tau^\ell$. We call $c_r$ the *relative constant*, and Har-Peled & Mendel set $c_r = 13$.

We add a new, easy to implement condition on cover trees. We require that children of a node $p^\ell$ are closer to $p$ than to any other point in $L_\ell$.

## 3 From cover trees to net-trees

It is not hard to prove the following lemmas.

**Lemma 1** *For each node $p^\ell$ in $T \in \text{CT}(\tau, c_p, c_c)$, $|\text{ch}(p^\ell)| = O(\rho^{\lg c_c\tau/c_p})$ and $|\text{Rel}(p^\ell)| = O(\rho^{\lg c_r/c_p})$.*

**Lemma 2** *For each node $x^f$ that is a descendant of $p^\ell$ in $T \in \text{CT}(\tau, c_p, c_c)$, $\mathbf{d}(p,x) < \frac{c_c\tau}{\tau-1}\tau^\ell$*

**Theorem 3** *For all $\tau > \frac{2c_c}{c_p} + 1$, if $T \in \text{CT}(\tau, c_p, c_c)$, then $T \in \text{NT}(\tau, \frac{c_p(\tau-1)-2c_c}{2(\tau-1)}, \frac{c_c\tau}{\tau-1})$.*

**Proof.** From Lemma 2, for a node $p^\ell \in T$, $P_{p^\ell} \subset \text{B}(p, \frac{c_c\tau}{\tau-1}\tau^\ell)$. We prove the packing property by contradiction. Suppose for contradiction there exists a point $r \in \text{B}(p, \frac{c_p(\tau-1)-2c_c}{2(\tau-1)}\tau^\ell)$ such that $r \notin P_{p^\ell}$. Then, there exists a node $x^f \in T$ which is the lowest node with $f \ge \ell$ and $r \in P_{x^f}$. Let $y^g$ be the child of $x^f$ such that $r \in P_{y^g}$. It is clear that $g < \ell$. By the parent property, $\mathbf{d}(y,p) > \mathbf{d}(y,x)$. So, $\mathbf{d}(y,p) \ge \mathbf{d}(p,x) - \mathbf{d}(x,y) > c_p\tau^\ell - \mathbf{d}(y,p) > c_p\tau^\ell/2$. Also, by the triangle inequality, $\mathbf{d}(y,r) \ge \mathbf{d}(y,p) - \mathbf{d}(p,r) > \frac{c_p}{2}\tau^\ell - \frac{c_p(\tau-1)-2c_c}{2(\tau-1)}\tau^\ell > \frac{c_c\tau^\ell}{\tau-1}$. On the other hand, $\mathbf{d}(y,r) \le \frac{c_c\tau}{\tau-1}\tau^g \le \frac{c_c\tau^\ell}{\tau-1}$, which results a contradiction. □

Note that by the definition, a cover tree does not maintain relative links. The following theorem shows that a cover tree can be augmented with relatives in linear time. This algorithm is similar to the find relative algorithm in [2]. However, it gives a smaller neighborhood space.

**Theorem 4** *For a cover tree $T \in \text{CT}(\tau, c_p, c_c)$, there is an algorithm to augment $T$ with relatives in $O(\rho^{\lg(\frac{c_c\tau}{c_p(\tau-1)})^2\tau} n)$ time with $c_r = \frac{c_c\tau^2}{(\tau-1)^2}$.*

For a cover tree, there is a trade-off between the height of the tree and the scale factor. In the following, we define two operations to change scale factor of a given tree. A *coarsening* algorithm modifies the tree to increase the scale factor. Similarly, a refining algorithm results a tree with smaller scale factor. Note that in Theorem 3, we assumed that $\tau > \frac{2c_c}{c_p} + 1$. However, in many cases we may have $\tau \le \frac{2c_c}{c_p} + 1$. For example, Beygelzimer et. al. [1] set $\tau = 2$, and they found $\tau = 1.3$ is even more efficient in practice. In these situations, we can use the coarsening operation to get the stronger packing and covering conditions of net-trees.

**Theorem 5** *Given a cover tree $T \in \text{CT}(\tau, c_p, c_c)$ and a constant integer $k > 1$. There exists a coarsening algorithm that turns $T$ into $T' \in \text{CT}(\tau^k, c_p, \frac{c_c\tau}{\tau-1})$ in $O(\rho^{\lg(\frac{c_c\tau}{c_p(\tau-1)})^2\tau} n \log k)$ time.*

**Proof.** This algorithm can be seen as combining every $k$ levels of $T$ into one level in $T'$. We define a mapping between nodes of $T$ and $T'$. In this mapping, each node $p^\ell$ in $T$ maps to a node $p^{\ell'} = p^{\lfloor \ell/k \rfloor}$ in $T'$. Here, we use prime as a function that indicates the level of the node in $T'$ that corresponds to $p^\ell$, i.e. $\ell' = \lfloor \ell/k \rfloor$

The algorithm starts with augmenting $T$ with relatives using Theorem 4 and $c_r = \frac{3c_c\tau^2}{(\tau-1)^2}$. Then, we process nodes of $T$ level by level from the lowest to the highest level. For a node $p^\ell$, let $q^m$ be the lowest ancestor of it such that $m' > \ell'$. If $p = q$, we assign $p^{\ell'}$ as a child of $p^{\ell'+1}$. Otherwise, we search the relatives of $q^m$ to find the right parent of $p^{\ell'}$. For each $x^f \in \text{Rel}(q^m)$, if $f' = \ell' + 1$, then we do a nearest neighbor search on the part of the subtree rooted at $x^f$ such that their corresponding levels in $T'$ are greater that $\ell'$. In addition, we store the closest node to $p^\ell$ in the search among relatives of $q^m$. Let $r^e$ be the closest node to $p^\ell$ in the previous search. Now, it only remains to assign $r^{\ell'}$ and $p^{\ell'}$ as children of $r^{\ell+1}$. Note that $m'$ can be greater than $\ell' + 1$. In this case, we find relatives of a dummy node $q^{k(\lfloor \ell/k \rfloor + 2)-1}$ and we do the same search on this list. Note that this computation needs to be done at most $O(n)$ times for coarsening of the entire tree, because the number of edges in $T$ is $O(n)$. Because of the page limitation we omit proof of correctness and time complexity for this algorithm. □

Refining operation is the reverse of coarsening. In this operation, each level of $T$ is split into at most $k$ levels. The refining algorithm is similar to coarsening, and we omit details due to lack of space.

**Theorem 6** *Given a cover tree $T \in \text{CT}(\tau, c_p, c_c)$ and a constant integer $k > 1$. There exists a refining algorithm that turns $T$ into $T' \in \text{CT}(\tau^{1/k}, c_p, c_c)$ in $O((\rho^{\lg(\frac{c_c\tau}{c_p(\tau-1)})^2\tau} + k)n)$ time.*

## References


[1] A. Beygelzimer, S. Kakade, and J. Langford. Cover trees for nearest neighbor. In *ICML*, 2006.

[2] S. Har-Peled and M. Mendel. Fast construction of nets in low dimensional metrics, and their applications. *SIAM Journal on Computing*, 35(5):1148–1184, 2006.

# Approximate Range Counting Revisited*

Saladi Rahul†

## 1 Standard geometric intersection query (Standard GIQ)

In a standard *geometric intersection query (GIQ)*, a set $S$ of $n$ geometric objects in $\mathbb{R}^d$ is preprocessed into an efficient data structure so that for any geometric query object, $q$, all the objects in $S$ intersected by $q$ can be reported (*reporting query*) or counted (*counting query*) quickly. In an *approximate counting query*, an approximate value of the number of objects in $S$ intersecting $q$ has to be reported; specifically, any value $\tau$ which lies in the range $[(1-\varepsilon)k, (1+\varepsilon)k]$, where $k = |S \cap q|$ and $\varepsilon \in (0,1)$. In an *emptiness query*, we want to decide if $|S \cap q| = 0$ or not. Notice that the approximate counting query is *at least* as hard as the emptiness query: When $k = 0$, we do not tolerate any error. Therefore, a natural goal while solving an approximate counting query is to *match* the space and the query time bounds of the corresponding emptiness query.

Approximate counting queries is the focus of this paper. Arguably, the three most popular types of *GIQ* problems are (i) *orthogonal range searching* ($S$ contains points, $q$ is an axes-parallel rectangle), (ii) *rectangle stabbing* ($S$ contains axes-parallel rectangles, $q$ is a point), and (iii) *halfspace range searching* ($S$ contains points, $q$ is a halfspace). While approximate counting queries have been well studied for (i) and (iii), there has been no concrete study of (ii). The full version presents a comprehensive summary of the previously known results.

**Our results for standard GIQ problems:** In this paper we study the halfspace range searching problem and the rectangle stabbing problem.

*Approximate Halfspace Range Counting in $\mathbb{R}^d, d \geq 4$:* We present a structure for halfspace range counting which is *sensitive* to the value of $k$. The data structure occupies $O(n)$ space and solves the query in $\tilde{O}\left((n/k)^{1-1/\lfloor d/2 \rfloor}\right)$ time. The answer is correct with high probability. When $k = \Theta(n)$, then the query time is $\tilde{O}(1)$, which is an attractive property to have. Previously, such sensitive data structures were known only in $d = 2, 3$ [1]. In $\mathbb{R}^d, d \geq 4$, existing structures occupy $\tilde{O}(n)$ space and solve the query in $\tilde{O}\left(n^{1-1/\lfloor d/2 \rfloor}\right)$ time.

*Approximate Rectangle Stabbing Counting in $\mathbb{R}^2$:* This paper initiates the concrete study of approximate rectangle stabbing counting. This specific problem is studied in the word-RAM model of computation. Consider the following two settings:

(1) $S$ contains 2-sided rectangles of the form $[x_1, \infty) \times [y_1, \infty)$. It is easy to see that this is a 2D dominance query. There is a *gap* between the 2D dominance counting query and the 2D dominance emptiness query. For 2D dominance counting query, Patrascu [6] gave a lower bound of $\Omega\left(\frac{\log n}{\log \log n}\right)$ query time for any data structure which uses $O(n \text{ polylog } n)$ space. On the other hand, for 2D dominance emptiness query, there is a linear space structure with query time $O(\log \log n)$.

(2) $S$ contains 3-sided rectangles of the form $[x_1, \infty) \times [y_1, y_2]$. This problem also has a gap between the counting query and the emptiness query. The bounds mentioned above for the counting query and the emptiness query hold true for this setting as well.

Our first result is a solution for approximate 2D dominance counting query and approximate 3-sided rectangle stabbing counting query whose bounds *match* their corresponding emptiness query: An $O(n/\varepsilon)$ size data structure for answering the query in $O(\log \log(n/\varepsilon k))$ time. Adapting existing techniques (for e.g., Afshani, Hamilton and Zeh [1]) leads to a solution for these problems with $\Theta((\log \log n)^2)$ query time, and with costlier dependency on $\varepsilon$ in the space and the query time.

We do not study the case where $S$ contains 4-sided rectangles of the form $[x_1, x_2] \times [y_1, y_2]$; because, this problem *does not* have a gap between the counting query and the emptiness query. For the emptiness query and the counting query, Patrascu [7] and Patrascu [6], respectively, gave a lower bound of $\Omega\left(\frac{\log n}{\log \log n}\right)$ query time for any data structure which uses $O(n \text{ polylog } n)$ space. JaJa, Mortensen and Shi [4] gave a linear space structure with matching query time for both the problems.

## 2 Colored-GIQ

Several practical applications have motivated the study of a more general class of *GIQ* problems, known as *colored-GIQ* problems [3, 5]. In this setting, the set $S$ of $n$ geometric objects in $\mathbb{R}^d$ come aggregated in dis-

*A full version of this paper has been posted on arXiv.org: http://arxiv.org/pdf/1512.01713v1.pdf

†Department of Computer Science and Engineering, University of Minnesota, *sala0198@umn.edu*

joint groups. Each group is assigned a unique color. Given a geometric query object, $q$, we are interested in reporting (*colored reporting query*) or counting (*colored counting query*) the colors which have at least one object intersected by $q$. Note that a standard $GIQ$ problem is a special case of its corresponding colored-$GIQ$ problem (assign each object in the standard $GIQ$ problem a unique color). The most popular and well studied colored-$GIQ$ problem is the *orthogonal colored range searching* problem: $S$ is a set of $n$ points in $\mathbb{R}^d$ and $q$ is an axes-parallel rectangle in $\mathbb{R}^d$. A motivating example for this problem would be the following database query: "How many countries have employees aged between 30 and 40 while earning more than 80,000 per year". Each employee can be represented as a point $(age, salary)$ and the query is represented as an axes-parallel orthogonal rectangle (unbounded in one direction) $[30, 40] \times [80{,}000, \infty)$. Each employee is assigned a color based on his nationality.

*A general technique for hard counting problems:* Unfortunately, for most colored counting queries the known space and query time bounds are very expensive. For example, for orthogonal colored range searching problem in $\mathbb{R}^d$, existing structures use $O(n^d)$ space to achieve polylogarithmic query time. Any substantial improvement in these bounds would require improving the best exponent of matrix multiplication [5]. Instead of an exact count, if one is willing to settle for an approximate count, then this paper presents a result with attractive bounds: an $O(n \text{ polylog } n)$ space data structure and an $O(\text{polylog } n)$ query time algorithm.

In an *approximate colored counting query*, an approximate value of the number of colors in $S$ intersecting $q$ has to be reported; specifically, any value $\tau$ which lies in the range $[(1-\varepsilon)k, (1+\varepsilon)k]$, where $k$ is the number of colors which have at least one object intersected by $q$.

*Is linear space and* $\log n$ *query time possible?* There are some instances of colored-$GIQ$ problems which are not "hard". For example, for orthogonal colored range searching, there are two settings for which exact counting can be done using $O(n \text{ polylog } n)$ space and in $O(\text{polylog } n)$ query time: (a) points lying in $\mathbb{R}^1$ and the query is an interval $[x_1, x_2]$, and (b) points lying in $\mathbb{R}^2$ and the query is a 3-sided rectangle of the form $[x_1, x_2] \times [y, \infty)$. So, a natural question is whether by allowing approximation a linear space data structure and an $O(\log n)$ query time solution can be obtained for these problems? In this paper we show that it is indeed possible. Specifically, we study the setting in (b) as it is more challenging.

## 3 Our techniques

This paper introduces some new ideas and also combines previous techniques in a non-trivial manner . The highlights are the following:

- A general technique for solving approximate counting of standard GIQ and colored-GIQ problems. Our technique can be viewed as an enhancement of Aronov and Har-Peled's approximate counting technique [2]. We introduce the idea of performing random sampling on *colors* (instead of input objects) to approximately count the colors intersecting the query object.
- The result for approximate rectangle stabbing counting is obtained by a non-trivial reduction to planar point location.
- The result for approximate orthogonal colored range counting is obtained by a non-trivial combination of two different types of random sampling techniques and a reduction to non-colored range searching problem.

## References


[1] Peyman Afshani, Chris H. Hamilton, and Norbert Zeh. A general approach for cache-oblivious range reporting and approximate range counting. *Computational Geometry: Theory and Applications*, 43(8):700–712, 2010.

[2] Boris Aronov and Sariel Har-Peled. On approximating the depth and related problems. *SIAM Journal of Computing*, 38(3):899–921, 2008.

[3] Prosenjit Gupta, Ravi Janardan, and Michiel H. M. Smid. Further results on generalized intersection searching problems: Counting, reporting, and dynamization. *Journal of Algorithms*, 19(2):282–317, 1995.

[4] Joseph JáJá, Christian Worm Mortensen, and Qingmin Shi. Space-efficient and fast algorithms for multidimensional dominance reporting and counting. In *Algorithms and Computation, 15th International Symposium, ISAAC 2004, Hong Kong, China, December 20-22, 2004, Proceedings*, pages 558–568, 2004.

[5] Haim Kaplan, Natan Rubin, Micha Sharir, and Elad Verbin. Counting colors in boxes. In *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 785–794, 2007.

[6] Mihai Patrascu. Lower bounds for 2-dimensional range counting. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing, San Diego, California, USA, June 11-13, 2007*, pages 40–46, 2007.

[7] Mihai Patrascu. Unifying the landscape of cell-probe lower bounds. *SIAM Journal of Computing*, 40(3):827–847, 2011.

# Almost All Even Yao-Yao Graphs Are Spanners (CG:YRF)

Wei Zhan* Jian Li†


## Abstract

In this abstract we show that, for any integer $k \geq 42$, the Yao-Yao graph $\mathsf{YY}_{2k}$ is a $t_k$-spanner, with stretch factor $t_k = 4.27 + O(k^{-1})$ when $k$ tends to infinity. Our result generalizes the best known result which asserts that all $\mathsf{YY}_{6k}$ are spanners for $k$ large enough [Bauer and Damian, SODA'13].


## 1 Introduction

A classic geometric spanner, Yao graph, was first introduced by Andrew Yao in his seminal work on high-dimensional Euclidean minimum spanning trees [3]. The construction of Yao graph $\mathsf{Y}_k$ is described in the following process.

Initially $\mathsf{Y}_k$ is an empty graph.
For each point $u$:
  For each $j = 0, \ldots, k-1$:
    Take the cone $C$ attached to $u$
      with angle in range $[2j\pi/k, 2(j+1)\pi/k)$;
    Select $v \in C \cap \mathcal{P}$ such that $|uv|$ is the shortest;
    Add edge $\overrightarrow{uv}$ into $\mathsf{Y}_k$.

The above process is usually referred to as a 'Yao step'.

One may notice that a Yao graph may not have a bounded degree, which was realized by Li et al. [2]. To address the issue, they proposed a modified construction with two Yao steps: the second step, which is called the 'reverse Yao step', eliminating a subset of edges of $\mathsf{Y}_k$ to ensure the maximum degree is bounded. It can be described by the following procedure:

Initially $\mathsf{YY}_k$ is an empty graph.
For each point $u$:
  For each $j = 0, \ldots, k-1$:
    Take the cone $C$ attached to $u$
      with angle in range $[2j\pi/k, 2(j+1)\pi/k)$;
    Select $v \in C \cap \mathcal{P}$, $\overrightarrow{vu} \in Y_k$
      such that $|uv|$ is the shortest;
    Add edge $\overrightarrow{vu}$ into $\mathsf{YY}_k$.

The resulting graph, $\mathsf{YY}_k$, is named as 'Yao-Yao graph'. The degrees in $\mathsf{YY}_k$ are upper-bounded by $2k$ and it has long been conjectured that $\mathsf{YY}_k$ are also spanners when $k$ is large [2, 1]:

**Conjecture 1** *There exists a $k_0$ such that for any integer $k > k_0$, $\mathsf{YY}_k$ is a geometric spanner.*

Our knowledge about the spanning properties of Yao-Yao graphs is still quite limited. Recently substantial progress was made by Bauer and Damian [1], who showed that $\mathsf{YY}_{6k}$ are spanners with stretch factor 11.76 for all integer $k \geq 6$. In this abstract, we present a step towards the resolution of Conjecture 1, by showing that almost all Yao-Yao graphs with even $k$ are geometric spanners for $k$ large enough. Formally, our results is as follows:

**Theorem 1** *For any $k \geq 42$, $\mathsf{YY}_{2k}$ is a $t_k$-spanner, where $t_k = 4.27 + O(k^{-1})$.*

## 2 Preliminaries

Our proof contains two major steps. We first introduce a class of intermediate graphs, called *trapezoidal Yao graphs* (denoted as $\mathsf{TY}_k$). We can show that $\mathsf{TY}_k$ is a geometric spanner; In step 2, we show that $\mathsf{YY}_{2k}$ spans $\mathsf{TY}_{2k}$ (i.e., the shortest $u$-$v$ path in $\mathsf{YY}_{2k}$ is at most a constant times the shortest $u$-$v$ path in $\mathsf{TY}_{2k}$). The definition of $\mathsf{TY}_k$ is similar to Yao graphs, but takes advantage of a shape called 'curved trapezoid'.

**Definition 1** *A curved trapezoid is an open shape, requiring the parameter $\theta \in [\pi/4, \pi/3)$:*

$$T_\theta = \{u(x, y) \mid 0 < x < 1, 0 < y < \sin\theta, |ou| < 1, |pu| < 1\}.$$

We regard $T_\theta$ as a shape attached to the origin $o$, and the closed arc not incident on $o$ is called the *critical arc*. See Figure 1 for an example.

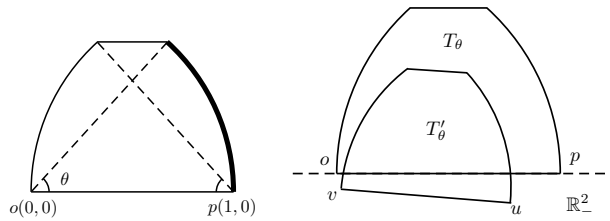



Figure 1: Left: The curved trapezoid. The critical arc is shown in bold. Right: With strict requirements on position, we have $T'_\theta - T_\theta \subset \mathbb{R}^2_-$.

*Tsinghua University
†Tsinghua University

We also introduce some geometric notations. $\mathcal{P}$ is the underlying set of points in $\mathbb{R}^2$. A cone between polar angles $\gamma_1 < \gamma_2$ with apex at the origin is denoted as $C(\gamma_1, \gamma_2)$. Some transformations are denoted by:

- (Dilation) $\lambda\mathcal{O} = \{\lambda z \mid z \in \mathcal{O}\}$.
- (Translation) $u + \mathcal{O} = \{u + z \mid z \in \mathcal{O}\}$.
- (Rotation) If $\mathcal{O}$ is rotated an angle $\gamma$ counterclockwise at the origin, the result is denoted by $\mathcal{O}^{\circlearrowleft\gamma}$.
- (Reflection) If $\mathcal{O}$ is reflected through the $x$-axis, the result is denoted by $\mathcal{O}^{-}$.

## 3 First Step

First we define another intermediate graph:

**Definition 2** *Let $\gamma = \lceil \frac{k}{4} \rceil \frac{2\pi}{k}$. For every $u \in \mathcal{P}$ and $j = 0, \ldots, k-1$, select the shortest $\overrightarrow{uv}$ with $v \in C_u(2j\pi/k, 2j\pi/k + \gamma)$. The chosen edges form the overlapping Yao graph $\mathsf{OY}_k(\mathcal{P})$.*

**Lemma 2** *If $k > 24$, then $\mathsf{OY}_k$ is a $\tau_k$-spanner where $\tau_k = \left(1 - 2\sin\left(\frac{\pi}{k} + \frac{\pi}{8}\right)\right)^{-1}$.*

**Definition 3** *Let $\theta = \lceil \frac{k}{8} \rceil \frac{2\pi}{k}$. For every $u \in \mathcal{P}$ and $j = 0, \ldots, k-1$, define two curved trapezoids*

$$\Gamma_{j1} = (T_\theta)^{\circlearrowleft 2j\pi/k} \quad \text{and} \quad \Gamma_{j2} = (T_\theta^-)^{\circlearrowleft 2j\pi/k}.$$

*Note that their bottom sides lie on the ray of angle $2j\pi/k$. For each $i = 1, 2$, select $\overrightarrow{uv}$ satisfying: there is a $\lambda > 0$ such that $u + \lambda\Gamma_{ji}$ has empty interior, with $v$ on its critical arc. All the selected edges form the graph $\mathsf{TY}_k(\mathcal{P})$.*

Since a sector in Definition 2 can be completely covered by two curved trapezoids, it actually holds:

**Lemma 3** *For any integer $k > 24$, $\mathsf{OY}_k$ is a subgraph of $\mathsf{TY}_k$ and thus $\mathsf{TY}_k$ is a $\tau_k$-spanner.*

## 4 Second Step

**Lemma 4** *Suppose $o \in \mathcal{P}$ and $T_\theta$ has an empty interior. If there is a point $a \in \mathcal{P}$ such that $0 < x_a < 1$, $y_a \le 0$ and $0 < \varphi(ap) < \pi/6$, then*

$$d_{\mathsf{TY}}(oa) \le x_a + (2\tau_{2k} + 1)|y_a|.$$

**Proof.** (Sketch) We present an iterative algorithm for finding a path from $a$ to $o$, which calls the shortest path in $\mathsf{TY}_{2k}$ as a subroutine. We can bound its length as desired, since intuitively the algorithm is trying to move along a direction as close to the negative direction of $x$-axis as possible.

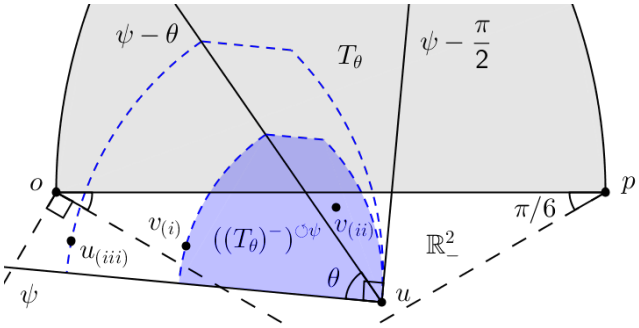



Figure 2: Illustration of the proof of Lemma 4.

Let $u$ be the current point in $\mathcal{P}$; Initially $u$ is set to be $a$.
While $u \neq o$ and $\varphi(ou) > -\frac{\pi}{6}$:
  Let $\psi = \min\{j\pi/k \mid j\pi/k > \varphi(uo), j = 0, 1, \ldots, 2k-1\}$;
  If $C_u(\psi - \frac{\pi}{2}, \psi - \theta) \cap \mathbb{R}^2_- \cap \mathcal{P}$ is empty:
    (i) $\mathsf{TY}_{2k}$ must contain an edge $\overrightarrow{uv}$ for $(T_\theta^-)^{\circlearrowleft\psi}$.
    We add $\overrightarrow{uv}$ to our path;
  Otherwise:
    Let $v$ be an arbitrary point in
      $C_u(\psi - \frac{\pi}{2}, \psi - \theta) \cap \mathbb{R}^2_- \cap \mathcal{P}$;
    (ii) Add the shortest path from $u$ to $v$ in $\mathsf{TY}_{2k}$;
  Set $u$ to be $v$ and proceed to the next iteration.
If now $u = o$, the path is already found. Otherwise
(iii) take the shortest path from $u$ to $o$ in $\mathsf{TY}_{2k}$ at last.

□

With Lemma 4 we can prove:

**Lemma 5** *On a Yao-Yao graph $\mathsf{YY}_{2k}(\mathcal{P})$ with $k \ge 42$, if $\overrightarrow{uv} \in \mathsf{TY}_{2k}$, then $d_{\mathsf{YY}}(uv) \le \tau'_k|uv|$, where $\tau'_k = 1 + O(k^{-1})$ is a constant depending only on $k$.*

Now combining Lemma 3 and 5, with the observation that $\tau_k = (1 - 2\sin(\pi/8))^{-1} + O(k^{-1}) = 4.27 + O(k^{-1})$, we can get our main result Theorem 1.

## References

[1] M. Bauer and M. Damian. An infinite class of sparse-Yao spanners. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 184–196. SIAM, 2013.

[2] X.-Y. Li, P.-J. Wan, Y. Wang, and O. Frieder. Sparse power efficient topology for wireless networks. In *System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on*, pages 3839–3848. IEEE, 2002.

[3] A. C.-C. Yao. On constructing minimum spanning trees in k-dimensional spaces and related problems. *SIAM Journal on Computing*, 11(4):721–736, 1982.

# Computing the planar slope number

Udo Hoffmann
Université libre de Bruxelles (ULB)
uhoffman@math.tu-berlin.de

**Abstract**

The *planar slope number* of a planar graph $G$ is defined as the minimum number of slopes that is required for a crossing-free straight line drawing of $G$. We show that determining the planar slope number is hard in the *existential theory of the reals*. We point out consequences for drawings that minimize the planar slope number.

## 1 Introduction

The *slope number* of a non-degenerate straight line drawing $D$ of a graph $G$ is defined to be the number of distinct slopes that is used to draw the edges of $G$ in $D$. The minimum slope number over all straight line drawings of $G$ is the *slope number* of $G$. Similarly, the *planar slope number* of a planar graph $G$ is the minimum slope number over all planar straight line drawing of $G$.

In this paper, we consider the computational complexity of computing the planar slope number. In Section 2, we show that determining the planar slope number of a graph is hard in the *existential theory of the reals*. Afterwards, in Section 3, we point out consequences for drawings that minimize the slope number: There are planar graphs, such that each drawing that minimizes the planar slope number requires irrational coordinates for the vertices and slopes of the edges. Furthermore, it is complete in the *existential theory of the rationals* ($\exists\mathbb{Q}$) (and thus possibly undecidable) to decide whether a planar graph has a drawing on the grid that minimizes the planar slope number.

### 1.1 The slope number

The slope number of a graph has mainly been studied for the relation of the maximum degree to the slope number: A simple lower bound for the slope number of a graph $G$ is $\lceil \Delta(G)/2 \rceil$, where $\Delta(G)$ denotes the maximum degree of $G$, since at most two edges of the same slope are incident to one vertex. The main work in this area deals with the question whether the slope number of a graph is also bounded from above by a function in the maximum degree. This was answered negatively [1, 10, 3] by examples of families of graphs of maximum degree 5 with arbitrarily large slope number. In contrast, Keszegh, Pach, and Pálvölgyi have shown that the planar slope number is bounded by an exponential function in the maximum degree [7].

From the computational point of view, it is known to be NP-complete to decide whether a graph has slope number 2 [4], and it is NP-complete to decide whether a planar graph has planar slope number 2 [5].

### 1.2 The existential theory of the reals

The *existential theory of the reals* ($\exists\mathbb{R}$) is a complexity class defined by the following complete problem: Given a quantifier-free formula $F(x_1, \ldots, x_n)$ that consists of logic connections of polynomial equalities and inequalities in the variables $x_1, \ldots, x_n$ with integer coefficients, is there an assignment of real values to the variables, such that the formula is satisfied? This problem can be reduced to deciding the solvability of a polynomial inequality system over the reals. Starting with Mnëv's universality theorem [9] many geometric problems have been shown to be hard in $\exists\mathbb{R}$.

We focus on the problem of *pseudoline stretchability*: Given a collection of curves in the plane, is there a homeomorphism of the plane, that maps the curves onto lines? For a good overview on the $\exists\mathbb{R}$-reduction for the stretchability problem we refer to [8]. The class $\exists\mathbb{R}$ is contained in PSPACE [2], and it is known to be NP-hard. Solvability of an $\exists\mathbb{R}$ instance with rational numbers ($\exists\mathbb{Q}$) is not known to be decidable (cf. [11]).

## 2 The reduction

We show that computing the planar slope number of a graph is hard in $\exists\mathbb{R}$.

**Theorem 1** *Deciding if the planar slope number of a planar graph with even maximum degree $\Delta$ is $\Delta/2$ is complete in $\exists\mathbb{R}$.*

We reduce pseudoline stretchability to the problem of computing the planar slope number. The general idea of the proof is indicated in Figure 1: We consider the arrangement graph $G_L$, that is obtained by placing a vertex (blue) on each intersection point of a pseudoline

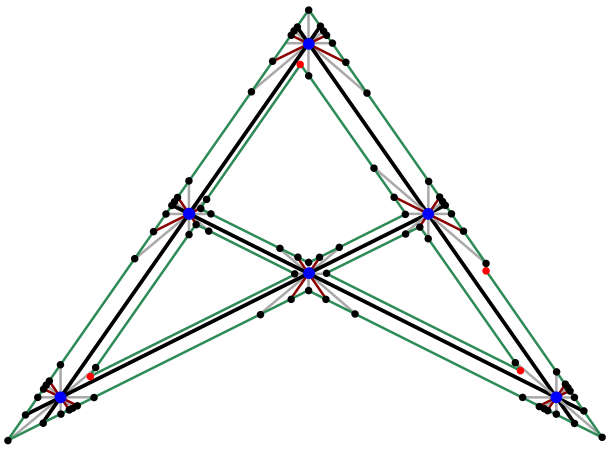


Figure 1: Constructing the planar graph from a (pseudo)line arrangement (black).

arrangement $L$ of $n$ pseudolines and consider the pseudosegments between two points as edges (black). We modify $G_L$ to a graph $G'_L$, such that each vertex of $G_L$ has degree $4n$ in $G'_L$. The graph $G'_L$ will be 3-connected (up to some subdivisions of edges). Thus $G'_L$ has the same rotation system of edges in each planar embedding. We make sure that the edges originating from one pseudoline are opposite in this rotation system. By the following observation a drawing of $G'_L$ with slope number $2n$ guarantees the stretchability of the pseudoline arrangement.

**Observation 1** *Let $G$ be a planar graph with even maximum degree $\Delta$, and let $D$ be a planar straight line drawing of $G$ with slope number $\Delta/2$. The opposite edges of a vertex of degree $\Delta$ in $D$ have the same slope.*

We describe the construction of $G'_L$ from a line arrangement. (The same construction can be obtained from a pseudoline arrangement.) From a realization of $L$ we add edges to the arrangement graph, such that each vertex of $G_L$ has two neighbors for each slope used by the arrangement (brown edges). We add $2n$ neighbors to each vertex of $G_L$ by using $n$ intermediate slopes, one slope between each slope of the arrangement (gray edges); this step will ensure that the graph is 3-connected. In each face (including the unbounded one) we connect the vertices by a cycle (green). We subdivide one edge of each cycle (red vertex) to ensure that this cycle can be realized using a polygon that is parallel to the bounding edges of the face. Each drawing of $G'_L$ with $2n$ slopes contains a realization of $L$, which concludes the proof.

## 3 Consequences

The fact that there are non-simple line arrangements that have an irrational coordinates in each representation [6] directly translates into the following result.

**Corollary 2** *There are planar graphs, such that each planar drawing that minimizes the slope number has at least one vertex with an irrational coordinate.*

We want to point out, that deciding the existence of a drawing on the grid (equivalently by scaling: with rational coordinates) is complete in $\exists\mathbb{Q}$ (cf. [11]). If $\exists\mathbb{Q}$ is undecidable, this has an interesting consequence for upper bounds on the grid size.

**Theorem 3** *Assume $\exists\mathbb{Q}$ is undecidable. Then there exists no computable function $f$, such that every graph $G$, that has a slope minimizing drawing on the grid, can be drawn with this slope number on a grid of size $f(|V(G)|) \times f(|V(G)|)$.*

**Proof.** Assume there exists such a computable function $f$. Then checking all possible drawings on $f(|V(G)|) \times f(|V(G)|)$ leads to an algorithm that decides $\exists\mathbb{Q}$. □

## References


[1] J. Barát, J. Matoušek, and D. R. Wood. Bounded-degree graphs have arbitrarily large geometric thickness. *European Journal of Combinatorics*, 13:R3, 2006.

[2] J. Canny. Some algebraic and geometric computations in PSPACE. In *STOC*, pages 460–467. ACM, 1988.

[3] V. Dujmović, M. Suderman, and D. R. Wood. Graph drawings with few slopes. *Computational Geometry*, 38:181–193, 2007.

[4] M. Formann, T. Hagerup, J. Haralambides, M. Kaufmann, F. T. Leighton, A. Symvonis, E. Welzl, and G. J. Woeginger. Drawing graphs in the plane with high resolution. *SIAM Journal on Computing*, 22:1035–1052, 1993.

[5] A. Garg and R. Tamassia. On the computational complexity of upward and rectilinear planarity testing. *SIAM Journal on Computing*, 31:601–625, 2001.

[6] B. Grünbaum. *Convex Polytopes*, volume 221 of *Graduate Texts in Mathematics*. Springer-Verlag, 2003.

[7] B. Keszegh, J. Pach, and D. Pálvölgyi. Drawing planar graphs of bounded degree with few slopes. *SIAM Journal on Discrete Mathematics*, 27:1171–1183, 2013.

[8] J. Matoušek. Intersection graphs of segments and $\exists\mathbb{R}$. *arXiv:1406.2636*, 2014.

[9] N. E. Mnëv. The universality theorems on the classification problem of configuration varieties and convex polytopes varieties. In *Topology and Geometry – Rohlin Seminar*, LNM, pages 527–543. Springer, 1988.

[10] J. Pach and D. Pálvölgyi. Bounded-degree graphs can have arbitrarily large slope numbers. *European Journal of Combinatorics*, 13:N1, 2006.

[11] B. Sturmfels. On the decidability of diophantine problems in combinatorial geometry. *Bulletin of the American Mathematical Society*, 17:121–124, 1987.

# Network Optimization on Partitioned Pairs of Points

Esther M. Arkin † Aritra Banik *
Paz Carmi * Gui Citovsky † Su Jia †
Matthew J. Katz * Tyler Mayer †
Joseph S. B. Mitchell †

**Abstract**

Given a set $S = \bigcup_{i=1}^{n}\{p_i, q_i\}$ of $n$ pairs of points in a metric space, we study the problem of computing, what we call, a *feasible* partition $S = S_1 \cup S_2$ such that $p_i \in S_1$ if and only $q_i \in S_2 \; \forall \; i$. The partition should optimize the cost of a *pair* of networks, one built on $S_1$, and one built on $S_2$. In this work we consider the network structures to be matchings, minimum spanning trees (MSTs), traveling salesman tours (TSP tours), or their bottleneck equivalents. Let $f(X)$ be some network structure computed on point set $X$ and let $\lambda(f(X))$ be the bottleneck edge of that network. For each of the aforementioned network structures we consider the objective of (1) minimizing $|f(S_1)| + |f(S_2)|$, (2) minimizing $max\{|f(S_1)|, |f(S_2)|\}$, or (3) minimizing $max\{|\lambda(f(S_1))|, |\lambda(f(S_2))|\}$. Here, $|\cdot|$ denotes the sum of the edge lengths. We show several hardness results and an $O(1)$ approximation for every objective considered. Our results are summarized in Table 1 and full details can be found in [1].

## 1 Introduction

There is a large literature on generalized or "one-of-a-set" network optimization problems.

*Dept. of Computer Science, Ben-Gurion University, Israel. {aritrabanik,carmip}@gmail.com, matya@cs.bgu.ac.il

†Dept. of Applied Mathematics and Statistics, Stony Brook University. {esther.arkin, gui.citovsky, su.jia, tyler.mayer, joseph.mitchell}@stonybrook.edu

| Objective: | (1) | (2) | (3) |
|---|---|---|---|
| Matching | 2 | 3 | 3 |
| MST | $3\alpha$ | 9 | $4\alpha$ |
| TSP tour | $3\beta$ | 18 | $6\beta$ |

Table 1: $O(1)$ approximation results for 3 objective functions. $\alpha$ is the Steiner ratio, $\beta$ is the approximation factor for the TSP.

Here, one is given a point set $S$ and subsets $U_i \subseteq S$ and asked to select one, or at least one, point from each subset such that the total weight of an MST, TSP tour, or other single network structure computed over the chosen subset of points is minimized; see [3] for a survey.

As far as we know, the problem of partitioning points from pairs into two sets in order to optimize a cost function over networks computed on *both* sets has not been extensively studied. One recent work of Arkin et al. [2] does address the problem of partitioning points from pairs in the plane into two sets so as to minimize the sum or max of the radii of the minimum enclosing disks for sets $S_1$, and $S_2$.

## 2 Matchings

**Theorem 1** *Let $M(X)$ be the minimum weight matching on point set $X$. There exists a 2-approximation algorithm for the problem of computing a feasible partition $S = S_1 \cup S_2$ which minimizes $|M(S_1)| + |M(S_2)|$ in general metric spaces.*

**Proof.** Let $S = S_1^* \cup S_2^*$ be the optimal partition and $OPT = |M(S_1^*)| + |M(S_2^*)|$ be the cost of this solution. Let $M(S)$ be the minimum weight perfect matching on point set $S$ (excluding edges $(p_i, q_i) \forall i$). Note that $|M(S)| \leq OPT$.

Let $\hat{M}(s)$ be the minimum weight one-of-a-pair matching. That is, select exactly one point from each input pair $\{p_i, q_i\}$ such that the weight of the minimum weight matching over the chosen $n$ points is as small as possible, and call this matching $\hat{M}(S)$. As $OPT$ is the

sum of the weights of two such one-of-a-pair matchings over point set $S$, $|\hat{M}(S)| \leq \frac{OPT}{2}$.

To obtain a 2-approximation to this problem, compute $\hat{M}(S)$ and let $S_1$ be the set of input points which have an edge of this matching incident on them and let $S_2 = S \setminus S_1$. Return the partition $S = S_1 \cup S_2$.

Note $|M(S_1)| = |\hat{M}(S)| \leq \frac{OPT}{2}$. To bound $|M(S_2)|$ consider the multigraph $G = (V = S, E = M(S) \cup \hat{M}(S))$. All $v \in S_2$ have degree 1, from $M(S)$, and all $u \in S_1$ have degree 2 from $M(S) \cup \hat{M}(S)$. For each $v_i \in S_2$, either $v_i$ is matched to $v_j \in S_2$, or $v_i$ is matched to $u_i \in S_1$. In the former case one can consider $v_i, v_j$ matched and charge the weight of this edge to $|M(S)|$. In the latter case, note that each $u \in S_1$ is part of a unique cycle, or path. If $u \in S_1$ is part of a cycle then no vertex in that cycle belongs to $S_2$ by degree constraints. Thus, if $v_i \in S_2$ is matched to $u_i \in S_1$, $u_i$ is part of a unique path whose other terminal vertex $x$ belongs to $S_2$. One can consider $v_i, x$ matched and charge this weight to the unique path connecting them in $G$. Thus, $|M(S_2)|$ can be charged entirely to edges of $M(S) \cup \hat{M}(S)$ and has weight at most $|M(S)| + |\hat{M}(S)| \leq \frac{3}{2}OPT$.

Therefore, the approximate partition $S = S_1 \cup S_2$ permits one to build a matching on each set so that $|M(S_1)| + |M(S_2)| \leq 2OPT$. □

## 3 Spanning Trees

**Theorem 2** *Let $MST(X)$ be an $MST$ on point set $X$. Given $S$ and $k \in \mathbb{R}^1$, deciding if there exists a feasible partition $S = S_1 \cup S_2$ : $max\{|MST(S_1)|, |MST(S_2)|\} \leq k$ is weakly NP-Complete.*

**Proof.** The reduction is from PARTITION: given a set $S = \{x_1, x_2, ..., x_n\}$ of $n$ integers, decide if there is a partition $S = S_1 \cup S_2$ : $S_1 \cap S_2 = \emptyset$, with $\sum_{i \in S_1} x_i = \sum_{j \in S_2} x_j$. Let $M = \sum_i x_i$. Given any instance $S$ of PARTITION, create a geometric instance of our problem, as shown in Figure 1.

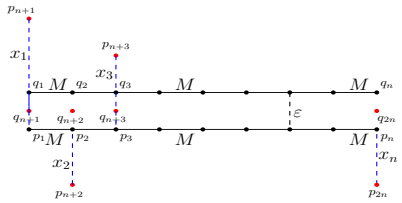



Figure 1: Transformation from PARTITION.

Create a "backbone" for each tree by placing $n$ pairs of points $\{p_i, q_i\}_{i=1}^n$ along two parallel $\epsilon$-separated lines spaced $M$ apart so that $p_i, q_i$ are vertically adjacent. For each integer $x_i$ in the instance of partition place a point $p_{n+i}$ at vertical distance $x_i$ from $p_i$, and its corresponding pair $q_{n+i}$ at distance $\epsilon/2$ from $q_i$ and $p_i$. One tree will incur an "extra cost" of $\epsilon/2$ to pick up $q_{n+i}$ and the other will incur an extra cost of $x_i$ to pick up $p_{n+i}$. Therefore, any algorithm that constructs a partition which minimizes the weight of the heavier tree also minimizes $max\{\sum_{i \in S_1} x_i, \sum_{j \in S_2} x_j\}$. Thus, for $\epsilon > 0$ small enough the instance of partition is solvable iff there exists a feasible partition such that both trees have weight $(n-1)M + M/2 + o(1)$. □

## References

[1] E. M. Arkin, A. Banik, P. Carmi, G. Citovsky, S. Jia, M. J. Katz, T. Mayer, and J. S. B. Mitchell. Network optimization on partitioned pairs of points. *Submitted*, 2016.

[2] E. M. Arkin, J. M. Díaz-Báñez, F. Hurtado, P. Kumar, J. S. B. Mitchell, B. Palop, P. Pérez-Lantero, M. Saumell, and R. I. Silveira. Bichromatic 2-center of pairs of points. *Comput. Geom.*, 48(2):94–107, 2015.

[3] G. Gutin and A. P. Punnen. *The traveling salesman problem and its variations*, volume 12. Springer Science & Business Media, 2006.

# A geometric approach to $k$-SUM

Jean Cardinal*, John Iacono†, Aurélien Ooms‡


## Abstract

It is known that $k$-SUM can be solved using $\tilde{O}(n^{\lceil \frac{k}{2} \rceil})$ time and linear queries (here the notation $\tilde{O}$ ignores polylogarithmic factors). On the other hand, there is a point location algorithm due to Meiser that shows the existence of $\tilde{O}(n^4)$-depth algebraic computation trees for $k$-SUM. By streamlining Meiser's algorithm, we prove $k$-SUM can be solved using $\tilde{O}(n^3)$ expected linear queries in $\tilde{O}(n^{\lceil \frac{k}{2} \rceil + 8})$ expected time. Thus, we show that it is possible to have an algorithm with a runtime almost identical (up to the $+8$) to the best known algorithm but for the first time also with the number of queries on the input a polynomial that is independent of $k$. The new algorithms we present rely heavily on fundamental tools from computational geometry: $\varepsilon$-nets and cuttings. A preprint is available at http://arxiv.org/abs/1512.06678.


## 1 Introduction

The $k$-SUM problem is defined as follows: given a collection of $n$ real numbers, decide whether any $k$ of them sum to zero, where $k$ is a constant. It is a fixed-parameter version of the subset-sum problem, a standard *NP*-complete problem. The $k$-SUM problem, and in particular the special case of 3SUM, has proved to be a cornerstone of the fine-grained complexity program aiming at the construction of a complexity theory for problems in $P$. In particular, there are deep connections between the complexity of $k$-SUM, the Strong Exponential Time Hypothesis [3, 10], and the complexity of many other major problems in $P$ [1, 4, 6, 7].

It has been long known that the $k$-SUM problem can be solved in time $O(n^{\frac{k}{2}} \log n)$ for even $k$, and $O(n^{\frac{k+1}{2}})$ for odd $k$. Erickson [5] proved a near-matching lower bound in the $k$-linear decision tree model. In this model, the complexity is measured by the depth of a decision tree, every node of which corresponds to a query of the form $q_{i_1} + q_{i_2} + \cdots + q_{i_k} \leq^? 0$, where $q_1, q_2, \ldots, q_n$ are the input numbers. In a recent breakthrough paper, Grønlund and Pettie [7] showed that in the $(2k-2)$-linear decision tree model, where queries test the sign of weighted sums of up to $2k-2$ input numbers, only $O(n^{\frac{k}{2}} \sqrt{\log n})$ queries are required for odd values of $k$. In particular, there exists a 4-linear decision tree for 3SUM of depth $\tilde{O}(n^{\frac{3}{2}})$, while every 3-linear decision tree has depth $\Omega(n^2)$ [5]. This indicates that increasing the size of the queries, defined as the maximum number of input numbers involved in a query, can yield significant improvements on the depth of the minimal-height decision tree. Ailon and Chazelle [2] slightly extended the range of query sizes for which a nontrivial lower bound could be established, elaborating on Erickson's technique.

It has been well-established that there exist nonuniform polynomial-time algorithms for the subset-sum problem. One of them was described by Meiser [9], and is derived from a data structure for point location in arrangements of hyperplanes using the bottom vertex decomposition.

## 2 Our results

Our first contribution is to show, through a careful implementation of Meiser's basic algorithm idea [9], the existence of an $n$-linear decision tree of depth $\tilde{O}(n^3)$ for $k$-SUM.

We use standard results on $\varepsilon$-nets. Using a theorem due to Haussler and Welzl [8], it is possible to construct an $\varepsilon$-net for the range space defined by hyperplanes and simplices using random sampling. If no hyperplane in this sample intersects a given simplex, then with high probability at most a fraction of the hyperplanes will intersect the simplex. We use this theorem to design a prune and search algorithm for the problem of locating a point in an arrangement of hyperplanes. This algorithm will perform linear queries about the coordinates of the point to locate. The number of such queries the algorithm will need to perform is polynomial in the dimension and logarithmic in the number of hyperplanes.

Although the high-level algorithm itself is not new, we refine the implementation and analysis for the $k$-SUM problem. Meiser presented his algorithm as a general method of point location in $m$ given $n$-dimensional hyperplanes that yielded a $\tilde{O}(n^4 \log m)$-depth algebraic

*Supported by the "Action de Recherche Concertée" (ARC) COPHYMA, convention number 4.110.H.000023. Université libre de Bruxelles (ULB), Brussels, Belgium, jcardin@ulb.ac.be.

†Research partially completed while on on sabbatical at the Algorithms Research Group of the Département d'Informatique at the Université Libre de Bruxelles with support from a Fulbright Research Fellowship, the Fonds de la Recherche Scientifique — FNRS, and NSF grants CNS-1229185, CCF-1319648, and CCF-1533564. New York University, New York, United States of America, icalp2016submission@johniacono.com.

‡Supported by the Fund for Research Training in Industry and Agriculture (FRIA). Université libre de Bruxelles (ULB), Brussels, Belgium, aureooms@ulb.ac.be.

computation tree; when viewing the $k$-SUM problem as a point location problem, $m$ is $O(n^k)$ and thus Meiser's algorithm can be viewed as giving a $\tilde{O}(n^4)$-depth algebraic computation tree.

Our first result shows that while the original algorithm was cast as a nonuniform polynomial-time algorithm, it can be implemented in the linear decision tree model with an improved upper bound. Moreover, this result implies the same improved upper bound on the depth of algebraic computation trees for the $k$-SUM problem.

**Theorem 1** *There exist linear decision trees and algebraic computation trees of depth at most $O(n^3 \log^3 n)$ solving the $k$-SUM problem.*

There are two subtleties to this result. The first is inherent to the chosen complexity model: even if the number of queries to the input is small (in particular, the degree of the polynomial complexity is invariant on $k$), the time required to *determine which queries should be performed* may be arbitrary. We present two Las Vegas algorithms that implement those decision trees efficiently in the RAM model. With a first naïve algorithm, we show the query determination time can be trivially bounded by $\tilde{O}(n^{k+2})$ on expectation. The second algorithm builds on the first one and reduces this time to $\tilde{O}(n^{\lceil \frac{k}{2} \rceil + 8})$.

**Theorem 2** *There exists an $\tilde{O}(n^{\lceil \frac{k}{2} \rceil + 8})$ Las Vegas algorithm in the RAM model expected to perform $O(n^3 \log^3 n)$ linear queries solving the $k$-SUM problem.*

The second issue we address is that the linear queries in the above algorithm may have size $n$, that is, they may use all the components of the input. The lower bound of Erickson shows that if the queries are of minimal size, the number of queries cannot be a polynomial independent of $k$ such as what we obtain, so non-minimal query size is clearly essential to a drastic reduction in the number of queries needed. This gives rise to the natural question concerning the relation between query size and number of queries. In particular, one question is whether queries of size less than $n$ would still allow the problem to be solved using a number of queries that is a polynomial independent of $k$. We show that this is possible; we introduce a range of algorithms exhibiting an explicit tradeoff between the number of queries and their size. Using a blocking scheme, we show that we can restrict to $o(n)$-linear decision trees.

**Theorem 3** *There exists an $o(n)$-linear decision tree of depth $\tilde{O}(n^3)$ solving the $k$-SUM problem. Moreover, this decision tree can be implemented as an $\tilde{O}(n^{\lceil \frac{k}{2} \rceil + 8})$ Las Vegas algorithm.*

We also give a range of tradeoffs for $O(n^{1-\alpha})$-linear decision trees.

**Theorem 4** *For any $\alpha$ such that $0 < \alpha < 1$, there exists an $O(n^{1-\alpha})$-linear decision tree of depth $\tilde{O}(n^{3+(k-4)\alpha})$ solving the $k$-SUM problem. Moreover, this decision tree can be implemented as an $\tilde{O}(n^{(1+\alpha)\frac{k}{2}+8.5})$ Las Vegas algorithm.*

Although the proposed algorithms still involve nonconstant-size queries, this is the first time such tradeoffs are explicitly tackled.

## References


[1] Amir Abboud, Kevin Lewi, and Ryan Williams. Losing weight by gaining edges. In *European Symposium on Algorithms (ESA 2014)*, pages 1–12. Springer, 2014.

[2] Nir Ailon and Bernard Chazelle. Lower bounds for linear degeneracy testing. *J. ACM*, 52(2):157–171, 2005.

[3] Marco Carmosino, Jiawei Gao, Russell Impagliazzo, Ivan Mikhailin, Ramamohan Paturi, and Stefan Schneider. Nondeterministic extensions of the strong exponential time hypothesis and consequences for non-reducibility. *Electronic Colloquium on Computational Complexity (ECCC 2015)*, 22:148, 2015.

[4] Timothy M. Chan and Moshe Lewenstein. Clustered integer 3SUM via additive combinatorics. In Rocco A. Servedio and Ronitt Rubinfeld, editors, *Symposium on Theory of Computing (STOC 2015)*, pages 31–40. ACM, 2015.

[5] Jeff Erickson. Lower bounds for linear satisfiability problems. *Chicago Journal of Theoretical Computer Science*, 1999.

[6] Anka Gajentaan and Mark H. Overmars. On a class of $O(n^2)$ problems in computational geometry. *Comput. Geom.*, 5:165–185, 1995.

[7] Allan Grønlund and Seth Pettie. Threesomes, degenerates, and love triangles. In *Foundations of Computer Science (FOCS 2014)*, pages 621–630. IEEE, 2014.

[8] David Haussler and Emo Welzl. $\varepsilon$-nets and simplex range queries. *Discrete & Computational Geometry*, 2(1):127–151, 1987.

[9] S. Meiser. Point location in arrangements of hyperplanes. *Information and Computation*, 106(2):286–303, 1993.

[10] Mihai Patrascu and Ryan Williams. On the possibility of faster SAT algorithms. In *Symposium on Discrete Algorithms (SODA 2010)*, pages 1065–1075, 2010.

# Classification of Normal Curves on a Tetrahedron

Clément Maria* Jonathan Spreer†


## Abstract

In this article, we give a combinatorial classification of all normal curves drawn on the boundary of a tetrahedron. We characterise normal curves in terms of intersection numbers with the edges of the tetrahedron.


## 1 Introduction

*Normal curves* are a restricted family of loops drawn on the boundary of a tetrahedron. They play a central role in low dimensional topology through the *Haken theory of normal curves and surfaces*, which is at the heart of most of the algorithmic breakthroughs in low dimensional topology over the past 25 years. This paper gives an exhaustive classification of all possible normal curves in terms of their intersection patterns with the edges of the tetrahedron. A full version of this abstract [3] applies this work in the context of *quantum topology*.

**Background:** Let $t$ be a tetrahedron and $\partial t$ its boundary. A *normal curve* $L$ on $\partial t$ is a *simple closed loop* that:

(i) is in general position w.r.t. $t$, i.e., does not pass through the vertices of $t$ and intersects the edges of $t$ transversally,

(ii) the intersection of $L$ with any triangular face of $t$ is a collection of disjoint arcs joining two *distinct* edges of the triangle.

We know [2] that, up to (normal) isotopy, normal curves are entirely defined by the number of times they intersect each edge of the tetrahedron. For a tetrahedron $t$ with edges $\{e_0, \dots, e_5\}$, and a normal curve $L$ on $\partial t$, we define $\phi(e_i) := |L \cap e_i|$, i.e., the number of times $L$ crosses $e_i$. The *intersection symbol of* $L$ is defined to be the $2 \times 3$ matrix of the values $\phi(e_i)$, where the first row contains intersection numbers of the edges of a triangle, and the intersection numbers of opposite edges appear in the same column; see Figure 1 left.

## 2 Classification of normal curves by their intersection symbols

We characterise normal curves on the boundary of a tetrahedron $t$ in terms of their intersections symbols. To do so, we translate the problem into planar topology and give an inductive argument based on intersection symbols.

**From tetrahedron to punctured disk:** Topologically, studying normal curves on a tetrahedron with vertices $\{a, b, c, d\}$ is equivalent to studying simple closed curves on the 3-*punctured disk* $\mathbb{D}_3$, i.e., the closed 2-dimensional disk from which three arbitrary but fixed points $a$, $b$ and $c$ in its interior have been removed (and $d$ is sent to $\partial\mathbb{D}$); see Figure 1 left.

We say that a loop in $\mathbb{D}_3$ is *reduced* if it does not cross a tetrahedron edge twice in a row. We define the *intersection symbol of a reduced loop* in $\mathbb{D}_3$ to be the $2 \times 3$ integer matrix of intersection numbers of the reduced loop with the tetrahedron edges embedded in $\mathbb{D}_3$. Note that reduced loops on the punctured disk are the equivalent of the normal curves on the boundary of a tetrahedron. Naturally, the intersection symbol of a reduced loop constitutes a valid tetrahedron intersection symbol. We refer to the full version of this work [3] and [1] for a more formal introduction to these concepts.

**Classification of loops in the punctured disk:** For a loop $L$ in $\mathbb{D}_3$, we denote its isotopy class by $[L]$; it is the class of all loops isotopic to $L$ in $\mathbb{D}_3$. We prove that the "intersection symbol" is well-defined for isotopy classes of loops.

**Lemma 1** *The following is true: (i) any isotopy class of loops in $\mathbb{D}_3$ admits a reduced loop, (ii) any two isotopic reduced loops have equal intersection symbols, (iii) any two non-isotopic reduced loops have distinct intersection symbols.*

The statements follow from an iterative reduction of the loop (i), a simple construction using the fundamental group of $\mathbb{D}_3$ (ii), and the construction of a "canonical loop" associated to an intersection symbol (iii). See the full version [3] for details.

It follows that we can refer to the intersection symbol of an isotopy class of loops $[L]$ as the intersection symbol of any reduced loop in $[L]$. By a small abuse of notation, we also refer to the intersection symbol of a loop $L$ as the intersection symbol of $[L]$.

By virtue of the Jordan-Schoenflies theorem [1]—that says that a loop separates the disk into two regions, the *inside* and the *outside*—we distinguish three types of loops: (i) loops containing no puncture in the inside;

*The University of Queensland, c.maria@uq.edu.au
†The University of Queensland, j.spreer@uq.edu.au

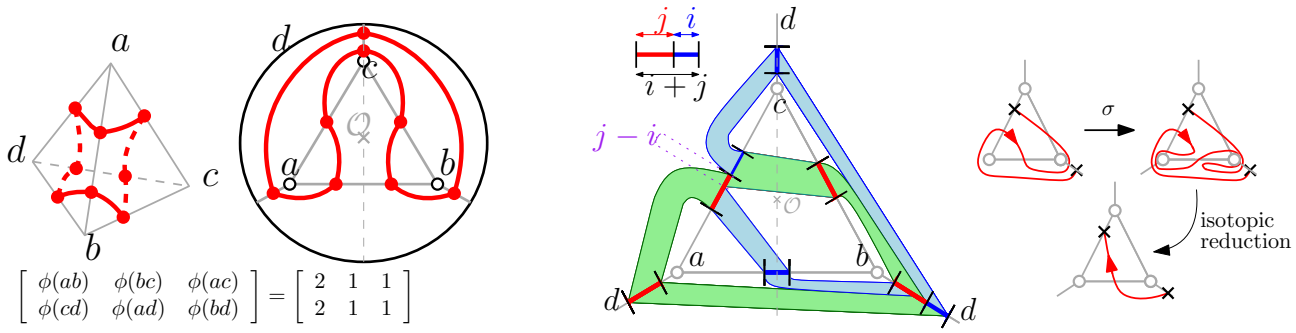



Figure 1: *Left:* Tetrahedron with a normal curve formed by 8 normal arcs. Equivalent representation of the curve as a reduced loop in the disk with punctures $\{a, b, c\}$, and $d$ sent to the boundary. *Middle:* Reduced loop with symbol $[i \;\; j \;\; i+j]$, $j \geq i$. Blue and green domains represent respectively $i$ and $j$ parallel segments. Note in particular that the $j$ segments, in the green domain inside the central triangle, originating from edge $bc$ and crossing edge $ac$, split into $j-i$ segments crossing $ad$ and $i$ segments crossing $cd$. *Right:* Action of $\sigma_{ab}$, with isotopic reduction, on a piece of loop and its neighbourhood (we call this loop configuration $O_1$).

(ii) loops separating one puncture from the three others; and (iii) loops separating two punctures from the two others. Note that here we call the outer boundary of $\mathbb{D}_3$ "puncture" as well. Naturally, loops of type (i) are trivial and have intersection symbol $[0 \;\; 0 \;\; 0]$, and loops of type (ii) can be isotoped to a circle in a small neighbourhood of the puncture in their inside, and hence have $(2 \times 3)$-intersection symbol $[1 \;\; 0 \;\; 1] \times [0 \;\; 1 \;\; 0]$, up to tetrahedron permutations. The case of loops of type (iii) is more interesting; we call these loops *balanced*. We prove:

**Lemma 2** *For any two loops $L_1$ and $L_2$ of same type (i), (ii) or (iii), there exists a homeomorphism of $\mathbb{D}_3$, constant on $\partial\mathbb{D}$, sending $L_1$ to $L_2$.*

This is a consequence of the Jordan-Schoenflies theorem. See the full version [3] for details.

We finally prove the main theorem of this article.

**Theorem 3** *There is a bijection between isotopy classes of balanced loops in $\mathbb{D}_3$ and intersection symbols of the form $[i \;\; j \;\; i+j]$, up to tetrahedron permutation, with $i$, $j$ coprime non-negative integers.*

**Proof.** Using Lemma 2, we know that any balanced loop may be obtained by a homeomorphism on the balanced loop $L_0$ with intersection symbol $[0 \;\; 1 \;\; 1]$, satisfying the theorem. Additionally, we know [1], via the isomorphism between *mapping class group* of $\mathbb{D}_3$ and *braid group*, that any such homeomorphism may be simulated by a sequence of the two homeomorphisms $\sigma_{ab}$ and $\sigma_{bc}$, exchanging punctures $a$ and $b$, and punctures $b$ and $c$. The proof is inductive: we study the action of these "braiding" homeomorphisms on a loop with intersection symbol $[i \;\; j \;\; i+j]$. This general loop is represented in Figure 1 middle, where "parallel" curves are represented by rays. It can be shown that for each crossing of the loop with, say, edge $ab$, a short loop segment around this crossing may be considered independently. As an example, assume we want to consider the action of $\sigma_{ab}$ on the loop $[i \;\; j \;\; i+j]$, and that in a neighbourhood of each intersection of the loop with edge $ab$, we observe the piece of loop $O_1$ shown in Figure 1 right; there are $i$ such intersections. Figure 1 illustrates the action of $\sigma_{ab}$ on the piece of loop that is modified, after reduction, only in a neighbourhood of the intersection with edge $ab$; in terms of intersection symbols, $\sigma O_1$ loses one intersection with edges $bc, ac, ad$ and $bd$. Having $i$ configurations of type $O_1$ crossing edge $ab$, applying $\sigma_{ab}$ on the loop $[i \;\; j \;\; i+j]$ we obtain the loop with intersection symbol $[i \;\; j-i \;\; j]$, which satisfies the theorem. See the full version [3] for a more detailed argument, and all remaining configurations of pieces of loops around a crossing.

The fact that $i$ and $j$ are coprime integers follows from the "braiding" of the loop $[0 \;\; 1 \;\; 1]$ to obtain any loop $[i \;\; j \;\; i+j]$ (and permutations), as it simulates the *Euclidean algorithm* on the pair $(i, j)$. Again, we refer to [3] for a complete proof. □

## References

[1] Joan S. Birman. *Braids, links, and mapping class groups*. Annals of mathematics studies. Princeton University Press, 1975.

[2] Benjamin A. Burton, Clément Maria, and Jonathan Spreer. Algorithms and complexity for Turaev-Viro invariants. In *Proceedings of ICALP 2015*, pages 281–293, 2015.

[3] Clément Maria and Jonathan Spreer. Admissible colourings of 3-manifold triangulations for Turaev-Viro type invariants. *CoRR*, abs/1503.04099, 2015.

# Homology preserving simplification for top-based representations

Federico Iuricich[1], Riccardo Fellegara[1] and Leila De Floriani[2]

[1]Department of Computer Science and UMIACS, University of Maryland, College Park (MD), USA.
[2]Department of Computer Science, Bioengineering, Robotics, and Systems Engineering, University of Genova, Genova, Italy.

## Abstract

Topological features provide global information about a shape, such as the number of the connected components, and the number of holes and tunnels. These are especially important in high-dimensional data analysis, where pure geometric tools are usually not sufficient. When dealing with simplicial homology, the size of the simplicial complex $\Sigma$ is a major concern, since all the algorithms available in the literature are mainly affected by the number of simplices of $\Sigma$. Edge contraction has been the most common operator for simplifying simplicial complexes. It has been used in computer graphics and visualization and more recently in topological data analysis. Edge contraction on its own does not preserve the homological information but a check, called link condition [2], has been developed for verifying whether the contraction of an edge preserves homology or not. However, since the number of simplices in the link of an edge grows exponentially when the dimension of the complex increases, checking the link condition is costly. In our work, we consider the definition of an homology preserving simplification algorithm, introducing a new way for verifying the link condition. We focus on a specific class of representations for simplicial complexes that we call top-based. A *top-based* representation encodes only the vertices and top simplices (also called *facets*) of a simplicial complex $\Sigma$, thus providing a data structure scalable with the dimension and size of $\Sigma$.

**Background notions** Given a *simplex of dimension p* (briefly a *p-simplex*), any simplex $\sigma$ which is the convex hull of a non-empty subset of the points generating $\tau$ is called a *face* of $\tau$. Conversely, $\tau$ is called a *coface* of $\sigma$. Given a $p$-simplex $\sigma$, the set of simplices for which $\sigma$ is a face is called star of $\sigma$ (also denoted $St(\sigma)$). If $St(\sigma) = \emptyset$, $\sigma$ is called *top simplex* (or *facet*). A *simplicial complex* $\Sigma$ is a finite set of simplices, such that each face of a simplex in $\Sigma$ belongs to $\Sigma$, and each non-empty intersection of any two simplices in $\Sigma$ is a face of both. We say that $\Sigma$ is a $d$-simplicial complex if the largest dimension of its simplices is $d$.

Let $\Sigma$ a $d$-simplicial complex, an *edge contraction* acts on $\Sigma$ by contracting an edge $\varepsilon = (v_1, v_2)$ to one of its vertices (i.e. $v_1$). As a result, all the simplices in $St(\varepsilon)$ are removed from $\Sigma$ and all the simplices in $St(v_1) - St(\varepsilon)$ are mapped into $St(v_2)$ in such a way that, for each simplex $\sigma \in St(v_1) - St(\varepsilon)$, $\mu(\sigma) = (\sigma - v_1) \cup v_2$. Thus, edge contraction is an operation linear in the number of simplices in $St(v_1)$.

The *link* of a simplex $\sigma \in \Sigma$, denoted as $Lk(\sigma)$, is the set of faces of $St(\sigma)$ that do not intersect $\sigma$. An edge $\varepsilon = (v_1, v_2) \in \Sigma$ satisfies the *link condition* if and only if $Lk(v_1) \cap Lk(v_2) = Lk(\varepsilon)$. For reducing the computational cost of extracting the links all at once, a *weaker* condition, called $p$-link condition, has been introduced in [3]. An edge $\varepsilon = (v_1, v_2)$ satisfies the *p-link condition* if and only if either $p \leq 0$ or $p > 0$ and every $(p-1)$-simplex $\in Lk(v_1) \cap Lk(v_2)$ is also in $Lk(\varepsilon)$. Thus, an edge $\varepsilon = (v_1, v_2)$ satisfies the link condition if and only if it satisfies the $p$-link condition for all $p \leq d$. Despite the fundamental reduction in the computational cost, the numerosity of $p$-simplices in the link of two vertices, still, can be huge depending on the dimension $d$ of the complex. We consider solving this problem, by adapting the edge contraction and the link condition to perform on a top-based representation.

**Top-based homology preserving edge contraction** Encoding only the top-simplices and the vertices we can perform an edge contraction $\varepsilon = (v_1, v_2)$ by focusing on the simplices $St_{top}(\varepsilon)$, i.e. the set of top-simplices incident in the edge removed, and $St_{top}(v_1)$, i.e. the set of top-simplices incident in the vertex removed with $\varepsilon$. The key point is to modify the set of simplices maintaining the top-based representation valid. We recall that, in a top-based representation, each simplex $\sigma \in \Sigma$ is encoded if and only if $\sigma$ is a vertex or a top simplex. Thus, while removing the set of top simplices incident in $\varepsilon$, it is crucial to recognize if new top simplices must be introduced.

Algorithm 1 describes the procedure that can be implemented on a top-based representation for performing an edge contraction. Each top $p$-simplex $\sigma \in St_{top}(\varepsilon)$ is removed with the edge (rows 5 to 10). By definition of edge contraction, all the faces of $\sigma$ are removed with the exception of the $(p-1)$-faces $\gamma_1 = (\sigma - v_2)$ and $\gamma_2 = (\sigma - v_1)$, which will be merged in a single $(p-1)$-face (for example $\gamma_2$). Let $S = St_{top}(\gamma_1) \cup St_{top}(\gamma_2)$ be the set of top simplices in the star of either $\gamma_1$ or $\gamma_2$. $S$ cannot be empty before the edge contraction, as, at least $\sigma$ belongs to $S$. By merging $\gamma_1$ and $\gamma_2$ while removing $\sigma$, the star of the new simplex will be $St(\gamma_2) = S - \sigma$. Then, $\gamma_2$ is a new top simplex if and only if $St(\gamma_2) = \emptyset$. Notice that this condition can be verified before

This is an abstract of a presentation given at CG:YRF 2016. It has been made public for the benefit of the community and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear in a conference with formal proceedings and/or in a journal.

**Algorithm 1** contractEdge($\varepsilon$,$\Sigma$)

1: **Input:** $\Sigma$ is a simplicial complex
2: **Input:** $\varepsilon = (v_1, v_1)$ edge to be contracted
3: **Output:** $\Sigma'$ is a simplified simplicial complex
4: // *For each top simplex in the star of* $\varepsilon$
5: **for each** $\sigma$ in $St_{top}(\varepsilon)$ **do**
6: $\quad\gamma_1 = (\sigma - v_2)$
7: $\quad\gamma_2 = (\sigma - v_1)$
8: $\quad$**if** $St_{top}(\gamma_1) \cup St_{top}(\gamma_2) = \sigma$ **then**
9: $\quad\quad$addTop($\gamma_2$,$\Sigma$)
10: $\quad$removeTop($\sigma$,$\Sigma$)
11: // *For each top simplex in the star of* $v_1$
12: **for each** $\sigma$ in $St_{top}(v_1)$ **do**
13: $\quad\sigma = (\sigma - v_1) \cup v_2$
14: removeVertex($v_1$)

performing the edge contraction by checking if $S = \sigma$ (rows 8 to 9). Then, working on the set of top simplices incident in $v_1$ (rows 12 to 13), we update $St_{top}(v_1)$, replacing $v_1$ with $v_2$, without modifying the star of any other simplex. Finally, we remove $v_1$ from $\Sigma$ (row 14). The edge contraction becames here aan operation linear in the number of simplices in $St_{top}(v_1)$.

The link condition can be efficiently verified exploiting the top-based representation as well. From the definition of link, we can trivially say that $Lk(\varepsilon) \subseteq \{Lk(v_1) \cap Lk(v_2)\}$ thus, the link condition is satisfied when also the opposite is true. Notice that, the link of a simplex is a simplicial complex and, thus, also the intersection of two simplicial complexes is still a simplicial complex. So, we can conclude that the link condition is satisfied if the top simplices in $L = Lk(v_1) \cap Lk(v_2)$ are also in $Lk(\varepsilon)$. Computing the top simplices of $L$ is much faster than computing the links, but, still, it is an expensive operation. Let $T_s$ the set of simplices obtained by pairwise intersecting the simplices in $St_{top}(v_1)$ and $St_{top}(v_2)$. The top simplices of $L$ would be obtained by removing from $T_s$ those simplices that are not maximal in $L$. However, to improve scalability, we avoid storing $T_s$ thus considering all the simplices originated by the intersection. The space complexity of the top-based approach is then $O(|T_1| + |T_2|)$ since we only need to store the top simplices incident in $v_1$ and $v_2$ while the time complexity is $O(|T_1||T_2|)$, thus, depending on the size of $T_1$ and $T_2$. In practice, it is computationally faster than the traditional (*weak*) link condition since it avoids: (i) the extraction of the faces of the simplices $\sigma \in T_1, T_2$ ($2^d - 1$ faces for each simplex); (ii) the intersection of the two sets $Lk(v_1)$ and $Lk(v_2)$; (iii) and the comparison of the resulting set with $Lk(\varepsilon)$.

**Experimental results** In our evaluation, we use eleven simplicial complexes having from 9 thousand to 14 millions vertices. The dimension of the top simplices goes from 7 to 68. The hardware configuration used for these experiments is an Intel i7 3930K CPU at 3.20Ghz with 64 GB of RAM. We have implemented the simplification approach, using a specific top-based representation, the *Stellar tree* [4]. On the top of it we have implemented both the *weak* link condition [2] and the new *top-based* approach for verifying the link condition. The size of each simplicial complex $\Sigma$ is then reduced applying homology preserving edge contractions until no more can be applied without changing the homology of $\Sigma$. A simplification process is killed after 25 hours of computation. The simplification ratio is, on average, around 90%-95% of the initial number of vertices. From the results obtained the limitation of the approach based on the *weak* link condition is evident. Using the latter, the simplification process ends in very few cases (two datasets). Typically the computation exceeds the 25 hours and, in two cases, the process exceeds the 64GB of memory. Verified the computational improvement for checking the link condition we have evaluated the practical relevance of the proposed approach comparing the performances of our implementation with respect to the state-of-the-art data structure for performing edge contractions, the *Skeleton-Blocker* [2] (as implemented in [1]). From the results obtained we can say that the Stellar tree is generally faster taking 25% to 70% of the time required by the Skeleton-Blocker for simplifying a dataset. Focusing on the timings distribution, the Skeleton-Blocker needs more effort for updating the structure during the edge contraction while it is particularly fast at verifying the link condition. Conversely, the Stellar tree is faster at performing the contraction but it is slower at checking the link condition. This is an expected result due to the characteristics of the two data structures (details are are not included for brevity). Analyzing the memory peak, i.e. the amount of memory used at runtime for performing simplification, the Stellar tree is also more compact, using from 30% to 80% of memory required by Skeleton-Blocker.

## References

[1] Geometric Understanding in Higher Dimensions (GUDHI), 2016.

[2] D. Attali, A. Lieutier, and D. Salinas. Efficient Data Structure for Representing and Simplifying Simplicial Complexes in High Dimensions. In *International Journal of Computational Geometry & Applications*, volume 22, pages 279–303, New York, New York, USA, 2012. ACM Press.

[3] T. K. Dey, A. N. Hirani, B. Krishnamoorthy, and G. Smith. Edge Contractions and Simplicial Homology. *arXiv:1304.0664*, apr 2013.

[4] R. Fellegara. *A spatio-topological approach to the representation of simplicial complexes and beyond*. PhD thesis, Department of Computer Science (DIBRIS), Universitá degli Studi di Genova, 2015.

# Towards the Analysis of Multivariate Data based on Discrete Morse Theory

Sara Scaramuccia[1], Federico Iuricich[2], Claudia Landi[3] and Leila De Floriani[1]

[1]Department of Computer Science, Bioengineering, Robotics, and Systems Engineering, University of Genova, Genova, Italy.
[2]Department of Computer Science and UMIACS, University of Maryland, College Park (MD), USA.
[3]Department of Sciences and Methods for Engineering DISMI, University of Modena and Reggio Emilia, Modena-Reggio Emilia, Italy

## Abstract

Morse theory [3] studies the relationships between the topology of a shape and the critical points of a real-valued smooth function defined on it. Recently a discrete counterpart, called Discrete Morse Theory, has been proposed in an entirely combinatorial setting by Robin Forman for cell complexes [2]. Discrete Morse theory is the basis for computing Morse decompositions of discretized geometric shapes, and it provides a tool for retrieving topological invariants, like homology groups. In particular, the Forman gradient is the basis for an efficient and derivative-free computation of a segmentation of a discretized shape endowed with a scalar function $f$. It fully encodes the topological structure of the function and of its domain and the regions of influence of the critical points. Discrete Morse theory defines a vector as a pair of cells $(\sigma,\tau)$ where $\sigma$ is a $k$-cell in the immediate boundary of $\tau$ (i.e. their dimension differs by 1). A *discrete vector field* $V$ defined on a CW-complex $\Gamma$ is a collection of pairs such that each cell of $\Gamma$ is in at most one vector of $V$. A *critical cell* of $V$ of index $k$ is a $k$-cell $\gamma$ which does not appear in any pair of $V$. A *$V$-path* is a sequence $\sigma_1,\tau_1,\sigma_2,\tau_2,...,\sigma_r,\tau_r$ of $k$-cells $\sigma_i$ and $(k+1)$-cells $\tau_i, i=1,..,r$ with $r \geqslant 1$, such that $(\sigma_i,\tau_i) \in V$, $\sigma_{i+1}$ is a face of $\tau_i$, and $\sigma_i \neq \sigma_{i+1}$. A $V$-path with $r>1$ is *closed* if $\sigma_1$ is a face of $\tau_r$ different from $\sigma_{r-1}$. A discrete vector field $V$ is called a *Forman gradient* if and only if there are no closed V-paths in $V$ (see Figure 1c). A Forman gradient can be computed from a scalar field defined on the vertices of a CW-complex $\Gamma$, extending the scalar function from the vertices of $\Gamma$ to all its cells. In [4] a dimension-independent algorithm is proposed that processes the lower star of each vertex independently, where the *lower star* of a cell $\gamma$ is the subset of cells incident in $\gamma$ containing only cells with a lower function value than $\gamma$. By means of the lower star subdivision, we obtain a domain partition covering $\Gamma$. Cells are paired if they belong to the same lower star and if they have the same function value. The resulting algorithm is then easy to parallelize and it has been proved to identify critical cells in one-to-one correspondence with the topological changes in the sub-level sets (i.e. no spurious critical simplexes are created). Extending topological methods to multivariate functions is an active research area that has been investigated based on few techniques, including Jacobi sets, Reeb spaces, joint contour nets, and Pareto sets. For discrete Morse theory, the main problem arising in the multidimensional setting is that the lower star subdivision no longer provides a cover for $\Gamma$. More precisely, the vertex lower stars still do not intersect each other but they do not necessarily cover the entire domain. In [1] a solution has been proposed considering the lower star of each cell rather than for each vertex. Possible double classifications are avoided by defining a flag for each cell of $\Gamma$. The resulting approach is computationally expensive since it requires each cell of $\Gamma$ to be explicitly encoded. Moreover, by imposing a specific order on the cells of $\Gamma$, there is no trivial way to parallelize the computation.

Let $\Gamma$ be a CW-complex and $\Gamma_0 \xrightarrow{f} \mathbb{R}^n$ be a function on the set of vertices $\Gamma_0$, $f$ is required to be component-wise injective. We extend $f$ to all the cells $\gamma \in \Gamma$ by taking the least common upper bound of the $f$ values on their vertices. We denote such extended function, $F$.

The idea at the base of both [4] and [1] was to group cells in $\Gamma$ where possible pairings can be found. We refer to these subsets of $\Gamma$ as *filtration-based lower stars* (denoted $\mathrm{L}_F$). For the multidimensional settings the filtration-based lower star of a cell $\gamma \in \Gamma$ has been defined in [1] as

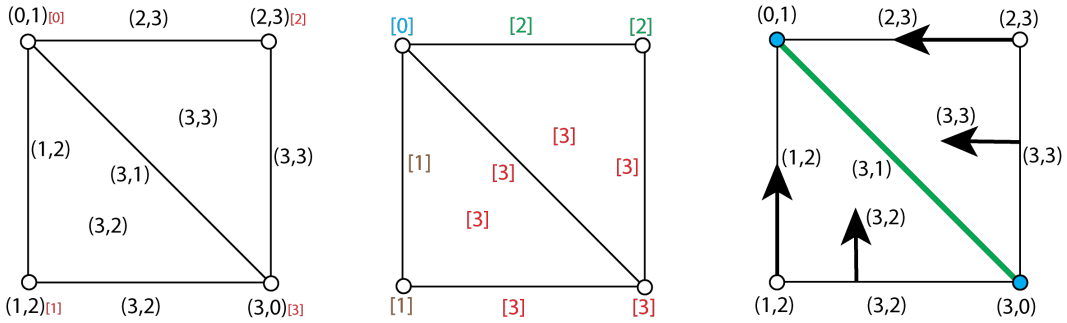
$$\mathrm{L}_F(\gamma) := \{\tau \in \Gamma \mid \gamma \subseteq \tau \wedge F(\tau) \preceq F(\gamma)\}$$

where $\preceq$ is the partial order on $\mathbb{R}^n$, $u \preceq v$ if and only if $u_i \leqslant v_i$ for all components.

We propose a new approach consisting of a generalization of the algorithm defined in [4] but still based on collecting the lower stars of the vertices only. We consider an indexing $I$ created on the vertices of $\Gamma$, based on $f$ (see Figure 1a). The new called *index-based lower stars* (denoted $\mathrm{L}_I$) are computed as follows. For each vertex $v$,

$$\mathrm{L}_I(v) := \{\gamma \in \Gamma \mid \tilde{I}(\gamma) = I(v)\}$$

where $\tilde{I}(\gamma)$ denotes the maximum of values $I(w)$ of the vertices $w$ of $\gamma$. In other words, given a vertex $v \in \gamma$, we say that $\gamma$ belongs to the lower star of $v$ if and only if $v$ is the vertex with index maximum among the vertices of $\gamma$ (see Figure 1b). Once the lower stars have been separated, cells




(a) (b) (c)

Figure 1: (a) The vector-valued function is extended from the vertices of $\Gamma$ to all the cells. The vertex indexing is depicted within square brackets. (b) The lower star of each vertex is computed based on the vertex indexing. Cells having the same index belong to the same lower star. (c) The Forman gradient is computed, on each lower star, pairing cells with the same function value. Critical vertices are depicted in blue and the critical edge is depicted in green.

can be paired independently in each one. Still, we admit two cells $\sigma, \tau$ to be paired only when they have the same function value (see Figure 1c). Notice that if two cells belong to the same lower star, this does not mean that they have the same function value (details on the pairing algorithm are skipped for brevity).

It is easy to see that both $\mathrm{L}_F$ and $\mathrm{L}_I$ cover the entire domain but, in order for the index-based partition not to affect any possible pairing of cells, we need to show that they are coherent (i.e. each filtration-based lower star must be contained in only one indexed-based lower stars). To this purpose, we require the indexing $I$ to be *well-extensible* with respect to the function $f$, i.e., for every two cells $\gamma$ and $\tau$, if $F(\gamma) \preceq F(\tau)$, then $\tilde{I}(\gamma) \leqslant \tilde{I}(\tau)$. A trivial way to obtain a well-extensible indexing is by ordering the vertexes according to the values of a single component $f_i$. This ensures that pair of cells belonging to the same filtration-based lower star are in the same index-based lower stars. The coherence between the index-based partition and filtration-based lower stars can be formalized as follow. If $I$ is a well-extensible indexing with respect to $f$, it follows that

$$\forall \gamma \in \Gamma, \quad \exists! v \in \Gamma_0, \quad \mathrm{L}_F(\gamma) \subseteq \mathrm{L}_I(v)$$

and $v$ belongs to $\gamma$. Indeed, let $\gamma$ be a cell in $\Gamma$. It is easy to check that there exists a unique vertex $v \in \gamma$ such that $\gamma$ belongs to $\mathrm{L}_I(v)$, the index-based lower star of $v$. Let $\alpha$ be a cell in $\mathrm{L}_F(\gamma)$. By definition of lowerstar, $\alpha \supseteq \gamma$ and $F(\alpha) \preceq F(\gamma)$. The former condition implies that $\tilde{I}(\alpha) \geqslant \tilde{I}(\gamma)$ and that $v \in \alpha$. The latter condition together with the assumption on $I$ being well-extensible give $\tilde{I}(\alpha) \leqslant \tilde{I}(\gamma)$. Hence, $\tilde{I}(\alpha) = \tilde{I}(\gamma) = I(v)$, which concludes the proof. Subdividing $\Gamma$ based on the new index-based lower star presents practical improvements with respect to [1]. Indeed, lower stars are retrieved locally and independently within the vertex-based partition, rather than globally. This implies fewer cells to be stored in memory at runtime and easy parallelization of computation, thus allowing for real world applications to data sets of medium and large size. Moreover, the proposed approach has been proved to retrieve a discrete vector field compatible with the multidimensional persistent homology. When $\Gamma$ is a simplicial complex, a vector-valued function $F$ induces a multidimensional filtration by sublevel sets on $\Gamma$. Then we can consider the multidimensional persistent homology of $\Gamma$ with respect to such filtration. In analogy with the Morse reductions in the single-valued setting, it has been proved that, if one reduces the complex $\Gamma$ according to the pairings of the output discrete vector-field, the reduction returns a smaller complex with the same multidimensional persistent homology with respect to $F$ as that of $\Gamma$. Our algorithm inherits the property of compatibility with the multidimensional persistent homology. It promises to be useful both for the computation of multidimensional persistent homology of simplicial complexes filtered by sublevel sets of vector-valued functions as well as for developing new visualization techniques for representing topological features of datasets characterized by multiple scalar fields.

## References

[1] M. Allili, T. Kaczynski, C. Landi, and F. Masoni. A new matching algorithm for multidimensional persistence. *ArXiv, Id:1511.05427*, Nov 2015.

[2] R. Forman. Morse theory for cell complexes. *Advances in mathematics*, 134(1):90–145, 1998.

[3] J. W. Milnor. *Morse Theory*. Annals of Mathematics Studies. Princeton University Press, 1963.

[4] 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, 2011.

# Local Structures for Approximating Rips Filtrations

Aruni Choudhary* Michael Kerber† Sharath Raghvendra‡

**Abstract**

(Vietoris-)Rips filtrations are important structures used to infer topological properties of metric spaces. Unfortunately, they pose significant computational challenges, particularly when the data has high dimension. We present a new technique to $O(\sqrt{d})$-approximate the topological information carried by the Rips filtration which can be represented using at most $n2^{O(d \log d)}$ $d$-simplices per scale in the filtration.

## 1 Introduction

Given a metric space, for instance a point cloud, one is often interested in studying its topological properties. A common requirement is to study those topological features of the space which persist over a large range of scales. Longer persisting features often correspond to important structures in the data. A preliminary step in computing such features is to build a simplicial complex on the point set. One such simplicial complex is the *Rips* complex, which we denote as $\mathcal{R}$. It is defined on a point set $P$ with respect to a scale parameter $\alpha$, $\mathcal{R}_\alpha = \{Q \subseteq P | diam(Q) \leq 2\alpha\}$. Varying the scale parameter over a range of scales, one gets a sequence of simplicial complexes called a filtration. The complexes of a filtration are connected by inclusion maps, and give rise to the persistence module $(H_*(\mathcal{R}_\alpha))_{\alpha \geq 0}$, which contains relevant topological information.

Although the Rips filtration is simple to describe, it poses computational challenges: for instance, given $n$ points in $\mathbb{R}^d$, the $k$-skeleton of $\mathcal{R}$ has size $O(n^{k+1})$, where $k \leq d$, which is practically feasible only for very small $k$. This makes it impractical to compute persistence of higher dimensional features, specially when $d$ is large. The common way to remedy this issue is to compute an approximate filtration which roughly preserves the topological information but is much more attractive computationally.

Sheehy [3] was the first to come up with an approximation scheme for the Rips filtration. The $(1+\varepsilon)$-approximate filtration had a $k$-skeleton of size $n(1/\varepsilon)^{O(k\delta)}$, $\delta$ being the doubling dimension of the space. Although the bound is linear in $n$, for large $\delta$ and $k$, this is prohibitively expensive. Recently, a $O(d)$-approximation scheme of size $n2^{O(d \log k)}$ per scale was proposed in [1]. Combined with dimension reduction techniques, this is the first scheme with polynomial complexity independent of dimension.

In this work, we show the existence of a $O(\sqrt{d})$-approximate Rips filtration. Our filtration consists of cell complexes connected by continuous maps at different scales. The cell complex has complexity at most $n2^{O(d \log d)}$ at any scale. This improves the approximation factor over our previous result in [1]. The reliance on continuous maps and cell complexes makes a computational treatment of our construction challenging. We discuss our ideas to address this issue at the end of this abstract.

## 2 Approximation scheme

**Construction** We denote by $P$, a set of $n$ points in $\mathbb{R}^d$. To construct the approximation complex, we use the grid lattice. Let $G$ denote a scaled version of the $d$-dimensional integer grid lattice with basis vectors $(\alpha e_1, \ldots, \alpha e_d)$ where $(e_1, \ldots, e_d)$ are the standard basis vectors of $\mathbb{R}^d$. $G$ can also be visualized as collection of $d$-dimensional hypercubes, whose vertices form $G$. Each such hypercube has $2^d$ vertices. We call such hypercubes *cells*.

For each point $p \in P$, we find its closest point in $G$ (we assume generic position of $P$ for simplicity, such that the closest point is unique). We call those points in $G$ whose Voronoi cell contains at least one point of $P$ as *full vertices*. The full vertices of $G$ are the vertices of our approximation complex. Consider any cell $C$ which is non-empty, that is, it contains at least one full vertex. Let $F_C$ be the set of full vertices incident to $C$. Then, we define

- $\mathcal{ULS}_\alpha$: let $\sigma_C$ denote the (abstract) simplex spanned by the vertices $F_C$. Note that $\sigma_C$ does not embed in $\mathbb{R}^d$ if $|F_C| \geq d+2$ for at least one $C$. We define $\mathcal{ULS}_\alpha$ as the abstract simplicial complex whose maximal simplices are the $\sigma_C$'s. In other words, $\mathcal{ULS}_\alpha$ is the union of simplices on the full vertices of each non-emtpy cell.
- $\mathcal{ULH}_\alpha$: let $H_C$ denote the convex hull of the vertices $F_C$. Then $\mathcal{ULH}_\alpha = \cup H_C$ over all non-empty cells of $G$. In other words, $\mathcal{ULH}_\alpha$ is the union of local convex hulls of the full vertices of each cell.

*Max Planck Institute for Informatics, Saarbrücken, Germany (aruni.choudhary@mpi-inf.mpg.de)
†Graz University of Technology, Austria(kerber@tugraz.at)
‡Virginia Tech, Blacksburg, USA (sharathr@vt.edu)

**Relation between $\mathcal{ULS}$ and Rips** By varying the scale $\alpha$, one can build a discrete filtration on $\mathcal{ULS}_\alpha$. We show that by using a suitable choice of scales, it is possible to approximate the Rips filtration using $\mathcal{ULS}$.

At any scale $\alpha$, we define two maps, $a_\alpha$, which maps a point in $P$ to its closest full vertex and $b_\alpha$ which maps a full vertex to its closest point in $P$. These vertex maps induce the simplicial maps $\phi_\alpha : \mathcal{R}_{\alpha/2} \to \mathcal{ULS}_\alpha$ and $\psi_\alpha : \mathcal{ULS}_\alpha \to \mathcal{R}_{\alpha 3\sqrt{d}}$ respectively and give rise to the diagram

$$\begin{array}{ccccccc} \cdots & \longrightarrow & \mathcal{R}_{\alpha/2} & \hookrightarrow & \mathcal{R}_{c\alpha/2} & \longrightarrow & \cdots \\ & & \downarrow{\scriptstyle\phi} & \nearrow{\scriptstyle\psi} & \downarrow{\scriptstyle\phi} & & \\ \cdots & \longrightarrow & \mathcal{ULS}_\alpha & \xrightarrow{\theta} & \mathcal{ULS}_{c\alpha} & \longrightarrow & \cdots \end{array} \tag{1}$$

where $c = 6\sqrt{d}$ and $\theta = \phi \circ \psi$ induces a discrete filtration $\left(\mathcal{ULS}_{\alpha c^k}\right)_{k\in\mathbb{Z}}$. It follows that the persistence module $\left(H_*(\mathcal{ULS}_{\alpha c^k})\right)_{k\in\mathbb{Z}}$ is a $c$-approximation of $(H_*(\mathcal{R}_\alpha))_{\alpha\geq 0}$. However, the size of $\mathcal{ULS}_\alpha$ per scale is $n2^{O(dk)}$, which is no improvement over existing $(1+\varepsilon)$-approximation techniques [3].

**Relation between $\mathcal{ULS}$ and $\mathcal{ULH}$** We write $A \overset{h}{\simeq} B$ if the spaces $A$ and $B$ are homotopically equivalent. We show that $\mathcal{ULS}_\alpha \overset{h}{\simeq} \mathcal{ULH}_\alpha$ using the following maps:

- $f : \mathcal{ULS}_\alpha \to \mathcal{ULH}_\alpha$: let $x$ be a point in a simplex $(e_1, \ldots, e_k) \in \mathcal{ULS}_\alpha$. Then $x = \sum_{i=1}^k a_i e_i$, for some $\sum a_i = 1, a_i \geq 0$. We set $f(x) = \sum_{i=1}^k a_i v_i$, where $v_i$ is the vertex in $\mathcal{ULH}_\alpha$ corresponding to $e_i$.

- $g : \mathcal{ULH}_\alpha \to \mathcal{ULS}_\alpha$: let $\mathcal{ULT}_\alpha$ be a triangulation of the space $\mathcal{ULH}_\alpha$. This triangulation can be generated by a union of local Delaunay triangulations in combination with simulation of simplicity. Any point $x \in \mathcal{ULH}_\alpha$ lies in the interior of a unique $k$-simplex $(v_0, \ldots, v_k)$ in $\mathcal{ULT}_\alpha$. It follows that $x = \sum_{i=0}^k \lambda_i v_i$, for some $\sum \lambda_i = 1, \lambda_i \geq 0$. We set $g(x) = \sum_{i=0}^k \lambda_i e_i$.

It can be verified that $f \circ g$ and $g \circ f$ are indeed homotopic to the respective identity maps on $\mathcal{ULH}_\alpha$ and $\mathcal{ULS}_\alpha$, proving the homotopic equivalence. The connections between $\mathcal{ULH}$ and $\mathcal{R}$ can be summarized as in Figure 1. By functoriality of homology, it follows that

**Theorem 1** *The module $\left(H_*(\mathcal{ULH}_{\alpha(6\sqrt{d})^k})\right)_{k\in\mathbb{Z}}$ is a $6\sqrt{d}$-approximation of $(H_*(\mathcal{R}_\alpha))_{\alpha\geq 0}$.*

$$\begin{array}{ccccccc} \cdots & \longrightarrow & \mathcal{R}_{\alpha/2} & \hookrightarrow & \mathcal{R}_{c\alpha/2} & \longrightarrow & \cdots \\ & & \downarrow{\scriptstyle\phi} & \nearrow{\scriptstyle\psi} & \downarrow{\scriptstyle\phi} & & \\ \cdots & \longrightarrow & \mathcal{ULS}_\alpha & \xrightarrow{\theta} & \mathcal{ULS}_{c\alpha} & \longrightarrow & \cdots \\ & & {\scriptstyle g}\uparrow\downarrow{\scriptstyle f} & & {\scriptstyle g}\uparrow\downarrow{\scriptstyle f} & & \\ \cdots & \longrightarrow & \mathcal{ULH}_\alpha & \xrightarrow{\eta} & \mathcal{ULH}_{c\alpha} & \longrightarrow & \cdots \end{array} \tag{2}$$

Figure 1: Diagram of spaces involved in our construction. Here $\eta = f \circ \theta \circ g$ is chosen to make the diagram commute and $c = 6\sqrt{d}$.

## 3 Outlook

Our filtration from Theorem 1 is a sequence of cell complexes connected by continuous maps. This is in contrast to known approximation schemes, where typically, a filtration is a sequence of simplicial complexes connected by inclusions or, more generally, simplicial maps. Clearly, a transition to such a combinatorial category appears necessary for an efficient computation of the persistent homology of the approximation.

There are several ways for converting $\mathcal{ULH}_\alpha$ into a simplicial complex; for instance, we can triangulate the convex hull of each cell locally such that the triangulations are consisted along common boundaries. Another approach is to pass to the barycentric subdivision. Both approaches yields quite small simplicial complex on each scale (in the same magnitude as in [1]). However, the maps $f$ and $g$ introduced earlier do not extend to simplicial maps between simplicial complexes on different scales, in general. In the spirit of the *simplicial approximation theorem* [2, §16] we are currently investigating whether there exists an (easily computable) simplicial map $h : \mathrm{sd}(S_\alpha) \to S_{c\alpha}$ between triangulations of $|\mathcal{ULH}_\alpha|$ and $|\mathcal{ULH}_{c\alpha}|$, where $\mathrm{sd}(S_\alpha)$ is the barycentric subdivision of $\mathrm{sd}(S_\alpha)$.

## References


[1] A. Choudhary, M. Kerber, and S. Raghvendra. Polynomial-sized Topological Approximations using the Permutahedron. In *32nd International Symposium on Computational Geometry*, 2016.

[2] J.R. Munkres. *Elements of algebraic topology*. Westview Press, 1984.

[3] D. Sheehy. Linear-size approximations to the Vietoris-Rips Filtration. *Discrete & Computational Geometry*, 49(4):778–796, 2013.

# Exact and Approximation Algorithms for Time-Window TSP

Jie Gao* Su Jia* Joseph S. B. Mitchell*

**Abstract**

We study a version of the *time-window traveling salesman problem (TWTSP)*: given a speed bound $s \leq \infty$ for a robot, a set of cities each having a time window during which it must be visited, find a shortest path for the robot to visit all cities within their respective time windows, if possible to do so. We give an exact algorithm for 1D instances with dyadic time windows and $s = \infty$ and a pseudo-polynomial time algorithm for $s < \infty$. For $s < \infty$ in a general metric space, when there are only two different time window lengths, we present an $(\alpha, \beta)$ dual approximation algorithm, using speed $\leq \alpha \cdot s$ and travel distance $\leq \beta \cdot d^*(s)$, where $d^*(s)$ is the shortest TSP path with speed bound $s$, $\alpha, \beta = O(1)$. Then we generalize these results to instances with arbitrary time windows and obtain an $(\alpha, \beta)$ dual approximation for $\alpha, \beta = O(\log L)$, where $L$ is the length of the longest time window (assuming the shortest time window is 1). We also generalize our methods to other time window problems in the TSP family.

## 1 Introduction

A natural variant of the classical TSP is to include a preferred time window for each city/job to be visited. This constraint appears in many scenarios: delivery of time-critical packages; data gathering and battery servicing of distributed sensors; etc. We consider the *time window traveling salesman problem (TWTSP)*: given a speed bound $s \leq \infty$, find a shortest path (if it exists) that visits each site $i$ within the time window $[r_i, d_i]$, for release time $r_i$ and deadline $d_i$.

Note that if $s$ is too small, there is no feasible solution. Let $s_{min}$ be the smallest speed $s$ such that there exists a feasible solution. For a given speed bound $s \geq s_{min}$, let $d^*(s)$ be the minimum possible travel distance. For $s \geq s_{min}$, an algorithm is an $(\alpha, \beta)$ dual approximation if the robot moves with speed $\leq \alpha s$, and travels distance $\leq \beta d^*(s)$.

*Related Work.* The problem of enforcing time windows in the TSP has been studied before, e.g., by Bansal *et al.* [1]): given $n$ sites in a metric space, each with a time window $[r_i, d_i]$, and a speed bound $s$, we wish to find a path/tour that visits *as many points* in their time window as possible. This problem of maximizing the number of sites might be called the *Time Window Prize Collecting TSP*. Bansal *et al.* [1] gave an $O(\log n)$-approximation for instances having release times $r_i = 0$ and arbitrary deadlines $d_i$, and an $O(\log^2 n)$-approximation for arbitrary time windows. The TWTSP has also been studied in the operations research literature, using integer programming and branch-and-bound techniques. These algorithms are not reviewed here, as our emphasis is on polytime approximation algorithms.

*Stony Brook University, Stony Brook, NY 11794. {jie.gao, su.jia, joseph.mitchell}@stonybrook.edu.

## 2 Case of Infinite Speed ($s = \infty$)

We say a set of time windows are *dyadic*, if the length of each time window is a power of two, and the release time is a nonnegative integer multiple of its length.

**Theorem 1** *For dyadic time windows, the 1D TWTSP problem with infinite speed ($s = \infty$) can be solved in polynomial time.*

**Theorem 2** *For $s = \infty$ and arbitrary time windows, there is a polytime $O(\log n)$-approximation in a metric space.*

## 3 Case of Bounded Speed ($s < \infty$)

### 3.1 One Dimension: TWTSP on a Line

The 1D version can be viewed as a 2D problem in space-time. Using dynamic programming, we prove the following result:

**Theorem 3** *Consider the 1D TWTSP with speed bound $s < \infty$, dyadic time windows with largest window length $L$, integer release times, and for each $i \leq L$ assume that there is at least one site with time window $[i, i+1]$. Then, in time polynomial$(n, s, L)$ we can either find $d^*(s)$ or report that there is no feasible solution for $s$.*

### 3.2 Bounded Speed in a Metric Space: A Special Case

We begin with a special case:

1. All release times are integers;

2. Each time window is either unit-length or $[0, L]$, for an integer $L$;

3. For each $i \leq L$, there is at least one site with time window $[i, i+1]$.

We refer to such an instance as a "simple instance"; our more general results build on this case.

Given a bipartite graph $H = (V_R \cup V_B, E)$, a subgraph $M$ is a $\Delta$-*matching* if for any $v \in V(M) \cap V_R$, $deg_M(v) = 1$ and for any $v \in V(M) \cap V_B$, $deg_M(v) \leqslant \Delta$. Given a tree $T$, and a number $\rho$, we let $shad(T) = \cup_{v \in V(T)} D_\rho(v)$ and call it the $\rho$-*shadow* (or simply *shadow*) of $T$, where $D_\rho(v)$ is the disk centered at $v$ with radius $\rho$. Call the sites with unit-length time windows *red sites* and the other sites *blue sites*. Our algorithm consists of the following steps:

1. Find an MST for all sites, and cut this tree into a set $\mathcal{T}$ of subtrees $T_1$,...,$T_k$, all but one having size at least $C_1 s$ and at most $C_2 s$, with the exceptional piece of size $< C_1 s$.

2. Build an unweighted bipartite graph on red sites $v_i$ and subtrees $T_j$, with an edge $(r_i, T_j)$ if they are at distance $\leq s$. Denote this graph $H = (V_R \cup \mathcal{T}, E)$.

3. Find a $\Delta$-matching in $H$.

4. For each red site, traverse the subtrees assigned to it, and connect the red sites respecting the order of their time windows.

**Theorem 4** *Under the assumptions above, this algorithm is an $(O(1), O(1))$ dual approximation.*

The proof relies on the following lemmas.

**Lemma 5** *Given a tree $T$ whose max edge length is less than $s$, and given any $C_1 \geq 1$ and $C_2 \geq C_1 + 1$, we can partition it into subtrees $T_i's$, s.t.*

1. *At most one subtree $T$ does* not *satisfy $C_1 s \leq |T| \leq C_2 s$. Moreover, this can be done in linear time. Call the exceptional piece the* remainder piece*;*

2. *The depth of overlapping is at most 2, i.e. each node is contained in at most two subtrees.*

**Lemma 6** *Recall that $\mathcal{T}$ is a family of trees, denote $|\mathcal{T}|$ the number of trees it contains. Let $\alpha_{St}$ be the Steiner ratio. Then, $|\mathcal{T}| \leq \frac{\alpha_{St}}{C_1} |V_R|$.*

**Lemma 7 (Generalized Hall's Marriage Theorem)** *In a bipartite graph $G = (V_R \cup V_B, E)$, if for any subset $A$ of $V_B$, we have $|\Gamma(A)| \geq \frac{1}{\Delta}|A|$, then there exists a $\Delta$-matching.*

**Lemma 8** *If $M$ is a $\Delta$-matching in $H$, then $w(M) \leq \Delta \cdot d^*(s)$, where $w(M)$ is the sum of edge lengths in $M$.*

**Lemma 9** *Given a weighted undirected graph $G = (V, E)$, let $T_1, ...T_m$ be some node-disjoint subtrees of $MST(G)$. Let $V' = \cup_i V(T_i)$ and $G'$ be the restriction of $G$ on $V'$. Then $MST(G') \geq \sum_i |T_i|$.*

**Lemma 10** *Suppose $s \geq s_{min}$, then for any set of blocks $A = \{T_1, ...T_k\} \subset \mathcal{T}$, we have $|\Gamma(A)| \geq \lambda |A|$, where $\lambda = \frac{C_1}{\alpha_{st}(C_2+3)}$.*

**Theorem 11** *The running time of the above algorithm is* $\min\{O(n^{2.5}), O(n^2 L)\}$.

We remark that the dominating term in the running time is time spent on bipartite max-matching.

### 3.3 Dyadic Time Windows in a Metric Space

Next we consider more general dyadic time windows. For any set of intervals $\mathcal{I}$ with integer endpoints and maximum length $L$, we can bucket them into $\log L$ classes. Define the *diversity parameter* $\gamma$ as the number of nonempty buckets. For example, if $\mathcal{I}$ is a set of unit-length time windows, then $\gamma = 1$. Notice that $\gamma$ is at most $\log L$.

**Theorem 12** *For inputs with diversity parameter $\gamma$, and having at least one site with time window $[i, i+1]$ for each integer $i \leq L$, there is an $(O(\gamma), O(\gamma))$-approximation for TWTSP.*

## 4 Other Variants

In the full paper, we also show how to use our method to give positive results for some other routing problems with time windows. These problems include (1) multi-robot TWTSP with some given information (For each robot $j$, and each $i \leq L$, there is a site $v_{i,j}$ with time window $[i, i+1]$ that robot $j$ is required to visit); and (2) special cases for TWTSPN.

## References

[1] N. Bansal, A. Blum, S. Chawla, and A. Meyerson. Approximation algorithms for deadline-tsp and vehicle routing with time-windows. In *Proceedings of the Thirty-sixth Annual ACM Symposium on Theory of Computing*, STOC '04, pages 166–174, 2004.

# Generalized Coverage in Homological Sensor Networks*

Nicholas J. Cavanna† Kirk Gardner‡

Donald R. Sheehy§

## 1 Introduction

In their seminal work on homological sensor networks, de Silva and Ghrist showed the surprising fact that it is possible to certify the coverage of a coordinate-free sensor network even with very minimal knowledge of the space to be covered [5]. We give a new, simpler proof of the de Silva-Ghrist *Topological Coverage Criterion (TCC)* that eliminates any assumptions about the smoothness of the boundary of the underlying space, allowing the results to be applied to much more general problems. The new proof factors the geometric, topological, and combinatorial aspects of this approach. This allows us to extend the TCC to support $k$-coverage, in which the domain is covered by $k$ sensors, and weighted coverage, in which sensors have varying radii.

## 2 Background

**Distances and Offsets.** For a set $X$, let $\mathcal{P}(X)$ denote the power set of $X$ and let $\binom{X}{k}$ denote the set of $k$-element subsets of $X$. Given a compact point set $A \subset \mathbb{R}^d$ with weights $w_a \geq 0$ for all $a \in A$ the **weighted distance** from a point $x$ to a weighted point $y$ is defined as the power distance

$$\rho_y(x)^2 := \|x - y\|^2 + w_y^2.$$

Such a set is referred to as a **weighted set**. We use weighted distances to model coverage by disks of varying radii, where larger weights correspond to smaller radii.

The **weighted $k$-nearest neighbor distance** from a point $x$ to $k$ points in a weighted compact set $A$ is defined as

$$\mathrm{d}_k(x, A)^2 := \inf_{K \in \binom{A}{k}} \max_{y \in K} \rho_y(x).$$

Note that if $w_a = 0$ for all $a \in A$ and $k = 1$ then

$$\mathrm{d}_1(x, A) = \mathrm{d}(x, A) := \min_{y \in A} \|x - y\|.$$

Such as set is said to be **unweighted**.

The **canonical offsets** of a set $A$ at a scale $\varepsilon$ are defined as

$$A^\varepsilon := \{x \in X \mid \mathrm{d}(x, A) \leq \varepsilon\}.$$

We use the word "canonical" to distinguish these offsets from the **weighted $(k, \varepsilon)$-offsets** of a weighted compact set $A$, defined to be

$$A_k^\varepsilon := \{x \in X \mid d_k(x, A) \leq \varepsilon\}.$$

If $A$ is unweighted we obtain the **$(k, \varepsilon)$-offsets**, the points within $\varepsilon$ of $k$ points in $A$. Note that for any weighted set $A$ we have $A_k^\varepsilon \subseteq A^\varepsilon$. Thus, $\varepsilon$ provides an upper bound on the radii.

**Čech and Rips Complexes.** The weighted **Čech complex** of a finite collection of points $A$ in $\mathbb{R}^d$ at scale $\varepsilon$ is defined as

$$\check{\mathrm{C}}\mathrm{ech}_\varepsilon(A) := \left\{\sigma \subseteq A \mid \exists x \in \mathbb{R}^d : \max_{p \in \sigma} \rho_p(x) \leq \varepsilon\right\}.$$

The **(Vietoris-)Rips complex** of $A$ at scale $\varepsilon$ is defined

$$\mathrm{Rips}_\varepsilon(A) := \left\{\sigma \subseteq A \mid \{p, q\} \in \check{\mathrm{C}}\mathrm{ech}_\varepsilon(A) \text{ for all } p, q \in \sigma\right\}.$$

The standard Rips and Čech complexes are obtained setting $w_p = 0$ for all $a \in A$.

An important result about the relationship of Čech and Rips complexes follows from Jung's Theorem [7] relating the diameter of a point set $A$ and the radius of the minimum enclosing ball:

$$\check{\mathrm{C}}\mathrm{ech}_\varepsilon(A) \subseteq \mathrm{Rips}_\varepsilon(A) \subseteq \check{\mathrm{C}}\mathrm{ech}_{\vartheta_d \varepsilon}(A), \qquad (1)$$

where the constant $\vartheta_d = \sqrt{\frac{2d}{d+1}}$ for unweighted sets and $\vartheta_d = 2$ for weighted sets (see [1]).

**The $k$-Barycentric Decomposition.** Given a simplicial complex $S$ we define a **flag** in $S$ to be an ordered subset of simplices $\{\sigma_1, \ldots, \sigma_t\} \subset S$ such that $\sigma_1 \subset \ldots \subset \sigma_t$. The **barycentric decomposition** of $S$ is the simplicial complex formed by the set of flags of $S$ and is defined as $\mathrm{Bary}(S) := \{U \subset S \mid U \text{ is a flag of } S\}$. The vertices of the barycentric decomposition are the simplices of $S$. We define the **degree** of a flag $\sigma_1 \subset \cdots \subset \sigma_t$ to be $|\sigma_1|$.

**Definition 1** *The **$k$-barycentric decomposition** of a complex $S$ is defined*

$$k\text{-Bary}(S) := \{U \subset S \mid U \text{ is a flag in } S \text{ of degree at least } k\}.$$

*This work was partially supported under grants CCF-1464379 and CCF-1525978.

†University of Connecticut

‡University of Connecticut

§University of Connecticut

The $k$-barycentric decomposition of the Čech and Rips complexes of a finite point set $A$ at a scale $\varepsilon$ are denoted $\check{\text{C}}\text{ech}^k_\varepsilon(A)$ and $\text{Rips}^k_\varepsilon(A)$ respectively. By [8], we have the following results relating the $k$-barycentric decomposition of Čech and Rips complexes to the $(k,\varepsilon)$-offsets of their vertex set $A$.

**Theorem 1** *Given a finite point set $A$, fixed $k$ and any $\varepsilon \geq 0$ the $k$-barycentric decomposition of the Čech complex $\check{\text{C}}\text{ech}^k_\varepsilon(A)$ is homotopy equivalent to the $(k,\varepsilon)$-offsets $A^\varepsilon_k$.*

**Theorem 2** *The $k$-barycentric decomposition of the Rips complex $\text{Rips}^k_\varepsilon(A)$ is a $\vartheta_d$-approximation to the $(k,\varepsilon)$-offsets $A^\varepsilon_k$.*

This result allows us to extend Equation 1 to the $k$-barycentric decomposition of the Čech and Rips complexes as follows:

$$\check{\text{C}}\text{ech}^k_\varepsilon(A) \subseteq \text{Rips}^k_\varepsilon(A) \subseteq \check{\text{C}}\text{ech}^k_{\vartheta_d\varepsilon}(A), \tag{2}$$

**Homology and Persistent Homology.** Homology is a tool from algebraic topology that gives a computable signature for a shape that is invariant under many kinds of topological equivalences. It gives a way to quantify the components, loops, and voids in a topological space. It is a favored tool for applications because its computation can be phrased as a matrix reduction problem with matrices representing a finite simplicial complex.

Throughout, we assume singular homology over a field, so the $n$th homology group $\text{H}_n(C)$ of a space $C$ is vector space. When considering the homology groups of all dimensions, we will write $\text{H}_*(C)$. We will make extensive use of relative homology. That is, for a pair of spaces $(A,B)$ with $B \subseteq A$, we write $\text{H}_*(A,B)$ for the homology of $A$ relative to $B$.

There are dual vector spaces to the homology groups called the **cohomology groups** and are denoted with superscripts as $\text{H}^*(C)$. For finite-dimensional homology groups the **Alexander duality** [6] implies that for pairs of nonempty locally-contractible spaces in $\mathbb{R}^d \cup \{\infty\}$, the $r$-dimensional homology is isomorphic to the $(d-r)$-dimensional cohomology of the complement space, i.e.

$$\text{H}_r(X,Y) \cong \text{H}^{d-r}(\overline{Y},\overline{X}).$$

## 3 Geometric Assumptions

Strange examples abound in topology. One must make some assumptions about the underlying domain to make the TCC possible. In this section, we will introduce and illustrate the minimal geometric properties that we require of the bounded domain to be covered. Our goal is to weaken the geometric assumptions on the domain required for the topological coverage criterion of de Silva & Ghrist to apply it to domains without smooth manifold boundaries. In particular, for a domain $\mathcal{D}$ with boundary $\mathcal{B}$ satisfying the following conditions for $0 \leq 3\alpha \leq \beta$, we want to certify that a sample $P \subset \mathcal{D}$ covers $\mathcal{D}$ at scale $\alpha$ in the sense that $\mathcal{D} \setminus \mathcal{B}^{2\alpha} \subset P^\alpha$.

**Assumptions**

1. *(Non-degenerate)* $\mathcal{D}$ is compact, locally contractible, full dimensional in $\mathbb{R}^d$ and there exists $\varepsilon > 0$ such that $\mathcal{D} \hookrightarrow \mathcal{D}^\varepsilon$ induces a homotopy equivalence.
2. *(Components are not too small)* The map $\text{H}_0(\mathcal{D} \setminus \mathcal{B}^{\alpha+\beta} \hookrightarrow \mathcal{D} \setminus \mathcal{B}^{2\alpha})$ is surjective.
3. *(Components are not too close)* The map $\text{H}_0(\mathcal{D} \setminus \mathcal{B}^{2\alpha} \hookrightarrow D^{2\alpha})$ is injective.

Assumption 1 disallows degenerate cases in which some of the theorems listed in Section 2 cannot be applied. For example, the Alexander duality as we have stated it requires distinct, bounded, locally contractible spaces.

Assumptions 2 and 3 prevent components from appearing and disappearing in the inclusions $\mathcal{D} \setminus \mathcal{B}^{\alpha+\beta} \hookrightarrow \mathcal{D} \setminus \mathcal{B}^{2\alpha}$ and $\mathcal{D} \setminus \mathcal{B}^{2\alpha} \hookrightarrow D^{2\alpha}$, respectively. These restrictions are in order to allow us to reliably compare the coverage region to the sampled subset of a disconnected domain in terms of the 0-dimensional homology, or connected components of related spaces. Specifically, Assumption 2 disallows domains with components that are too small to be included in the map from $\mathcal{D} \setminus \mathcal{B}^{\alpha+\beta} \hookrightarrow \mathcal{D} \setminus \mathcal{B}^{2\alpha}$. Fig. 1 illustrates a domain in which the induced map is not surjective, allowing our algorithm to potentially report a false positive.

Assumption 3 requires that the components of $\mathcal{D} \setminus \mathcal{B}^{2\alpha}$ are spaced out well enough so that no components are joined with inclusion into $\mathcal{D}^{2\alpha}$. This is required in order to be able to reliably bound the number of connected components of the shrunken domain by those of a computable combinatorial structure. Fig. 2 illustrates domains which violates Assumption 3 as components are lost in both $\mathcal{D} \setminus \mathcal{B}^{2\alpha} \hookrightarrow \mathcal{D}$ and $\mathcal{D} \hookrightarrow \mathcal{D}^{2\alpha}$.

**Relationship to Geometric Assumptions in Prior Work** Previous work on the TCC by de Silva & Ghrist is restricted to connected domains with a smooth boundary in which a particular region around the domain has uniform thickness. This region is parameterized for smooth manifolds in the work by the injectivity radius, which serves to bound the region around the boundary in which no topological changes occur. The injectivity radius is closely related to the **reach** of a compact set $K$ in $\mathbb{R}^d$, defined as the supremum of the distance $r$ from $K$ to any point $x$ with a unique closest point $y \in K$

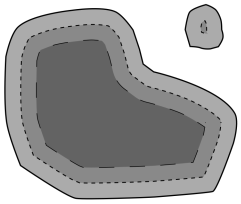


Figure 1: A domain that violates assumption 2 in which $\mathrm{H}_0(\mathcal{D} \setminus \mathcal{B}^{\alpha+\beta} \hookrightarrow \mathcal{D} \setminus \mathcal{B}^{2\alpha})$ is not surjective as the upper-right component is pinched out in $\mathcal{D} \setminus \mathcal{B}^{\alpha+\beta}$.

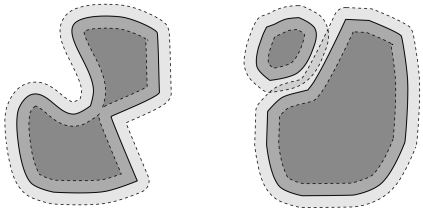


Figure 2: A domain that violates Assumption 3 in which $\mathrm{H}_0(\mathcal{D} \setminus \mathcal{B}^{2\alpha} \hookrightarrow D^{2\alpha})$ is not injective as components are lost in both the inclusions $\mathcal{D} \setminus \mathcal{B}^{2\alpha} \hookrightarrow D$ and $\mathcal{D} \hookrightarrow D^{2\alpha}$.

such that $d(x, y) = d(x, K)$. For non-smooth compact sets $K$ containing sharp corners, for example, note that the reach of $K$ will be equal to zero, as for all $r > 0$ there must exist some $x \in \mathbb{R}^d \setminus K$ with at least two closest points in $K$ a distance $r$ from $x$ approaching the sharp corner of the set.

This notion of feature size approaching a non-smooth feature is generalized in [2] as the **$\mu$-reach**. Roughly speaking, the $\mu$-reach parameterizes the reach in order to provide a meaningful measure of the region around a potentially non-smooth compact subset $K$ in which no topological changes occur. In particular, the $\mu$-reach is equal to the reach for $\mu = 1$, and converges to the minimum distance from $K$ to the critical points of the distance function $d(\cdot, K)$ as $\mu$ approaches 0. This minimum distance is known as the **weak feature size** of $\mathcal{B}$ and was introduced in [3] as a way to parameterize compact sets that may not be smooth manifolds. For our purposes it can be understood as the minimum size of the topological features of a compact set. Thus, for a compact subset $K$ of $\mathbb{R}^d$ we have that $\mathrm{reach}(K) \leq \mathrm{reach}_\mu(K) \leq \mathrm{wfs}(K)$.

We note that any bounded domain $\mathcal{D}$ such as that in Fig. 1 with a component in $\mathcal{D} \setminus \mathcal{B}^{2\alpha}$ in which the distance from every point in the component is at most $\alpha + \beta$ has a boundary with reach at most $\alpha + \beta$. Moreover, in the figures of 2 there exist points in $\mathcal{D} \setminus \mathcal{B}^{2\alpha}$ and $\mathcal{D}$ contained in distinct components within distance $2\alpha$ of each other, respectively. It follows that the minimum distance to a critical point of $\mathcal{B}$, a point which lies in the convex hull of its nearest neighbors in $\mathcal{B}$, is at most $2\alpha$. Conversely, for any $\mathcal{D}$ such that the reach of $\mathcal{B}$ is strictly greater than $\alpha + \beta$ Assumption 2 is implied. Assumption 3 is implied for any bounded domain such that the weak feature size of its boundary is strictly greater than $2\alpha$. As $\mathrm{reach}(K) \leq \mathrm{wfs}(K)$ we therefore maintain our geometric assumptions for any domain $\mathcal{D}$ with a boundary $\mathcal{B}$ such that $\mathrm{wfs}(\mathcal{B}) > \alpha + \beta \geq 4\alpha$.

**Relationship to the de Silva & Ghrist TCC** For the sake of contrast, we will state the Topological Coverage Criterion as states by de Silva & Ghrist in [5]. Here we will assume points in a fixed point set $P$ in a domain $\mathcal{D} \subset \mathbb{R}^d$ with boundary $\mathcal{B}$ have have uniform coverage radius $r_c$, fence detection radius (in which nodes can detect the boundary) $r_f$, and node-detection radii $r_s \leq \sqrt{2} r_c$, $r_w \geq r_s \sqrt{10}$. Moreover, we will let $Q = \{x \in P \mid d(x, \mathcal{B}) \leq r_f\}$ be the subset of $P$ consisting of points sufficiently close to the boundary $\mathcal{B}$.

**Theorem 3 (TCC (de Silva & Ghrist))** *Let $P$ be a fixed set of nodes in a domain $\mathcal{D} \subset \mathbb{R}^d$ with boundary $\mathcal{B}$ such that each $p \in P$ the restricted domain $\mathcal{D} \setminus \mathcal{B}^{r_f + r_s/\sqrt{2}}$ is connected and the hypersurface $\Sigma = \{x \in \mathcal{D} \mid d(x, \mathcal{B}) = r_f\}$ has internal injectivity radius at least $r_s/\sqrt{2}$ and external injectivity radius at least $r_s$. The sensor cover $P^{r_c}$ contains $\mathcal{D} \setminus \mathcal{B}^{r_f + r_s/\sqrt{2}}$ if the homomorphism*

$$\iota_* : H_d(\mathrm{Rips}_{r_s}(P), \mathrm{Rips}_{r_s}(Q)) \to H_d(\mathrm{Rips}_{r_w}(P), \mathrm{Rips}_{r_w}(Q))$$

*induced by the inclusion $\iota : \mathrm{Rips}_{r_s}(P) \hookrightarrow \mathrm{Rips}_{r_w}(P)$ is nonzero.*

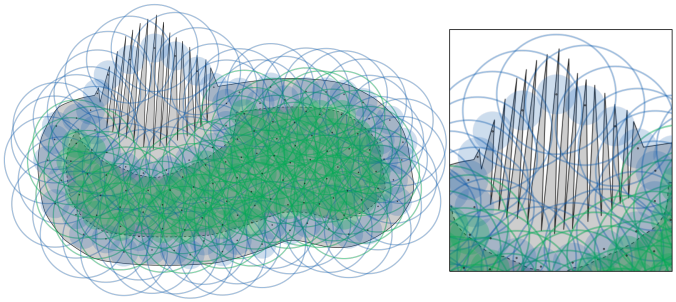


Figure 3: This instance illustrates the failure of Lemma 3.3 of [5] when the boundary is not smooth. A cycle that is trivial in the thickened boundary persists. This highlights the need to work with the relative homology of the domain modulo the boundary rather than the homology of the boundary alone. Such a cycle in the boundary cannot form a relative cycle.

According to [5, Remark 4.5], the smooth manifold hypothesis is a necessary requirement in order to ap-

ply the TCC without degenerating constants.[1] Because their analysis involves directly comparing this thickened region around boundary to this complex it was necessary to show that the thickness of this region is such that any topological noise in the complex is eliminated with inclusion from scale $\alpha$ to $\beta$. This amounts to proving that that cycles lying entirely in a thickening of the boundary of the domain cannot persist in the TCC, as in [5, Lemma 3.3]. Such a case is shown in Fig. 3, illustrating a domain without a smooth boundary in which the thickened boundary contains a cycle that persists across a range of scales. This is to be contrasted with our Lemma 5, requiring Assumptions 1 and 2, in which the persistence of this cycle does not indicate the persistence of a relative cycle. For example, although the domain in Fig. 3 clearly does not have a smooth boundary it does have weak feature size greater than $\alpha + \beta$, and therefore satisfies our geometric conditions. It is in this sense that Assumptions 2 and 3 serve to weaken the smoothness hypothesis in order to allow the TCC to be applied to domains, such as those with bounded weak feature size, which imply our minimal hypothesis.

## 4 The Generalized Topological Coverage Criterion

Consider a domain $\mathcal{D} \subset \mathbb{R}^d$, its boundary $\mathcal{B}$, and constants $\alpha$ and $\beta$ such that $0 \leq 3\alpha \leq \beta$ satisfying Assumptions 1–3. Given a weighted finite point set $P \subseteq \mathcal{D}$ we will give a sufficient condition to guarantee the $k$-coverage of a shrunken domain $\mathcal{D} \setminus \mathcal{B}^{2\alpha}$, where the $k$-covered region corresponds to the weighted offsets $P_k^\alpha$ as defined by $\mathrm{d}_k$. We define $Q := P \cap \mathcal{B}^\alpha$, i.e. the subsample of $P$ that is within distance $\alpha$ of the boundary.

Lemma 4 allows us to talk about the homology of the shrunken boundary in terms of relative homology where $\overline{A} := (\mathbb{R}^d \cup \infty) \setminus A$ denotes the **complement** of $A$ in the compactification of $\mathbb{R}^d$ homeomorphic to the $d$-sphere $S^d$.

**Lemma 4** *For all $\varepsilon > 0$, $\mathrm{H}_0(\mathcal{D} \setminus \mathcal{B}^\varepsilon) \cong \mathrm{H}_0(\overline{\mathcal{B}^\varepsilon}, \overline{\mathcal{D}^\varepsilon})$.*

**Proof.** Consider the inclusion $(\mathcal{D} \setminus \mathcal{B}^\varepsilon, \emptyset) \hookrightarrow (\overline{\mathcal{B}^\varepsilon}, \overline{\mathcal{D}^\varepsilon})$ and the corresponding map $\mathrm{H}_0((\mathcal{D} \setminus \mathcal{B}^\varepsilon, \emptyset) \hookrightarrow (\overline{\mathcal{B}^\varepsilon}, \overline{\mathcal{D}^\varepsilon}))$. For injectivity, given some non-trivial 0-chain $[x] \in \mathrm{H}_0(\mathcal{D} \setminus \mathcal{B}^\varepsilon)$, we can pick a representative point $x \in \mathcal{D} \setminus \mathcal{B}^\varepsilon \subseteq \overline{\mathcal{B}^\varepsilon}$. Given that $\mathcal{B}$ is the boundary of $\mathcal{D}$, a dimension-$n$ space, then there exists no paths from $\mathcal{D} \setminus \mathcal{B}^\varepsilon$ to $\overline{\mathcal{B}}$, so $[x] \neq 0 \in \mathrm{H}_0(\overline{\mathcal{B}^\varepsilon}, \overline{\mathcal{D}^\varepsilon})$. For surjectivity, given some $[x] \in \mathrm{H}_0(\overline{\mathcal{B}^\varepsilon}, \overline{\mathcal{D}^\varepsilon})$, it represents a point on a connected component on $\overline{\mathcal{B}^\varepsilon} \setminus \overline{\mathcal{D}^\varepsilon} = \mathcal{D} \setminus \mathcal{B}^\varepsilon$, and thus a homology class $[x] \in \mathrm{H}_0(\mathcal{D} \setminus \mathcal{B}^\varepsilon)$. □

[1] [5, Remark 4.5] states that, although domains with a polygonal boundary are admissible in practice, the constant $r_w$ would blow up along with the angle of the sharpest corner of the outermost boundary component.

We will assume non-negative weights $w_x \geq 0$ assigned to each $x \in P$, and that $w_x = 0$ for all points $x \in \mathcal{D} \setminus P$. This implies that $\mathcal{D}_k^\varepsilon = \mathcal{D}^\varepsilon$, and similarly $\mathcal{B}_k^\varepsilon = \mathcal{D}^\varepsilon$, so we will simply use the notation $\mathcal{D}^\varepsilon$ and $\mathcal{B}^\varepsilon$ throughout. Moreover, we know that $P_k^\alpha \subseteq \mathcal{D}_k^\alpha = \mathcal{D}^\alpha$ by the monotonicity of $\mathrm{d}_k$. For any arbitrary weighted compact set $A \subseteq \mathcal{D}$, $A_k^\varepsilon \subseteq A_1^\varepsilon \subseteq A^\varepsilon$ and $Q \subseteq B^\alpha$, for $\varepsilon \geq 0$, $Q_k^\varepsilon \subseteq Q^\varepsilon \subseteq B^{\alpha+\varepsilon}$.

Diagram (3) relates the connected components of the pairs, and induces the map $\pi_* : \mathrm{im}\ j_* \to \mathrm{im}\ i_*$.

$$\begin{array}{ccc} \mathrm{H}_0(\overline{\mathcal{B}^{\alpha+\beta}}, \overline{\mathcal{D}^{\alpha+\beta}}) & \xrightarrow{j_*} & \mathrm{H}_0(\overline{\mathcal{B}^{2\alpha}}, \overline{\mathcal{D}^{2\alpha}}) \\ \downarrow & & \downarrow \\ \mathrm{H}_0(\overline{Q_k^\beta}, \overline{P_k^\beta}) & \xrightarrow{i_*} & \mathrm{H}_0(\overline{Q_k^\alpha}, \overline{P_k^\alpha}) \end{array} \tag{3}$$

Though reversed and inverted by the dualities, this map describes the topology of the offsets embedded into the domain, where the scale change eliminates noise. That is, it captures exactly the topological information we want. Analyzing $\pi_*$ directly simplifies the proof and aids in eliminating some hypotheses.

The following two lemmas prove two important properties of $\pi_*$. These will be used to give a computable way to infer coverage from the rank of $i_*$.

**Lemma 5** *Given Assumptions 1 and 2, the map $\pi_*$ is surjective.*

**Proof.** Assumption 2 implies that $j_*$ is surjective by Alexander Duality. We choose a basis for $\mathrm{im}\ i_*$ such that each basis element is a point in $P_k^\alpha \setminus Q_k^\beta$. Consider $x \in P_k^\alpha \setminus Q_k^\beta$ such that $[x] \neq 0 \in \mathrm{im}\ i_*$. If $x \in \overline{\mathcal{B}^{2\alpha}}$, then $x \in \mathcal{D} \setminus \mathcal{B}^{2\alpha}$ so $[x] \neq 0 \in \mathrm{H}_0(\overline{\mathcal{B}^{2\alpha}}, \overline{\mathcal{D}^{2\alpha}})$. Because $j_*$ is surjective, $\mathrm{H}_0(\overline{\mathcal{B}^{2\alpha}}, \overline{\mathcal{D}^{2\alpha}}) = \mathrm{im}\ j_*$ and thus $\pi_*([x]) = [x]$ and so $[x] \in \mathrm{im}\ \pi_*$.

If $x \in \mathcal{B}^{2\alpha}$, then there is a point $y \in \mathcal{B}$ such that $\|x - y\| \leq 2\alpha$. Because $x \in \overline{Q_k^\beta}$ by hypothesis, $\mathrm{d}_k(x, Q) > \beta$. We will show that if a point $z$ is in the line segment $\overline{xy}$, then $z \in \overline{Q_k^\alpha}$. For any $z \in \overline{xy}$, we have $\|x - z\| \leq \|x - y\| \leq 2\alpha$. So,

$$\begin{aligned} \mathrm{d}_k(z, Q) &\geq \mathrm{d}_k(x, Q) - \|x - z\| && [\mathrm{d}_k \text{ is Lipschitz}] \\ &> \beta - 2\alpha && [\mathrm{d}_k(x, Q) > \beta \text{ and } \|x - z\| \leq 2\alpha] \\ &\geq \alpha && [\beta \geq 3\alpha] \end{aligned}$$

So, we conclude that $z \in \overline{Q_k^\alpha}$, and thus $\overline{xy} \subseteq \overline{Q_k^\alpha}$.

The definition of $Q$ implies that $\mathcal{B} \cap \overline{Q_k^\alpha} \subseteq \overline{P_k^\alpha}$, and so $y \in \overline{P_k^\alpha}$. Any path $\gamma : [0, 1] \to \overline{Q_k^\alpha}$ such that $\gamma(0) = x$ and $\gamma(1) = y$, generates a class $[\gamma]$ in the chain group $C_1(\overline{Q_k^\alpha})$ containing $\gamma$. For $[\gamma] \in C_1(\overline{Q^\alpha}_k, \overline{P_k^\alpha})$ it follows $\partial([\gamma]) = [x + y] = [x]$ as $y \in \overline{P_k^\alpha}$, and therefore that there must exist $z \in \overline{xy} \cap Q_k^\alpha$. This is a contradiction as we have shown that $\overline{xy} \cap Q_k^\alpha = \emptyset$, and thus we conclude $x$ cannot be in $\mathcal{B}^{2\alpha}$. □

The following lemma therefore allows us to confirm coverage by comparing the ranks of im $i_*$ and im $j_*$.

**Lemma 6** *Given Assumption 1, if $\pi_*$ is injective then $\mathcal{D} \setminus \mathcal{B}^{2\alpha} \subseteq P_k^\alpha$.*

**Proof.** The proof is essentially the same as that presented by de Silva & Ghrist [5]. We include it here in our own notation for completeness.

We will prove this by contradiction. Assume there exists $x \in (\mathcal{D} \setminus \mathcal{B}^{2\alpha}) \setminus P_k^\alpha$,and thus $[x] \neq 0 \in \mathrm{H}_0(\overline{\mathcal{B}^{2\alpha}}, \overline{\mathcal{D}^{2\alpha}})$. This is true because as we know that $x$ is in the interior of $\mathcal{D}$, so it is on a connected component of $\mathcal{D} \setminus \mathcal{B}^{2\alpha}$. Consider the following sequence:

$$\mathrm{H}_0(\overline{\mathcal{B}^{2\alpha}}, \overline{\mathcal{D}^{2\alpha}}) \xrightarrow{f_*} \mathrm{H}_0(\overline{\mathcal{B}^{2\alpha}}, \overline{\mathcal{D}^{2\alpha}} \cup \{x\}) \xrightarrow{g_*} \mathrm{H}_0(\overline{Q_k^\alpha}, \overline{P_k^\alpha})$$

As $f_*([x]) = 0 \in \mathrm{H}_0(\overline{\mathcal{B}^{2\alpha}}, \overline{\mathcal{D}^{2\alpha}} \cup \{x\})$, then $(g_* \circ f_*)([x]) = 0$. But we have a contradiction as $g_* \circ f_* = \pi_*$, and $\pi_*([x]) \neq 0$ by injectivity, so $\mathcal{D} \setminus \mathcal{B}^{2\alpha} \subseteq P_k^\alpha$. □

As Lemma 5 asserts that $\pi_*$ is surjective under our assumptions, Lemma 6 can therefore be used to confirm coverage by providing conditions in which $\pi_*$ is injective. Thus, the following theorem provides sufficient conditions to confirm $\mathcal{D} \setminus \mathcal{B}^{2\alpha} \subseteq P_k^\alpha$. Note that it will not yet give us an algorithm (that will come in Theorem 10), but instead gives a result about the offsets directly rather than an embedding of a Rips complex as was used in previous work.

**Theorem 7 (Geometric TCC)** *Consider $\mathcal{D} \subset \mathbb{R}^d$ with boundary $\mathcal{B}$ satisfying Assumptions 1 and 2. Let $\alpha$ and $\beta$ be constants such that $0 < 3\alpha \leq \beta$. Let $P \subset \mathcal{D}$ be a finite set with $Q = P \cap \mathcal{B}^\alpha$. Let $i_*$ and $j_*$ be the maps in Diagram (3). If* rk $i_* \geq$ rk $j_*$ *then $\mathcal{D} \setminus \mathcal{B}^{2\alpha} \subseteq P_k^\alpha$.*

**Proof.** Given Assumptions 1 and 2, Lemma 5 implies that $\pi_* :$ im $j_* \to$ im $i_*$ is surjective, and so rk $i_* \leq$ rk $j_*$. By hypothesis, rk $i_* \geq$ rk $j_*$, so it follows that rk $i_* =$ rk $j_*$. Because both the images are finite-dimensional, $\pi_*$ is an isomorphism, and therefore it is injective. Lemma 6 then implies $\mathcal{D} \setminus \mathcal{B}^{2\alpha} \subseteq P_k^\alpha$. □

## 5 Computing the TCC

In the previous section we prove sufficient conditions for generalized coverage in terms of the offsets of the input points. However, we may not be able to compute these offsets, because we do not know the positions of the points in $P$. Instead, we use Rips complexes in the algorithm.

Let $\mathrm{Rips}_\beta(X)$ and $\check{\mathrm{C}}\mathrm{ech}_\beta(X)$ denote respectively the Rips and Čech complexes of a set $X$ at scale $\beta$. Let $R_\beta^k$ be the pair of k-barycentric Rips complexes $(\mathrm{Rips}_\beta^k(P), \mathrm{Rips}_\beta^k(Q))$ and let $C_\beta^k$ be the pair of k-barycentric Čech complexes $(\check{\mathrm{C}}\mathrm{ech}_\beta^k(P), \check{\mathrm{C}}\mathrm{ech}_\beta^k(Q))$ as defined in Section 2. If $k = 1$ and $P$ is unweighted we define the standard Rips and Čech complex pairs $R_\beta^1 := (\mathrm{Rips}_\beta(P), \mathrm{Rips}_\beta(Q))$ and $C_\beta^1 := (\check{\mathrm{C}}\mathrm{ech}_\beta(P), \check{\mathrm{C}}\mathrm{ech}_\beta(Q))$.

Algorithm 1 is for checking $k$-coverage of the shrunken domain by a weighted sample $P$, i.e. that $\mathcal{D} \setminus \mathcal{B}^{2\alpha} \subseteq P_k^\alpha$. The algorithm requires that the point samples each of the connected components of $\mathcal{D} \setminus \mathcal{B}^{2\alpha}$. It first constructs three Rips complexes based on the input parameters $(\alpha, \beta, P, Q, k)$: $\mathrm{Rips}_\alpha(P)$, $R_{\alpha/\vartheta_d}^k$ and $R_\beta^k$. It then checks a condition relating the homology of the complexes, and if it satisfied, $k$-coverage is guaranteed. Note that if the algorithm's output is false it does not necessarily mean there is not coverage. Lemma 8, Lemma 9 and Theorem 10 together provide a proof of correctness of Algorithm 1. Lemma 8 bounds the rank of the map between the Rips complexes at different scales by rk $i_*$, in order to compare it to rank $j_*$ through Theorem 7. Lemma 9 states that if the components are separated enough, formally defined in Assumption 3, then the number of connected components of the Rips complex at scale $\alpha$ of $P$ provides an upper bound for the number of components of $\mathcal{D} \setminus \mathcal{B}^{2\alpha}$.

**Algorithm 1** Check if $\mathcal{D} \setminus B^{2\alpha} \subseteq P_k^\alpha$

```
procedure k-Coverage(α, β, P, Q, k)
    construct Rips_α(P)
    let c := dim H_0(Rips_α(P))
    construct R^k_{α/ϑ_d} and R^k_β
    let r := rk H_d(R^k_{α/ϑ_d} ↪ R^k_β)
    if c = r then return True
    else return False
```

**Lemma 8** *The rank of the map $\mathrm{H}_d(R_{\alpha/\vartheta_d}^k \hookrightarrow R_\beta^k)$ induced by inclusion is at most* rk $i_*$.

**Proof.** For the case of $k = 1$, the Persistent Nerve Lemma [4] says that for $\varepsilon \geq 0$, $\mathrm{H}_*(C_\varepsilon^1) \cong \mathrm{H}_*(P_1^\varepsilon, Q_1^\varepsilon)$. The Universal Coefficient Theorem with respect to Diagram (3) implies that $\mathrm{rk}(\mathrm{H}_d(C_\alpha^1 \hookrightarrow C_\beta^1)) =$ rk $i_*$. Moreover, the inclusion $R_{\alpha/\vartheta_d}^1 \hookrightarrow R_\beta^1$ can be factored as

$$R_{\alpha/\vartheta_d}^1 \hookrightarrow C_\alpha^1 \hookrightarrow C_\beta^1 \hookrightarrow R_\beta^1.$$

It follows that

$$\mathrm{rk}(\mathrm{H}_d(R_{\alpha/\vartheta_d}^1) \to \mathrm{H}_d(R_\beta^1)) \leq \mathrm{rk}(\mathrm{H}_d(C_\alpha^1) \to \mathrm{H}_d(C_\beta^1)) = \mathrm{rk}\, i_*.$$

For $k \geq 2$, Theorem 2 states that $(\mathrm{Rips}_\varepsilon^k(P), \mathrm{Rips}_\varepsilon^k(Q))$ is a $\vartheta_d$-approximation to $(P_k^\varepsilon, Q_k^\varepsilon)$. This implies that $\mathrm{H}_*(R_{\varepsilon/\vartheta_d}^k) \cong \mathrm{H}_*(P_k^\varepsilon, Q_k^\varepsilon)$, so the previous argument naturally follows for these cases as well. □

**Lemma 9** *Given $P$ has at least one point on each connected component of $\mathcal{D} \setminus \mathcal{B}^{2\alpha}$, if Assumptions 1 and 3 are satisfied then the number of connected components of $\mathrm{Rips}_\alpha(P)$ is greater than or equal to the number of connected components of $\mathcal{D} \setminus \mathcal{B}^{2\alpha}$.*

**Proof.** Assume there exists $p, q \in P$ such that $p$ and $q$ are connected in $\mathrm{Rips}_\alpha(P)$, but not in $\mathcal{D} \setminus \mathcal{B}^{2\alpha}$. This implies that $\|p - q\| \leq 2\alpha$ and $[p] \neq [q]$ in $\mathrm{H}_0(\mathcal{D} \setminus \mathcal{B}^{2\alpha})$. However, $\overline{pq} \in \mathcal{D}^{2\alpha}$ as the distance between $p$ and $q$ is less than $2\alpha$, so $[p] = [q]$ in $\mathrm{H}_0(\mathcal{D}^{2\alpha})$, which implies that $\mathrm{H}_0(\mathcal{D} \setminus \mathcal{B}^{2\alpha} \hookrightarrow \mathcal{D}^{2\alpha})$ is not injective, a contradiction to Assumption 3. □

**Theorem 10 (Algorithmic TCC)** *Consider a domain $\mathcal{D} \subset \mathbb{R}^d$ with boundary $\mathcal{B}$ and constants $\alpha, \beta$, where $0 \leq 3\alpha \leq \beta$, satisfying Assumptions 1, 2 and 3. Let $P \subset \mathcal{D}$ be a finite point sample, $|P| \geq \max\{k, m\}$, where $m = \mathrm{H}_0(D \setminus \mathcal{B}^{2\alpha})$, such that there is a point $p \in P$ in each of the $m$ connected components of $D \setminus \mathcal{B}^{2\alpha}$. If $\mathrm{rk}\ \mathrm{H}_d(R^k_{\alpha/\vartheta_d} \hookrightarrow R^k_\beta) = \dim\ H_0(\mathrm{Rips}_\alpha(P))$, then $\mathcal{D} \setminus \mathcal{B}^{2\alpha} \subseteq P^\alpha_k$.*

**Proof.** For simplicity, define $a_* := \mathrm{H}_d(R^k_{\alpha/\vartheta_d} \hookrightarrow R^k_\beta)$ and set $c = \dim\ \mathrm{H}_0(\mathrm{Rips}_\alpha(P))$. By our hypothesis and Lemma 8, $\mathrm{rk}\ i_* \geq \mathrm{rk}\ a_* = c$. By Lemma 9, $c \geq m$, and Assumption 2 implies that $j_*$ is surjective by Alexander duality, so $m = \mathrm{rk}\ j_*$. Thus $\mathrm{rk}\ i_* \geq \mathrm{rk}\ a_* = c \geq m = \mathrm{rk}\ j_*$, namely $\mathrm{rk}\ i_* \geq \mathrm{rk}\ j_*$, so by Theorem 7 we can conclude $\mathcal{D} \setminus \mathcal{B}^{2\alpha} \subseteq P^\alpha$. □

From this algorithm we can see that, even if we do not know the number of connected components of $D_0$, as long as we know which components have been sampled we can provide a condition to certify coverage of the subdomain that $P$ has been sampled from.

## 6 Conclusion

The TCC gives an effective algorithm for certifying coverage of coordinate-free sensors in an unknown domain. In this paper, we generalized the TCC to certify coverage in spaces whose boundaries may not be smooth. We replaced the smoothness assumption with much weaker conditions, that the domain is non-degenerate in some sense (Assumption 1), that the components are not too small (Assumption 2), and that the components are not to close (Assumption 3).

Although the language of homological sensor networks might imply that the application is restricted to sensors, we hope that the more general geometric conditions provided in this paper will lead to applications in data analysis. Specifically, eliminating the smoothness assumption should make this approach amenable to real data problems.

## References


[1] Mickaël Buchet, Frédéric Chazal, Steve Y. Oudot, and Donald R. Sheehy. Efficient and robust persistent homology for measures. In *ACM-SIAM Symposium on Discrete Algorithms*, pages 168–180, 2015.

[2] Frédéric Chazal, David Cohen-Steiner, and André Lieutier. A sampling theory for compact sets in euclidean space. *Discrete & Computational Geometry*, 41(3):461–479, 2009.

[3] Frédéric Chazal and André Lieutier. Weak feature size and persistent homology: Computing homology of solids in $R^n$ from noisy data samples. In *Proceedings of the 21st ACM Symposium on Computational Geometry*, pages 255–262, 2005.

[4] Frédéric Chazal and Steve Yann Oudot. Towards persistence-based reconstruction in euclidean spaces. In *Proceedings of the Twenty-fourth Annual Symposium on Computational Geometry*, SCG '08, pages 232–241, New York, NY, USA, 2008. ACM.

[5] Vin de Silva and Robert Ghrist. Coverage in sensor networks via persistent homology. *Algebraic & Geometric Topology*, 7:339–358, 2007.

[6] William Julian, Ray Mines, and Alexander. Alexander duality. 106:115–127, 1983.

[7] Heinrich Jung. Über die kleinste kugel, die eine räumliche figur einschließt. *J. Reine Angew. Math*, 123:214–257, 1901.

[8] Donald R. Sheehy. A multicover nerve for geometric inference. In *The Canadian Conference in Computational Geometry*, pages 309–314, 2012.

# Universal Guards: Guarding All Polygonalizations of a Point Set in the Plane

Sándor P. Fekete* Qian Li† Joseph S. B. Mitchell† Christian Scheffer*

## 1 Introduction

The Art Gallery Problem (AGP) seeks to find the fewest guards to see all of a given domain; in its classic combinatorial variant (posed by Victor Klee), it asks for the number of guards that always suffice and are sometimes necessary to guard any simple $n$-gon: the answer is the well known $\lfloor n/3 \rfloor$ [1, 3].

While Klee's question was posed about guarding an $n$-vertex *simple polygon*, a related question about *point sets* was posed at the 2014 Goodman-Pollack Fest (NYU, November 2014): Given a set $S$ of $n$ points in the plane, how many guards always suffice to guard any simple polygon with vertex set $S$? A set of guards that guard every polygonalization of $S$ is said to be a set of *universal guards* for the point set. The question is how many universal guards are always sufficient, and sometimes necessary, for any set of $n$ points? We give the first set of results on universal guarding. We focus here on the case in which guards must be placed at a subset (the *guarded points*) of the input set $S$ and thus will be vertex guards for any polygonalization of $S$.

Due to space limitations, we outline here two selected cases of results: The UGPI (universal guard problem using interior guards), in which guards are placed only at points of $S$ that are not on the convex hull of $S$, and the UGPG (universal guard problem on grids), in which the input set $S$ is a regular grid of points. We then mention results for the general UGP (guards placed at any points of $S$) and cases in which $S$ has a bounded number of convex layers. For the UGPI and UGP, it turns out that a fraction smaller than 1 is not possible: essentially all of the points of $S$ require guards for universal coverage of all polygonalizations of $S$. For the UGPG (on grids) and for cases with bounded convex layers, fractions less than 1 are possible, as we show. Details and further results appear in the paper [2].

*Preliminaries.* We say that three points $a, b, c \in S$ form a *spike* if there exists a subset $S' \subseteq S$ with $a, b, c \in S'$ and a simple polygonal chain, $\pi$, having vertex set $S'$ such that not all of $\triangle abc$ is seen by the points $S \setminus \{a, b, c\}$ when treating $\pi$ as a set of opaque edges. Refer to Figure 1. A point set $S$ is said to be in a *safe configuration with respect to spikes* if no 3 points of $S$ form a spike.

*TU Braunschweig, {fekete,scheffer}@ibr.cs.tu-bs.de
†Stony Brook University, {qian.li.1,joseph.mitchell}@stonybrook.edu

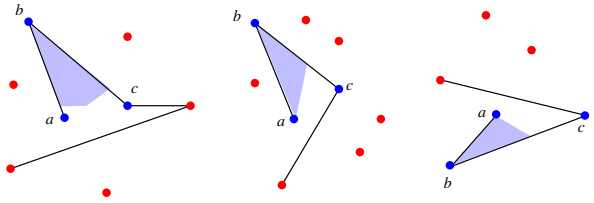



Figure 1: Examples of spikes on $a, b, c$; guards at red points fail to see all of $\triangle abc$.

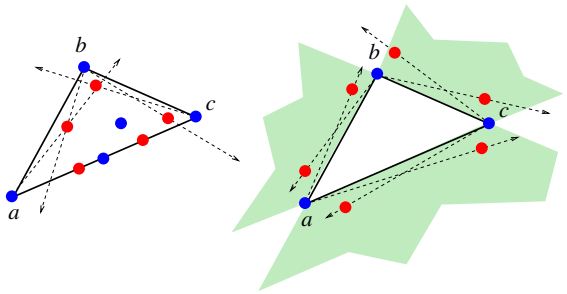



Figure 2: Safe conditions: Rules 1 and 2.

For three unguarded points $a, b, c \in S$ we say that they *satisfy the safe condition* if they satisfy either one of the following rules (refer to Figure 2):

*Rule 1:* There are points inside (or on the boundary of) $\triangle abc$, and within $\triangle abc$ a ray with apex in $\{a, b, c\}$ rotated inwards, starting from each incident edge to the apex, hits a guarded interior point before hitting an unguarded point.

*Rule 2:* There is no point of $S$ inside (or on the boundary of) $\triangle abc$, and, further, a ray with apex in $\{a, b, c\}$ rotated outwards, starting from each incident edge to the apex hits a guarded point that is within the corresponding "wedge" (shown in green in the figure), before hitting an unguarded point.

A key fact (proof omitted here) is the equivalence:

**Lemma 1** *A point set $S$ with guards at $G \subseteq S$ is in safe configuration with respect to spikes if and only if any three unguarded points of it satisfy the safe conditions.*

## 2 The UGPI: Using Interior Guards

In the UGPI we allow guards to be placed only at points of $S$ that are interior to the convex hull, $CH(S)$. Note that placing guards at *all* interior points is sufficient

to guard any polygonalization of $S$, since the $CH(S)$ vertices are convex vertices in any polygonalization of $S$, and a simple fact is that the reflex vertices of any simple polygon see all of the polygon. Our main result in this section is a proof that it is sometimes necessary to place guards at all interior points, in order to have a universal guard set:

**Theorem 2** *There exist configurations of $n$ points $S$, for arbitrarily large $n$, for which $CH(S)$ is a triangle, and the only universal guard set using only interior guards is the set of* all $n-3$ *interior points.*

*Proof sketch:* We utilize Lemma 1 and construct a careful configuration of points whose general structure is shown in Figure 3: The points $a, b, c \in S$ are the vertices of $CH(S)$. Six additional points (in red) are placed just inside each edge of $\triangle abc$, so that each is first hit by rays rotating inwards from the edges of $\triangle abc$. Then, very carefully located points are placed (in a sequence of "rounds") along each of three line segments (thick green in the figure), in such a way that all of these interior points must be guarded in order to avoid a spike (created by the unguarded point, together with two vertices of $\triangle abc$). (Each of the potential spikes is such that, in this configuration $S$, we can argue that there exists a polygonalization of $S$ that includes the spike.)

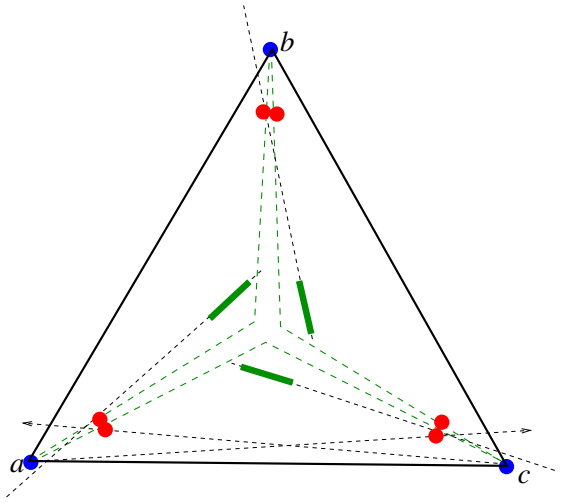



Figure 3: The overall configuration for the proof of Theorem 2.

## 3 The UGPG: Guarding Full Grids

**Theorem 3** *For $n = ab$ points $S$ on a regular $a \times b$ grid, $\lfloor \frac{n}{2} \rfloor$ guards are sufficient to guard all polygonalizations on $n$ points. Further, $\lfloor \frac{a}{2} \rfloor \cdot b$ guards are necessary to guard every polygonalization on $a \times b$ grid points ($a \leq b$) with size at least $4 \times 5$.*

The proof of sufficiency is based on either of two patterns of guard selection: (1) place guards at the odd posititions on odd-numbered rows and at even positions on even-numbered rows of the grid (i.e., place guards in the grid according to white squares on a checkboard); or (2) place guards at all positions on the even-numbered rows. We argue that with either placement strategy, any triangle with vertices at grid points, and no other grid points on the boundary or interior of the triangle, must have at leat one of its vertices guarded. This implies that the $\lfloor \frac{n}{2} \rfloor$ guards see every point in any polygonalization $P$ of $S$, since any such $P$ can be triangulated, and every triangle in any triangulation has at least one guard at a vertex. The proof of necessity is based on analyzing possible spikes in the grid, using the fact that in a solution an unguarded interior grid point cannot have both of its horizontal and vertical neighbors unguarded at the same time.

## 4 The General UGP, Bounded Layers, $k$-UGP

In [2] we prove a bound for the general UGP:

**Theorem 4** *For any $m = 2^h \geq 8$ such that $h \in \mathbb{N}$, there is a point set $P$ with $|P| = n = m^2 + 2 * m - 21$ that requires at least $(1 - \frac{5}{\sqrt{n}-1})n$ universal guards.*

The proof of this theorem is based on having the points evenly distributed on multiple convex layers in such a way that on each layer at most 4 points can be unguarded. We also consider sets $S$ on $m$ layers:

**Theorem 5** $(1 - \frac{1}{16n^{\frac{2m-1}{2m}}})n$ *are always sufficient to guard all polygonalizations for $n$ points that lie on $m$ convex layers.*

In [2] we also give results on the *$k$-universal* guarding problem, in which the guards must perform visibility coverage for a set of $k$ different polygonalizations of the input points (instead of *all* polygonalizations).

The complexity of deciding if a given set $S$ has a universal guard set of size at most $m$ is open; it is also open to obtain approximation algorithms for universal guarding.

## References

[1] V. Chvátal. A combinatorial theorem in plane geometry. *Journal of Combinatorial Theory, Series B*, 18:39–41, 1975.

[2] S. Fekete, Q. Li, J. Mitchell, C. Scheffer. The Universal Guards Problem. Submitted, 2016.

[3] S. Fisk. A short proof of Chvátal's watchman theorem. *Journal of Combinatorial Theory, Series B*, 24:374, 1978.

# Certified Homology Inference

Nicholas J. Cavanna* Kirk Gardner† Donald R. Sheehy‡

## 1 Introduction

The goal of homology inference is to compute a space's shape from a point cloud sampled near it. Given such a sample, one may want to know when we can reliably infer the homology of the space in question. Naturally, this requires making assumptions on the sample as well as the underlying space.

Niyogi, Smale, and Weinberger showed that one can infer the homology of a smooth manifold from finite points chosen uniformly at random near its surface [5]. Chazal and Lieutier relaxed this to include non-smooth bounded spaces in $\mathbb{R}^d$ via the so-called weak feature size [1]. Both assume that there is a sampled point within $\varepsilon$ of every point in the space in their sample. In their work on sensor networks, de Silva and Ghrist give sampling conditions for checking coverage of a shrunken version of a space assuming one can compute the distance from points to the boundary [3].

We show how these approaches can be combined in order to provide a computable inference of the homology of domain from a coordinate-free point sample. We do so on more general spaces in $\mathbb{R}^d$, only assuming a lower bound on the weak feature size of a compact, locally contractible domain, and that we can compute the distance to the boundary and between close pairs of sample points.

## 2 Background

**Distance Functions.** For a compact set $A \subset \mathbb{R}^d$, and metric $\mathrm{d}(\cdot,\cdot)$, define the distance function from $x \in \mathbb{R}^d$ to $A$ as $\mathrm{d}(x,A) := \min_{y\in A} \mathrm{d}(x,y)$. The $\varepsilon$-**offsets** of a set $A$ are defined as $A^\varepsilon := \{x \in X \mid \mathrm{d}(x,A) \leq \varepsilon\} = \cup_{x\in A}\mathrm{ball}_\varepsilon(x)$. Recall the set difference, or relative complement, of two sets $A$ and $B$ is defined as $A \setminus B := \{a \in A \mid a \notin B\}$. In this paper, the ambient space will be the one-point compactification of $\mathbb{R}^d$, $\mathbb{R}^d \cup \{\infty\}$, which is homeomorphic to the $d$-sphere, $S^d$, and the metric will be the Euclidean metric, $\|\cdot\|$.

Given a compact set $A \subset \mathbb{R}^d$, the critical value associated to a critical point $x$ of the distance function is $\mathrm{d}(x,A)$. The **weak feature size** of $A$ is defined to be the least positive critical value of $\mathrm{d}(\cdot,A)$ and is denoted $\mathrm{wfs}(A)$.

*University of Connecticut
†University of Connecticut
‡University of Connecticut

**Homology.** Homology is a tool from algebraic topology that gives a characterization of the shape of a space with regards to its $k$-dimensional holes. It is a topological invariant and as such it is preserved under homeomorphisms. We assume singular homology over a field, which implies that the resulting homology groups of a space $X$, written $H_*(X)$ when considered over all dimensions, are vector spaces. When referencing homology with respect to a specific dimension $k$, we write $H_k(X)$. If there exists a map between two spaces, e.g. $f : X \to Y$, then there is a map at the level of homology, $f_* : H_*(X) \to H_*(Y)$. We will denote such a map by $f_* := H_*(X \to Y)$.

**Čech and Rips Complexes.** When computing homology in practice, one often needs a finite simplicial complex to represent the space as their homology can be calculated via matrix multiplication.

Given a finite collection of points $P \in \mathbb{R}^d$, its **Čech complex** at scale $\varepsilon$ is defined as $\mathcal{C}_\varepsilon(P) := \{\sigma \subseteq P \mid \exists x \in \mathbb{R}^d : \max_{p\in\sigma} \|x-p\| \leq \varepsilon\}$.

Of note is that the Čech complex of a point set $P$ at scale $\varepsilon$ is the nerve of the collection $\{\mathrm{ball}_\varepsilon(p)\}_{p\in P}$, whose union is $P^\varepsilon$. By the nerve theorem [4] , $\mathcal{C}_\varepsilon(P)$ is homotopic to $P^\varepsilon$, and thus $H_*(\mathcal{C}_\varepsilon(P)) \cong H_*(P^\varepsilon)$. This implies that by studying the homology of the $\varepsilon$-Čech complex, one knows the homology of the $\varepsilon$-offsets.

As we simply know the distance between close points in the sample, this is not enough to compute the Čech complex. Instead we compute the Rips complex, as this can be computed by checking pairs of points' distance. The **(Vietoris-)Rips complex** of $P$ at scale $\varepsilon$ is defined as $\mathcal{R}_\varepsilon(P) := \{\sigma \subseteq P \mid \{p,q\} \in \mathcal{C}_\varepsilon(P) \text{ for all } p,q \in \sigma\}$. The following inclusions help us in relating the knowledge about a Čech complex to a Rips complex.

For all $\varepsilon > 0$ and a finite point set $P \subseteq \mathbb{R}^d$, $\mathcal{C}_\varepsilon(P) \subseteq \mathcal{R}_\varepsilon(P) \subseteq \mathcal{C}_{\vartheta_d\varepsilon}(P)$, where $\vartheta_d = \sqrt{\frac{2d}{d+1}}$.

## 3 Results

Throughout we assume a compact, locally contractible domain $\mathcal{D} \subset \mathbb{R}^d$, with boundary $\mathcal{B}$, and a finite point sampling $P \subset \mathcal{D}$. Given a constant $\beta \geq 0$, define $U_\beta := P \setminus B^\beta = \{p \in P \mid \mathrm{d}(p,\mathcal{B}) > \beta\}$. These are the points of $P$ at least a distance of $\beta$ away from the boundary. This definition leads to the following lemma relating a

subsampling to a shrunken domain, assuming we have coverage.

**Lemma 1** *Given a domain $\mathcal{D}$ with boundary $\mathcal{B}$ and a finite set $P \subset \mathcal{D}$ with $\alpha > 0$ such that $\mathcal{D} \setminus \mathcal{B}^{2\alpha} \subseteq P^{\alpha}$, for all $\gamma, \beta$ such that $\gamma, \beta \geq \alpha$, we have the following.*

$$U^{\gamma}_{\beta+\gamma} \subseteq \mathcal{D} \setminus \mathcal{B}^{\beta} \text{ and } \mathcal{D} \setminus \mathcal{B}^{\beta+\gamma} \subseteq U^{\gamma}_{\beta}.$$

The following lemma relates the homology of the inclusion between two sub-samplings of $P$ to the homology of the domain $\mathcal{D}$.

**Lemma 2** *Given a domain $\mathcal{D}$ with boundary $\mathcal{B}$ and a finite set $P \subset \mathcal{D}$ with $\alpha > 0$ such that $\mathcal{D} \setminus \mathcal{B}^{2\alpha} \subseteq P^{\alpha}$, let $\beta, \gamma, \varepsilon, \delta$ be constants such that $\delta \geq \varepsilon \geq \gamma \geq \alpha$ and $\beta \geq \varepsilon + \delta + \gamma$. If $\text{wfs}(\mathcal{B}) > \beta + \gamma$, then for all $\lambda \in (0, \text{wfs}(\mathcal{D}))$,*

$$rk(H_*(U^{\gamma}_{\beta} \hookrightarrow U^{\varepsilon}_{\delta})) = dim(H_*(\mathcal{D}^{\lambda})).$$

From this lemma we can prove our main theorem by switching to the Čech and Rips complexes. We assume that we know the distance between all $p, q \in P$ such that $\mathrm{d}(p, q) < \vartheta_d \alpha$, where $\alpha$ corresponds to the scale at which we have coverage. The constants used are a result of Lemma 2 and those required to achieve the Rips-Čech inclusions.

The theorem tells that we can compute the homology of a small offset of the domain by computing the Rips complexes at the two scales, and computing the image of the induced map between their homology groups. Furthermore, if $D$ is homotopic to $D^{\lambda}$, the image tells the homology of the domain itself.

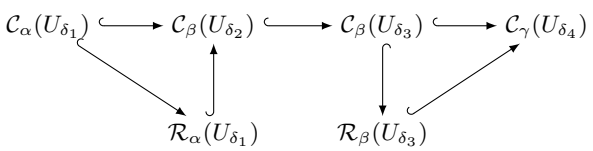
**Theorem 3** *Given $\mathcal{D} \subset \mathbb{R}^d$, a compact, locally contractible domain with boundary $\mathcal{B}$ such that $\text{wfs}(\mathcal{B}) > (2\vartheta_d^2 + 4\vartheta_d + 2)\alpha$, and finite point set $P \subset \mathcal{D}$ such that $\mathcal{D} \setminus \mathcal{B}^{2\alpha} \subseteq P^{\alpha}$, the following holds for all $\lambda \in (0, \text{wfs}(\mathcal{D}))$.*

$$im(H_*(\mathcal{R}_{\alpha}(U_{(2\vartheta_d^2+4\vartheta_d+1)\alpha}) \hookrightarrow \mathcal{R}_{\vartheta_d\alpha}(U_{(2\vartheta_d^2+\vartheta_d)\alpha}))) \cong H_*(\mathcal{D}^{\lambda}).$$

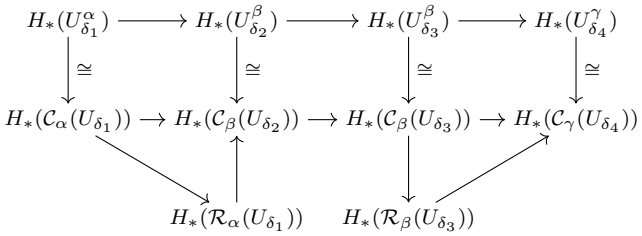
**Proof.** Let $0 < \alpha \leq \beta \leq \gamma$ and $\delta_1 \geq \delta_2 \geq \delta_3 \geq \delta_4 \geq 0$. This leads to the following inclusions of sub-samplings.

$$U^{\alpha}_{\delta_1} \hookrightarrow U^{\beta}_{\delta_2} \hookrightarrow U^{\beta}_{\delta_3} \hookrightarrow U^{\gamma}_{\delta_4}$$

If $\beta \geq \vartheta_d \alpha$ and $\gamma \geq \vartheta_d \beta \geq \vartheta_d^2 \alpha$, then we have the following diagram with Čech and Rips complexes, with the diagonal maps due to the Rips-Čech interleaving.




By applying the homology functor to the previous two diagrams, we get a commutative diagram of maps at the level of homology and vertical isomorphisms due to the Persistent Nerve Lemma [2].




If $\delta_1, \delta_4, \alpha, \gamma$ and $\delta_2, \delta_3, \beta$ are chosen such that they satisfy the assumptions of Lemma 2, then we have the following isomorphisms, as each vector space in question is finite-dimensional

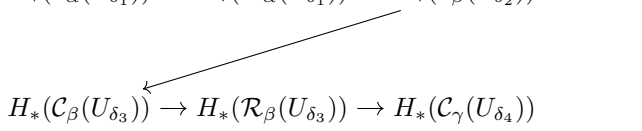
$$\text{im}(H_*(U^{\alpha}_{\delta_1} \hookrightarrow U^{\gamma}_{\delta_4})) \cong \text{im}(H_*(U^{\beta}_{\delta_2} \hookrightarrow U^{\beta}_{\delta_3})) \cong H_*(\mathcal{D}^{\lambda}).$$

This also gives us the following isomorphisms at the level of the Čech complexes.

$$\text{im}(H_*(\mathcal{C}_{\alpha}(U_{\delta_1}) \hookrightarrow \mathcal{C}_{\gamma}(U_{\delta_4})) \cong \text{im}(H_*(\mathcal{C}_{\beta}(U_{\delta_2}) \hookrightarrow \mathcal{C}_{\beta}(U_{\delta_3})) \cong H_*(\mathcal{D}^{\lambda}).$$

The conditions for the Rips-Čech interleaving and Lemma 2 are satisfied by making the following substitutions. $\beta = \vartheta_d \alpha$, $\gamma = \vartheta_d^2 \alpha$, $\delta_1 = (2\vartheta_d^2 + 4\vartheta_d + 1)\alpha$, $\delta_2 = (2\vartheta_d^2 + 3\vartheta_d)\alpha$, $\delta_3 = (2\vartheta_d^2 + \vartheta_d)\alpha$, and $\delta_4 = \vartheta_d^2 \alpha$.

Applying Lemma 3.2 from Chazal and Oudut [2] to the following homological sequence using the above Čech isomorphisms completes the proof.




□

## References


[1] Frédéric Chazal and André Lieutier. Weak feature size and persistent homology: Computing homology of solids in $R^n$ from noisy data samples. In *Proceedings of the 21st ACM Symposium on Computational Geometry*, pages 255–262, 2005.

[2] Frédéric Chazal and Steve Yann Oudot. Towards persistence-based reconstruction in euclidean spaces. In *Proceedings of the Twenty-fourth Annual Symposium on Computational Geometry*, SCG '08, pages 232–241, New York, NY, USA, 2008. ACM.

[3] Vin de Silva and Robert Ghrist. Homological sensor networks. *Notices Amer. Math. Soc.*, 54(1):10–17, 2007.

[4] Allen Hatcher. *Algebraic Topology*. Cambridge University Press, 2001.

[5] Partha Niyogi, Stephen Smale, and Shmuel Weinberger. Finding the homology of submanifolds with high confidence from random samples. *Discrete & Computational Geometry*, 39(1-3):419–441, 2008.

# Homology Localization by Hierarchical Blowups

Ahmed Abdelkader *


## Abstract

Topological descriptors such as the generators of homology groups are very useful in the analysis of complex data sets. It is often desired to find the smallest such generators to help localize the interesting features. One interpretation of localization utilizes a covering of the underlying space and computes generators contained within these covers. A similar construction was later used to compute persistence homology for smaller subsets in parallel before gluing the results. In this presentation, we describe a more efficient version of this construction and discuss how it can be used to find generators within a large class of subspaces.


## 1 Introduction

Persistent Homology is a crucial device in computational topology and finds wide application in data analysis and as a core component of a variety of algorithms. Although the formalism of homology provides efficient means to detect the existence of topological features, it cannot directly locate them within the space. This is often required to reason about the embedding of the data set in the measurement space and to allow further processing, e.g., cleaning up noises introduced in data collection or detecting holes in sensor networks.

One approach to locating topological features is to find the smallest generator for a given homology group [3]. Another approach is *localized homology* [6], which utilizes a covering of the space and computes homology bases compatible with the bases of the local pieces defined by the cover. However, choosing an appropriate cover was left to the domain expert.

The idea of computing homology through a cover was later reused to devise a parallel algorithm based on a hierarchical decomposition of the domain [4]. The algorithm performs reduction on local pieces in parallel before gluing the results, which is inherently expensive. Potentially better ways to go about gluing are provided by spectral sequences [2, 5].

In this presentation, we take a different look at homology localization utilizing the model of computation developed for the parallel setting. Using a hierarchical space decomposition, we aim to quickly report the topological descriptors within nearly arbitrary covers by gluing partial precomputed results. Motivated by recent developments in approximate range queries [1] we anticipate similar notions in topological data analysis.

We start by revisiting the construction used in [4] to enable finer decompositions of the domain as required for range queries. Then, we discuss the anticipated notion of *topological range queries* and their applications.

*University of Maryland, College Park, akader@cs.umd.edu

## 2 Preliminaries

A topological space $X$ may be represented by a *simplicial complex* $K$. A *filtration* is a nested sequence of simplicial complexes $K_0 \subseteq K_1 \subseteq \cdots \subseteq K_n = K$. Given a set of subcomplexes as a cover $\mathcal{C} = \{K^i\}_{i\in I}$, with $K = \cup\,\mathcal{C}$, the *Mayer-Vietoris Blowup Complex* $K^{\mathcal{C}} \subseteq K \times I$ is defined as:

$$K^{\mathcal{C}} = \cup_{J\subseteq I} \cup_{\sigma\in K^J} \sigma \times J, \quad \text{where } K^J = \cap_{j\in J} K^j. \quad (1)$$

Intuitively, the blowup complex creates one copy of each simplex $\sigma \in K$ for each of the covers containing it. This allows each cover to be processed independently. The blowup complex also includes additional copies of simplices where each subset of covers overlap. This marks the locations where these covers should be glued together to recover the original space.

Formally, the projection $\pi : K^{\mathcal{C}} \to K$ takes an element of the blowup to its first factor and induces a map on homology $\pi^* : H(K^{\mathcal{C}}) \to H(K)$. As $\pi$ is a homotopy equivalence [6], $\pi^*$ is an isomorphism. Then, by the Persistence Equivalence Theorem, the persistent homology of $K$ can be computed from $K^{\mathcal{C}}$ [4].

## 3 The Hierarchical Blowup Complex

We start by formalizing the notion of the blowup complex for a hierarchical cover as introduced in [4].

**Definition 1** *A Hierarchical Cover of height $h$ is a system of covers $\mathcal{H} = \{\mathcal{C}_i\}_{i\in[h]}$, with $\mathcal{H} = \cup\,\mathcal{C}_i$ $\forall i$, such that $\forall\, c^i_\alpha \in \mathcal{C}_i$, there is a unique $j$-parent, $c^j_{p_j(\alpha)} \in \mathcal{C}_j$, $\forall j \in [i]$, satisfying $c^i_\alpha \subseteq c^j_{p_j(\alpha)}$, where $p_i(c^i_\alpha) = c^i_\alpha$.*

Consider a simplex $\sigma \in K$ lying in the intersection of two subcovers at the lowest level, i.e., $\sigma \in c^h_x \cap c^h_y$ for $x \neq y$. We track $\sigma$ through the hierarchy until it possibly falls into a single subcover at a higher level. Note that $\sigma \in c^{i+1}_\alpha \implies \sigma \in c^i_{p_i(\alpha)}$.

Define the set of all levels where $\sigma$ falls into a pairwise intersection as $\lambda = \{j \in [h] \mid p_j(x) \neq p_j(y)\}$. It follows that $\sigma$ is contained in the intersection of $h + |\lambda|$ covers. Letting $\mathcal{F}_h = \cup_{i\in[h]}\mathcal{C}_i$, it follows from the product formula of the blowup complex that $K^{\mathcal{F}_h}$ will generate $2^{h+|\lambda|} - 1$ copies of $\sigma$. Although the cover sets form a hierarchy, they are distinct sets and $K^{\mathcal{F}_h}$ as defined in [4] does not seem to exploit their nesting structure. This makes it infeasible to work with deeper hierarchies where $h$ grows as a function of $|K|$ as $|K^{\mathcal{F}_h}| = O(2^h)|K|$.

To remedy this, we propose a recursive construction. Given a hierarchical cover $\mathcal{H}$, we lift each level of the hierarchy to a decomposition of the blowup complex at the above level. We denote the lifted cover $\mathcal{C}_i$ as $\hat{\mathcal{C}}_i$.

**Definition 2** *Given a hierarchical cover $\mathcal{H} = \{\mathcal{C}_i\}_{i\in[h]}$, the Hierarchical Blowup Complex $\{K^{\mathcal{H}_i}\}_{i\in[h]}$ is defined recursively as $K^{\mathcal{H}_1} = K^{\mathcal{C}_1}$ and*

$$K^{\mathcal{H}_{i+1}} = K^{\mathcal{H}_i{}^{\hat{\mathcal{C}}_{i+1}}}. \tag{2}$$

**Definition 3** *A lifted cover $\hat{\mathcal{C}}_i = \{\hat{c}_\alpha^{i+1} \mid c_\alpha^{i+1} \in \mathcal{C}_i\}$ where $\hat{c}_\alpha^1 = c_\alpha^1$ and*

$$\hat{c}_\alpha^{i+1} = \{(\sigma, J_1, \ldots, J_i) \in K^{\mathcal{H}_i} \mid \sigma \in c_\alpha^{i+1} \wedge p_i(\alpha) \in J_i\}.$$

Now, for $\sigma \in c_x^h \cap c_y^h$ and $i = \min\lambda$, $K^{\mathcal{H}_i}$ generates

$$\{(\hat{\sigma}_{i-1}, p_i(x)), (\hat{\sigma}_{i-1}, p_i(y)), (\hat{\sigma}_{i-1}, \{p_i(x), p_i(y)\})\},$$

where $\hat{\sigma}_{i-1}$ is the lifted copy of $\sigma$ in the common parent $\hat{c}^{i-1}_{p_{i-1}(x)}$. Further decomposing at level $j > i \in \lambda$, lifted simplices with only $p_{j-1}(x)$ or $p_{j-1}(y)$ get one new copy while mixed simplices again get three. Letting $s(j)$ be the number of copies of $\sigma$ at level $j$, we get $s(i) = 3$ and $s(j) = s(j-1) + 2$ for $j > i$. Summing over all levels, $K^{\mathcal{H}_h}$ will only create $h + O(|\lambda|^2)$ copies of $\sigma$. If $\mathcal{H}$ only has pairwise intersections at any level, $|K^{\mathcal{H}_h}| = O(h^2)|K|$. For typical decompositions $h = O(\log|K|)$ and we get only a polylogarithmic expansion.

Finally, we define the projections $\pi_1 : K^{\mathcal{H}_1} \to K$ and $\pi_i : K^{\mathcal{H}_{i+1}} \to K^{\mathcal{H}_i}$. Again, these maps are homotopy equivalences [6] and it follows that the induced maps on the homology of the hierarchical blowup complex are isomorphisms. We get the following:

**Theorem 1** *A filtration $K_1 \subseteq \cdots \subseteq K_i \subseteq \cdots \subseteq K$ induces filtrations of all levels of a hierarchical blowup complex $K_1^{\mathcal{H}_j} \subseteq \cdots \subseteq K_i^{\mathcal{H}_j} \subseteq \cdots \subseteq K^{\mathcal{H}_j}$. Passing to homology, we get a sequence of homology groups connected by isomorphisms at each level. By the Persistence Equivalence Theorem, the persistence pairs in all levels are the same.*

$$\begin{array}{ccccccc}
H(K_1) & \longrightarrow \ldots \longrightarrow & H(K_i) & \longrightarrow \ldots \longrightarrow & H(K_n) \\
\uparrow{\scriptstyle \pi^*_{1,1}} & & \uparrow{\scriptstyle \pi^*_{1,i}} & & \uparrow{\scriptstyle \pi^*_{1,n}} \\
H(K_1^{\mathcal{H}_1}) & \longrightarrow \ldots \longrightarrow & H(K_i^{\mathcal{H}_1}) & \longrightarrow \ldots \longrightarrow & H(K_n^{\mathcal{H}_1}) \\
\uparrow{\scriptstyle \pi^*_{2,1}} & & \uparrow{\scriptstyle \pi^*_{2,i}} & & \uparrow{\scriptstyle \pi^*_{2,n}} \\
H(K_1^{\mathcal{H}_2}) & \longrightarrow \ldots \longrightarrow & H(K_i^{\mathcal{H}_2}) & \longrightarrow \ldots \longrightarrow & H(K_n^{\mathcal{H}_2})
\end{array}$$

In this example, for a filtration with a 2-level blowup complex, the diagram commutes. Specifically, the persistence pairs of $\mathcal{C}_j$ encoded into $K^{\mathcal{H}_j}$ may be computed by gluing the results for subcovers at any level.

## 4 Topological Range Queries

We envision queries of the form $(\mathcal{Q}^r, \alpha, \epsilon)$, where $\mathcal{Q}^r$ is a *parameterized range* and $\alpha$ is a *filtration parameter*. For example, given a set of points $\mathcal{P} \subset \mathbb{R}^d$ the query can be defined over an *implicit* Vietoris-Rips Complex $\mathcal{R}^\alpha(\mathcal{P} \cap \mathcal{Q}^r)$, with approximations of both $\mathcal{Q}^r$ and $\mathcal{R}^\alpha$ within $\epsilon$. The query can then ask for the Betti numbers at a given $(r, \alpha)$ or possibly the persistence barcodes for a range of parameter settings, e.g., $[r_0, r_1]$ or $[\alpha_0, \alpha_1]$.

In $\mathbb{R}^d$, ranges are often simple geometric primitives, e.g., hyperrectangles and hyperspheres. To support a larger class of embeddings, it would be interesting to define corresponding notions of ranges in more general metric spaces, for which a variety of algorithmic results in the Euclidean setting were adapted. Another direction is to consider ranges defined directly on the complex, e.g., geodesic balls, utilizing tools from graph theory like recursive separators.

## References


[1] S. Arya, D. M. Mount, and E. Park. Approximate Geometric MST Range Queries. In *31st International Symposium on Computational Geometry (SoCG 2015)*, pages 781–795, 2015.

[2] U. Bauer, M. Kerber, and J. Reininghaus. Distributed computation of persistent homology. In *Proceedings of the Meeting on Algorithm Engineering & Experiments*, pages 31–38, 2014.

[3] T. K. Dey, A. N. Hirani, and B. Krishnamoorthy. Optimal homologous cycles, total unimodularity, and linear programming. *SIAM Journal on Computing*, 40(4):1026–1044, 2011.

[4] R. Lewis and D. Morozov. Parallel computation of persistent homology using the blowup complex. In *Proceedings of the 27th ACM Symposium on Parallelism in Algorithms and Architectures*, pages 323–331, 2015.

[5] D. Lipsky, P. Skraba, and M. Vejdemo-Johansson. A spectral sequence for parallelized persistence. *CoRR*, abs/1112.1245, 2011.

[6] A. Zomorodian and G. Carlsson. Localized homology. *Computational Geometry*, 41(3):126 – 148, 2008.

# Improved Bounds on the Growth Constant of Polyiamonds

Gill Barequet* Mira Shalah*

**Abstract**

In this paper we provide improved lower and upper bounds on the asymptotic growth constant of polyiamonds, proving that it is between 2.8424 and 3.6050.

## 1 Introduction

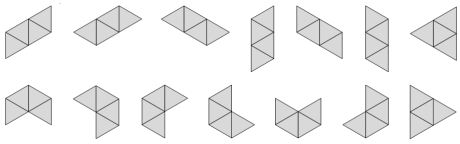


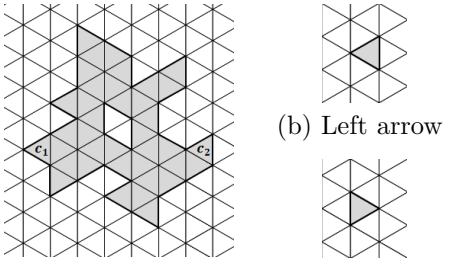
Figure 1: 14 tetriamonds

A *polyomino* of size $n$ is an edge-connected set of $n$ cells on the square lattice $\mathbb{Z}^2$. Similarly, a *polyiamond* of size $n$ is an edge-connected set of $n$ cells on the triangular lattice. *Fixed* polyiamonds are considered distinct if they have different *shapes* or *orientations*. In this paper we consider only fixed polyiamonds, and so we refer to them simply as "polyiamonds." Figure 1 shows all the polyiamonds of size 4.

In general, a connected set of cells on a lattice is called a *lattice animal*. The fundamental combinatorial problem concerning lattice animals is "How many animals with $n$ cells are there?" and is one of the long-standing open problems in combinatorial geometry.

The symbol $A(n)$ usually denotes the number of polyominoes of size $n$. As no analytic formula for the number of animals is yet known for any nontrivial lattice, a great portion of the research has so far focused on investigating the growth constant of animals, and a few asymptotic results are known. Klarner [2] showed that the limit $\lambda := \lim_{n\to\infty} \sqrt[n]{A(n)}$ exists. The convergence of $\lim_{n\to\infty} A(n+1)/A(n)$ to $\lambda$ was proven only three decades later by Madras [4]. Similarly, $T(n)$ usually denotes the number of polyiamonds of size $n$. Elements of the sequence $T(n)$ were computed up to $n = 75$ [1], and the limits $\lim_{n\to\infty} \sqrt[n]{T(n)}$ and $\lim_{n\to\infty} T(n+1)/T(n)$ exist and are equal. Let, then, $\lambda_T = \lim_{n\to\infty} \sqrt[n]{T(n)} = \lim_{n\to\infty} T(n+1)/T(n)$ denote the growth constant of polyiamonds.

Klarner [2] showed that $\lambda_T \geq 2.13$, by taking the square root of 4.54, a lower bound he computed for the growth constant of animals on the rhomboidal lattice. Rands and Welsh [5] showed that $\lambda_T \geq (T(n)/(2(1+\lambda_T)))^{1/n}$ for any $n \in \mathbb{N}$. Using the upper bound $\lambda_T \leq 4$ (see below), and knowing at that time $T(n)$ for $1 \leq n \leq 20$ only, they showed that $\lambda_T \geq (T(20)/10)^{1/20} \approx 2.3011$. Nowadays, knowing $T(n)$ up to $n = 75$, we can obtain, using the same method, that $\lambda_T \geq (T(75)/10)^{1/75} \approx 2.7714$. Moreover, using the upper bound we obtain in Section 3, we get that $\lambda_T \geq (T(75)/(2(1+3.6050)))^{1/75} \approx 2.7744$. However, we take a different approach and improve on this.




(a) A polyiamond (b) Left arrow (c) Right arrow

Figure 2: Polyiamonds on the triangular lattice

Using a simple argument, Lunnon [3] proved that $\lambda_T \leq \lim_{n\to\infty} \binom{2(n-1)}{n-1}^{1/n} = 4$. As can be seen, there is a large gap between the lower and upper bounds on $\lambda_T$. In this paper we improve both bounds, showing that $2.8424 \leq \lambda_T \leq 3.6050$.

## 2 Lower Bound

A concatenation of two polyiamonds $P_1, P_2$ is the translation of $P_1$ relative to $P_2$, so that $P_1, P_2$ do not overlap but their union is a valid (connected) polyiamond, and all the translated versions of the cells of $P_1$ are smaller than the cells of $P_2$ under a proper definition of a lexicographic order on the cells of the lattice.

**Theorem 1** $\lambda_T \geq 2.8424$.

**Proof.** We orient the triangular lattice as is shown in Figure 2(a), and define a lexicographic order on the cells of the lattice as follows: A cell $c_1$ is *smaller* than cell $c_2 \neq c_1$ if the lattice column of $c_1$ is to the left of the column of $c_2$, or if $c_1, c_2$ are in the same column and $c_1$ is below $c_2$. Denote triangles which look like a "left arrow" (Figure 2(b)) as triangles of Type 1, and triangles which look like a "right arrow" (Figure 2(c)) as

*Dept. of Computer Science, The Technion, Haifa 32000, Israel. E-mail: {barequet,mshalah}@cs.technion.ac.il

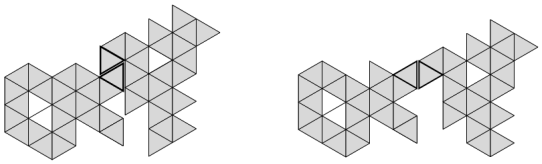


Figure 3: Concatenations of Type 1 and Type 2 triangles

triangles of Type 2. Let $T_1(n)$ be the number of polyiamonds of size $n$ whose *largest* triangle is of Type 1, and let $T_2(n)$ be the number of polyiamonds of size $n$ whose *largest* triangle is of Type 2. Obviously, we have $T(n) = T_1(n) + T_2(n)$. By rotational symmetry, the number of polyiamonds of size $n$, whose *smallest* triangle is of Type 2, is also $T_1(n)$, and the number of polyiamonds, whose *smallest* triangle is of Type 1, is $T_2(n)$.

We now proceed with a standard concatenation argument, tailored to the specific case of the triangular lattice. Interestingly, not all pairs of polyiamonds of size $n$ can be concatenated. In addition, there exist many polyiamonds of size $2n$ which cannot be represented as the concatenation of two polyiamonds of size $n$. Polyiamonds, whose largest triangle is of Type 1, can be concatenated only to polyiamonds whose smallest triangle is of Type 2, and this can be done in two different ways: horizontal and vertical (see Figure 3), yielding $2(T_1(n))^2$ concatenations. Polyiamonds whose largest triangle is of Type 2, can be concatenated, in a single way, only to polyiamonds whose smallest triangle is of Type 1, yielding $(T_2(n))^2$ concatenations. Thus, as argued above,

$$2(T_1(n))^2 + (T_2(n))^2 \leq T(2n). \qquad (1)$$

Let $x = x(n)$ be the fraction of $T_1(n)$ out of $T(n)$, i.e., $T_1(n) = xT(n)$ and $T_2(n) = (1-x)T(n)$. Eq. (1) can then be rewritten as $T(2n) \geq 2(xT(n))^2 + ((1-x)T(n))^2 = (3x^2 - 2x + 1)T^2(n)$. The function $f(x) = 3x^2 - 2x + 1$ assumes its minimum at $x = 1/3$ and $f(1/3) = 2/3$. Hence, $\frac{2}{3}T^2(n) \leq T(2n)$. By manipulating this relation, we obtain $\left(\frac{2}{3}T(n)\right)^{1/n} \leq \left(\frac{2}{3}T(2n)\right)^{1/(2n)}$. This implies that the sequence $\left(\frac{2}{3}T(k)\right)^{1/k}, \left(\frac{2}{3}T(2k)\right)^{1/(2k)}, \left(\frac{2}{3}T(4k)\right)^{1/(4k)}, \ldots$ is monotone increasing for any value of $k$, and, as a subsequence of $\left(\left(\frac{2}{3}T(n)\right)^{1/n}\right)$, it converges to $\lambda_T$ too. Therefore, any term of the form $\left(\frac{2}{3}T(n)\right)^{1/n}$ is a lower bound on $\lambda_T$. In particular, $\lambda_T \geq (\frac{2}{3}T(75))^{1/75} \approx 2.8424$. □

## 3 Upper Bound

### 3.1 Number of Compositions

**Definition 1** *A polyiamond $P$ can be* decomposed *into two polyiamonds $P_1, P_2$ if the cell set of $P$ can be split into two complementing non-empty subsets, such that each subset is a valid (connected) polyiamond. We also say that the polyiamonds $P_1, P_2$ can be* composed *so as to yield the polyiamond $P$.*

**Theorem 2** *(Composition) Let $P_1, P_2$ be two polyiamonds of sizes $n_1$ and $n_2$, respectively. Then, $P_1$ and $P_2$ can be composed and yield at most $(n_1+2)(n_2+2)/2$ different polyiamonds.*

### 3.2 Balanced Decompositions

**Definition 2** *A decomposition of a polyiamond of size $n$ into two polyiamonds $P_1, P_2$ is* $k$-balanced *if $k \leq |P_i| \leq n-k$ (for $i = 1, 2$).*

**Theorem 3** *Every polyiamond of size $n$ has at least one $\lceil (n-1)/3 \rceil$-balanced decomposition.*

### 3.3 The Bound

**Theorem 4** $\lambda_T \leq 3.6050$.

**Proof.** Theorems 2 and 3 imply that

$$T(n) \leq \sum_{k=\lceil \frac{n-1}{3} \rceil}^{\lfloor n/2 \rfloor} (1 - \frac{\delta_{k,n/2}}{2}) \frac{(k+2)(n-k+2)}{2} T(k)T(n-k).$$

(The factor $(1-\delta_{k,n/2}/2)$ compensates for double counting which occurs when $P_1, P_2$ are of the same size.) Now define the sequence $T'(n)$ as follows.

$$T'(n) = \begin{cases} T(n) & 1 \leq n \leq 75; \\ \sum_{k=\lceil \frac{n-1}{3} \rceil}^{\lfloor n/2 \rfloor} (1 - \frac{\delta_{k,n/2}}{2}) \frac{(k+2)(n-k+2)}{2} T'(k)T'(n-k) & n > 75. \end{cases}$$

Since $T'(n) \geq T(n)$ for any value of $n \in \mathbb{N}$, the growth constant of $T'(n)$, if it exists, is an upper bound on $\lambda_T$. Numerical calculations show that $T'(n)$ has an asymptotic growth constant of about 3.6050. □

## References


[1] A.J. Guttmann, *Polygons, Polyominoes, and Polycubes*, *Lecture Notes in Physics*, 775, Springer and Canopus Academic Publishing Ltd., 2009.

[2] D.A. Klarner, Cell growth problems, *Canadian J. of Mathematics*, **19**, 851–863, 1967.

[3] W.F. Lunnon, Couting hexagonal and triangular polyominoes, in: *Graph Theory and Computing* (R.C. Read, ed.), Academic Press, New York, 1972, 87–100.

[4] N. Madras, A pattern theorem for lattice clusters, *Annals of Combinatorics*, **3**, 357–384, 1999.

[5] B.M.I. Rands and D.J.A. Welsh, Animals, trees and renewal sequences, *IMA J. of Applied Mathemathics*, **27**, 1–17, 1981; Corrigendum, **28**, 107, 1982.

# New results on gap ratio and correlation with other uniformity measures

Arijit Bishnu* Sameer Desai* Arijit Ghosh*

## 1 Introduction

The gap ratio was introduced as a measure of spatial uniformity by Teramoto et al. [15], motivated by combinatorial approaches and applications in digital halftoning [1, 3, 4, 12]. They defined the problem of maintaining low gap ratio in an online setting. We generalised the problem by extending the definition of gap ratio to general spaces and removing its online nature in [5]. We recall that definition below.

**Definition 1** *Let $(\mathcal{M}, \delta)$ be a metric space and $P$ be a set of $k$ points from $\mathcal{M}$. Define the minimum gap as $r_P \coloneqq \min_{p,q \in P, p \neq q} \frac{\delta(p,q)}{2}$. The maximum gap brings into play the interrelation between the metric space $\mathcal{M}$ and $P(\subset \mathcal{M})$, and is defined as $R_P \coloneqq \sup_{q \in \mathcal{M}} \delta(q, P)$ where $\delta(q, P) \coloneqq \min_{p \in P} \delta(q, p)$. The gap ratio for the point set $P$ is defined as $GR_P \coloneqq \frac{R_P}{r_P}$.*

The space $\mathcal{M}$, as in Definition 1 can be either continuous or discrete. The measure itself makes sense over unbounded spaces as well, if, we remove the finiteness of $P$ from the definition.

In [5] we use the generalized definition of gap ratio to pose combinatorial optimization questions where $\mathcal{M}$, for example, can be a set $S$ of $N$ points, and we would like to choose a subset $P \subset S$ of $k$ points from $S$, such that the gap ratio is minimized. The formal statement of the problem is as follows.

**Definition 2 (The gap ratio problem)** *Given a metric space $(\mathcal{M}, \delta)$, an integer $k$ ($k < |\mathcal{M}|$, if $\mathcal{M}$ is finite) and a parameter $g$, find a set $P \subset \mathcal{M}$ such that $|P| = k$ and $GR_P \leqslant g$.*

We showed lower bounds, NP-hardness and approximation results for various spaces (discrete and continuous) in [5].

In a geometric sense, the maximum gap is analogous to the minimum radius required to cover $\mathcal{M}$ with equal sized balls (i.e., *covering balls*) around each point of $P$, and the minimum gap is the maximum radius of equal sized balls around each point of $P$ having pairwise disjoint interiors (i.e., *packing balls*). This makes the problem particularly difficult in continuous spaces. The problems on optimal packing and optimal covering are classical problems and not much progress has been made. For example, proven optimal packings of $k$ disks in a square are known for only upto $k = 30$ [14].

**Previous works.** Gap ratio as a uniformity measure has been studied by researchers, both in computer science [2, 5, 15, 16] and mathematics [6, 7, 8, 11, 13]. Statisticians also look at uniformity measure. Ong et al. [10] review seven statistical measures of uniformity dividing them into three classes viz., (i) point-to-point measures, characterized by their focus on pair wise distances, (ii) volumetric measures which require the Voronoi diagram to be computed and (iii) discrepancy which seek to distribute points proportionally in every fraction of the metric space. Gap ratio due to its use of covering and packing radius is a volumetric measure. In this paper our metric space is the unit square. Our results are given in the next section.

## 2 Results

**Lower bounds and a construction.** We explore a lower bound for gap ratio in the unit square.

**Theorem 1** *Let the point set $P$ be the vertex set of a uniform constrained Delaunay triangulation of the unit square with $|P|$ points such that the maximum angle is at least $\frac{\pi}{2}$. Then $GR_P \geqslant \sqrt{2}$.*

We can construct point sets of certain sizes which achieve the lower bound using the farthest point algorithm [5] shown in Algorithm 1. This gives a gap ratio either 2 or $\sqrt{2}$ in the unit square. The point sets achieving the lower bound are characterized by the following theorem.

**Theorem 2** *If the set $P$ of $k \geqslant 4$ points is sampled from the unit square using Algorithm 1, then the gap ratio is $\sqrt{2}$ if and only if $k = G_j$ where,*

$$G_j = \begin{cases} 4^{i-1} + 2^i + 1 & j = 2i - 1 \\ 2 \cdot 4^{i-1} + 2^i + 1 & j = 2i \end{cases}$$

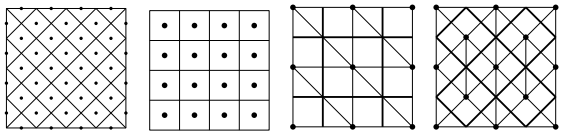
*for $i \in \mathbb{N}$.*

Figure 1 gives examples of some cases where the optimal gap ratio is achieved.

*ACM Unit, Indian Statistical Institute, Kolkata

**Algorithm 1** Pseudocode of *Farthest-point-insertion*$(\mathcal{M}, k)$

**Input:** metric space $(\mathcal{M}, \delta)$ and $k$;
**Initialize:** find $q_1, q_2 \in \mathcal{M}$ with $\delta(q_1, q_2) = diam(\mathcal{M})$ and $S_2 = \{q_1, q_2\}$;
**for** $i = 2$ to $k - 1$ **do**
  $q_{i+1} \leftarrow \text{argmax}_{p \in \mathcal{M}} \delta(p, S_i)$;
  $S_{i+1} \leftarrow S_i \cup \{q_{i+1}\}$;
**end for**
**Output:** $S_k$ and $GR_{S_k} = \frac{R_{S_k}}{r_{S_k}}$;



(a) (b) (c) (d)

Figure 1: Some cases when gap ratio is $\sqrt{2}$ (a) $k = 40$ (b) $k = 16$ (c) $k = 9$ (d) $k = 13$

**Correlation with other measures.** To establish correlations with other uniformity measures we do extensive experiments by sampling points using the Poisson disk sampling technique, and also obtain centroidal Voronoi tessellations and compute the Spearman's Rank correlation coefficient of gap ratio with discrepancy [9] and some other measures reviewed by Ong et al. [10]. These experiments establish that gap ratio is highly correlated with other measures of uniformity. The correlation obtained for gap ratio with other measures is over 0.95 when the measures are computed 1000 times with samples taken using Poisson disk sampling or over separate iterations of Lloyd's algorithm to obtain a centroidal Voronoi tessellation. The correlation is extremely bad when the same is done using random samples.

We expect a good correlation between gap ratio and discrepancy when both have low values and thus the point set is reasonably uniform. The following theorem captures that.

**Theorem 3** *For $k$ points in the interval $[0, 1]$ with discrepancy $D_k < \frac{2}{k}$, $GR \leqslant \frac{6}{2 - kD_k} - 2$.*

## References


[1] T. Asano. Computational Geometric and Combinatorial Approaches to Digital Halftoning. In *Proc. CATS2006*, page 3, 2006.

[2] T. Asano. Online uniformity of integer points on a line. *IPL*, 109(1):57–60, 2008.

[3] T. Asano, N. Katoh, K. Obokata, and T. Tokuyama. Combinatorial and Geometric Problems Related to Digital Halftoning. In *Theor. Found. Comput. Vision*, pages 58–71, 2002.

[4] T. Asano, N. Katoh, K. Obokata, and T. Tokuyama. Matrix Rounding under the $L_p$-Discrepancy Measure and Its Application to Digital Halftoning. *SIAM J. Comput.*, 32(6):1423–1435, 2003.

[5] A. Bishnu, S. Desai, A. Ghosh, M. Goswami, and S. Paul. Uniformity of point samples in metric spaces using gap ratio. *ArXiv e-prints*, 2014, 1411.7819.

[6] A. V. Bondarenko, D. P. Hardin, and E. B. Saff. Mesh ratios for best-packing and limits of minimal energy configurations. *Acta Mathematica Hungarica*, 142(1):118–131, 2013.

[7] J. S. Brauchart and P. J. Grabner. Distributing many points on spheres: Minimal energy and designs. *J. Complexity*, 31(3):293 – 326, 2015.

[8] J. C. Lagarias and P. A. B. Pleasants. Local complexity of delone sets and crystallinity. Canad. Math. Bull. Vol 48 (4), 2002 pp. 634-652, 2001, arXiv:math/0105088.

[9] J. Matoušek. *Geometric Discrepancy: An Illustrated Guide*. Springer, 1999.

[10] M. S. Ong, Y. C. Kuang, and M. P.-L. Ooi. Statistical measures of two dimensional point set uniformity. *Comput. Stat. & Data Analy.*, 56(6):2159 – 2181, 2012.

[11] S. S. Ryshkov. Density of an (r, R)-system. *Math. notes Acad. Sciences USSR*, 16(3):855–858.

[12] K. Sadakane, N. Takki-Chebihi, and T. Tokuyama. Discrepancy-Based Digital Halftoning: Automatic Evaluation and Optimization. In *Theoretical Foundations of Computer Vision*, pages 301–319, 2002.

[13] A. Schürmann and F. Vallentin. Computational approaches to lattice packing and covering problems. In *DCG. 35 (2006) 73116.*

[14] P. G. Szabó, M. C. Markót, and T. Csendes. *Essays and Surveys in Global Optimization*, chapter Global Opt. in Geom.– Circle Packing into the Square, pages 233–265. Springer US, 2005.

[15] S. Teramoto, T. Asano, N. Katoh, and B. Doerr. Inserting Points Uniformly at Every Instance. *IEICE TRANSACTIONS Inf. and Syst.*, 89-D(8):2348–2356, 2006.

[16] Y. Zhang, Z. Chang, F. Y. L. Chin, H.-F. Ting, and Y. H. Yung H. Tsin. Uniformly inserting points on square grid. *IPL*, 111(16):773–779, 2011.

# All-Pairs Shortest Paths in Unit Disk Graphs in Slightly Subquadratic Time

Timothy M. Chan* Dimitrios Skrepetos*


## Abstract

We study the all-pairs shortest paths problem in (unweighted) unit disk graphs. The previous best solution for this problem required $O(n^2 \log n)$ time, by running the $O(n \log n)$-time single-source shortest path algorithm of Cabello and Jejčič (2015) from every source vertex, where $n$ is the number of vertices. We not only manage to eliminate the logarithmic factor, but also obtain the first (slightly) subquadratic algorithm for the problem, running in $O(n^2\sqrt{\frac{\log\log n}{\log n}})$ time. Our algorithm computes an implicit representation of all the shortest paths, and, in the same amount of time, can also compute the diameter of the graph.


## 1 Introduction

The all-pairs shortest paths (APSP) problem is one of the most well known problems in the field of algorithms: given a graph $G = (V, E)$ with $n$ vertices, where each edge has a real weight, we are asked to compute the shortest paths between all pairs of vertices. The classical algorithm of Floyd and Warshall solves the problem in $O(n^3)$ time. The current fastest algorithm achieves superpolylogarithmic speedup, requiring $\frac{n^3}{2^{\Omega(\sqrt{\log n})}}$ time [11, 3]. For the case of unweighted undirected graphs, the problem can be solved in matrix multiplication time [10, 5].

One very important class of graphs arising from computational geometry is unit disk graphs. A unit disk graph is the intersection graph of a set of unit disks, which is defined by creating a vertex for each unit disk and an edge between any two unit disks that intersect each other. Equivalently, given a set $S$ of $n$ points in the plane (the disk centers, after rescaling by a half), the unit disk graph is defined by setting $V = S$ and creating an edge between any two points of $S$ whose Euclidean distance is at most 1. The edges are unweighted. The unit disk graph with $n$ vertices may contain $\Theta(n^2)$ edges, as every unit disk may intersect with every other unit disk. Since we aim for a subquadratic solution to the all-pairs shortest path problem, we do not construct the set of the edges explicitly in our algorithms.

Unit disk graphs are among the most well-studied families of graphs in geometry, with motivation from communication networks. Unit disk graphs are related to planar graphs: by the circle packing theorem any planar graph can be represented as a coin disk graph, although the disks may have different radii; on the other hand, planar graphs do not have large cliques unlike unit disk graphs. Frederickson [4] gave an $O(n^2)$-time algorithm for solving the APSP problem in weighted planar graphs. Chan [2] improved the bound for unweighted directed planar graphs with an $O(n^2\frac{\log\log n}{\log n})$-time algorithm (and also considered general unweighted undirected sparse graphs). Wulff-Nielsen [12] independently developed another $O(n^2\frac{\log\log n}{\log n})$-time algorithm for computing the diameter of unweighted undirected planar graphs (and also announced similar results for the weighted case).

## 2 Our Contributions

In this paper, we provide an algorithm for the APSP problem in unit disk graphs that requires $O(n^2\sqrt{\frac{\log\log n}{\log n}})$ time. The previously fastest solution was to run from each vertex the single-source shortest path algorithm of Cabello and Jejčič [1], which required $O(n^2 \log n)$ total time. (See [9, 6] for other previous results on shortest paths in unit disk graphs.) Therefore, we not only shave off the extra logarithmic factor of the previous result, but also provide the first (slightly) subquadratic solution to the problem. Our algorithm computes an implicit representation of the shortest paths: we encode the $O(n^2)$ shortest path distances and predecessors using bit-packing techniques, so that we are still able to retrieve the shortest path distance of a pair of vertices in $O(1)$ time and the shortest path $\pi$ itself in time linear in the number of vertices of $\pi$. In the same amount of time, we can also compute the diameter of the graph. Our algorithm operates in the RAM model of computation assuming $(\log n)$-bit words, which is standard in most algorithm analysis.

In recent years, obtaining polylogarithmic speedup for standard algorithmic problems have received considerable renewed attention. Such problems include 3SUM, Fréchet distance, combinatorial Boolean matrix multiplication, $k$-cliques, Klee's measure problem, CFL reachability, and many more. Our result can be seen as another contribution along this line of research.

*Cheriton School of Computer Science, University of Waterloo ({tmchan,dskrepetos}@uwaterloo.ca)

## 3 Our Techniques

The polylogarithmic improvement that we obtain for APSP in unit disk graphs goes beyond standard word RAM tricks. First, we present a new algorithm for the *single-source* shortest path problem for unit disk graphs, running in linear time after presorting the $x$- and $y$-coordinates of the input points. This improves over Cabello and Jejčič's single-source algorithm [1]. Their algorithm started with the Delaunay triangulation and performed repeated nearest neighbor queries, which inherently required $\Omega(n \log n)$ time even excluding preprocessing cost. Our new algorithm is instead based on a simple grid approach and exploits a linear-time Graham-scan-like procedure for computing upper envelopes of presorted unit disks. (According to [1], Efrat has also observed an alternative, grid-based $O(n \log n)$-time algorithm, but his suggested solution seemed a bit more complicated and used a semi-dynamic data structure of Efrat et al. [8], which also inherently required $\Omega(n \log n)$ time even after presorting.)

Second, we extend the single-source algorithm to the case of *multiple* ($k = o(\log n)$) sources that lie in a cluster, i.e., in a common grid cell. In this case, we have to construct not just one but $k$ upper envelopes, one for each source. This leads to a new kind of data structure problem: preprocess a set of unit disks (a "universe") so that given any subset of the universe, we can compute the upper envelope of the subset in slightly *sublinear* time. Our ideas can similarly be applied to the following (even more natural) problem of independent interest: preprocess a point set (a universe) in 2D so that given any subset of the universe, we can compute its convex hull, again in slightly sublinear time. Note that the input subset and the output can be encoded with linear number of bits, and thus slightly sublinear number of words. Solving problems for "preprocessed universes" is a relatively recent research direction; our result with slightly sublinear time provides an unusual addition to this body of work.

Finally, to obtain our subquadratic-time APSP algorithm, we draw inspiration from previous algorithms for planar graphs [2, 12], which use planar separators to decompose into regions of polylogarithmic size, and table lookup techniques to handle each region. However, this approach does not directly apply to unit disk graphs, because there could be large cliques and no small separators. On the other hand, when there are many large cliques, intuitively we should be able to exploit the multi-source algorithm developed earlier to handle such clusters more efficiently. The challenge lies in how to carefully combine these two approaches. For unit disk graphs, we end up avoiding planar separators and instead adopting a simpler *shifted grid* strategy [7]. This strategy is standard for geometric approximation algorithms, but we use the technique in a new and interesting way to design an *exact* algorithm (with an intricate balancing of parameters).

## References


[1] Sergio Cabello and Miha Jejčič. Shortest paths in intersection graphs of unit disks. *Computational Geometry*, 48:360–367, 2015.

[2] Timothy M. Chan. All-pairs shortest paths for unweighted undirected graphs in $o(mn)$ time. *ACM Transactions on Algorithms*, 8:1–17, 2012.

[3] Timothy M. Chan and Ryan Williams. Deterministic APSP, orthogonal vectors, and more: Quickly derandomizing Razborov–Smolensky. In *Proceedings of the 27th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1246–1255, 2016.

[4] Greg N. Frederickson. Fast algorithms for shortest paths in planar graphs, with applications. *SIAM Journal on Computing*, 16:1004–1022, 1987.

[5] Zvi Galil and Oded Margalit. All pairs shortest distances for graphs with small integer length edges. *Information and Computation*, 134(2):103–139, 1997.

[6] Jie Gao and Li Zhang. Well-separated pair decomposition for the unit-disk graph metric and its applications. *SIAM Journal on Computing*, 35(1):151–169, 2005.

[7] Dorit S. Hochbaum and Wolfgang Maass. Approximation schemes for covering and packing problems in image processing and VLSI. *Journal of the ACM*, 32(1):130–136, 1985.

[8] Alon Efrat, Alon Itai, and Matthew J. Katz. Geometry helps in bottleneck matching and related problems. *Algorithmica*, 31(1):1–28, 2001.

[9] Liam Roditty and Michael Segal. On bounded leg shortest paths problems. *Algorithmica*, 59(4):583–600, 2011.

[10] Raimund Seidel. On the all-pairs-shortest-path problem in unweighted undirected graphs. *Journal of Computer and System Sciences*, 51(3):400–403, 1995.

[11] Ryan Williams. Faster all-pairs shortest paths via circuit complexity. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing (STOC)*, pages 664–673, 2014.

[12] Christian Wulff-Nilsen. Constant time distance queries in planar unweighted graphs with subquadratic preprocessing time. *Computational Geometry*, 46(7):831–838, 2013.

# On Minimum Area Homotopies

Brittany Terese Fasy, Selcuk Karakoc, Carola Wenk

## 1 Introduction

The minimum homotopy area between two simple curves has been defined in [1], where a polynomial-time dynamic programming algorithm has been given. This algorithm, however, relies crucially on the fact that both input curves are simple. We generalize minimum homotopy area to arbitrary (normal) curves and provide the first algorithm to compute it. We show that minimum-area homotopies can be represented as contractions of self-overlapping subcurves of the input curve. Even though the runtime of our algorithm is currently exponential, we believe that our structural results lay the groundwork for a polynomial-time algorithm.

## 2 Normal Curves and Homotopies

A closed curve $C$ is a continuous map $C : [0,1] \to \mathbb{R}^2$ such that $C(0) = C(1)$. In this paper, we will use $[C]$ to denote the image of a map $C$. We assume that $C(0)$ is on the boundary of the outer face. A point $p \in [C]$ is *ordinary* if the preimage $C^{-1}(p)$ is a single point. A point $p \in [C]$ is called a *simple crossing point* if there exist exactly two points $t_1, t_2 \in [0,1]$ such that $p = C(t_1) = C(t_2)$ and if $C'(t_1)$ and $C'(t_2)$ are linearly independent. We assume $C^{-1}(C(0)) = \{0,1\}$, i.e., $C(0)$ is not a crossing point. A closed curve $C$ is called *normal* if there exist only a finite number of simple crossing points and all points in $[C]$, other than $C(0)$ and simple crossing points, are ordinary.

Whitney has shown that any closed curve can be made normal by arbitrary small deformation [5]. Hence we only consider normal curves from now on. We call a curve $C$ *almost normal* if it has a finite number of simple crossing points, type II and III singularities, and all other points are ordinary. $C$ has a type II singularity at $p \in [C]$ if there exist $t_1, t_2 \in [0,1]$ such that $C^{-1}(p) = \{t_1, t_2\}$ and if $C'(t_1)$ and $C'(t_2)$ exist but are linearly dependent. $C$ has a type III singularity if there exist three numbers $t_1, t_2, t_3 \in [0,1]$ such that $C^{-1}(p) = \{t_1, t_2, t_3\}$.

A homotopy $H : [0,1] \times [0,1] \to \mathbb{R}^2$ between two normal curves, $C$ and $C'$, is *normal* if there exists a finite set $S = \{s_1, \ldots, s_m\} \in [0,1]$ such that for each $t \in [0,1] - S$, $H_t$ is normal, and for each $s \in S$, $H_s$ is almost normal. Here, $H_s$ denotes an intermediate curve of the homotopy $H$.

Combinatorially, any normal homotopy of a normal closed curve can be described as a sequence of three types of events which we call *Titus moves* [3]:

- Type I: Destroying ($\mathrm{I}_a$) or creating ($\mathrm{I}_b$) a simple monogon.
- Type II: Destroying ($\mathrm{II}_a$) or creating ($\mathrm{II}_b$) a simple bigon.
- Type III: Inverting a triangle.

A $\mathrm{I}_a$ move takes a simple sub-loop $\gamma$ of an intermediate curve $H_s$ and contracts it to a point $b \in [H]$. We call the point $b$ an *anchor point* of the homotopy $H$. If an anchor point $b$ does not move until $H$ contracts $\gamma$ to $b$, then $b$ remains on the curve. In this case, we say that the homotopy $H$ *fixes* $b$. If $H$ fixes all of its anchor points, we say that the homotopy is *stable*.

## 3 Homotopy Area & Winding Area of Closed Curves

Let $wn(p, C)$ be the winding number of a closed curve $C$ at a point $p \in \mathbb{R}^2 - [C]$. We define the *winding area* $W(C)$:

$$W(C) = \int_{\mathbb{R}^2 - [C]} |wn(x, C)| dx \ ,$$

where $dx$ is the two-dimensional Lebesgue measure.

Let $C$ and $C'$ be two closed curves and let $H$ be a homotopy between them. We define the homotopy area:

$$Area(H) = \int_{\mathbb{R}^2} E_H(x) \ dx \ .$$

Here, $E_H : \mathbb{R}^2 \to \mathbb{Z}$ is a function, where $E_H(x)$ is the number of connected components of $H^{-1}(x)$, for any $x \in \mathbb{R}^2$. In other words, $E_H$ counts how many times the intermediate curves $H_s$ sweep over $x$.

Let $\mathfrak{H}(C, C')$ be the set of all normal homotopies between $C$ and $C'$. We define the *minimum homotopy area* of $C$ and $C'$, denoted as $\sigma(C, C')$:

$$\sigma(C, C') = \inf_{H \in \mathfrak{H}(C, C')} Area(H)$$

If $p_0 : [0,1] \to \mathbb{R}^2$ is the constant path, i.e. $p_0(t) = C(0)$ for all $t \in [0,1]$, we denote $\sigma(C) = \sigma(C, p_0)$. A *minimum homotopy* $H$ is a normal homotopy such that $Area(H) = \sigma(C)$.

The relation between the homotopy area and winding area can be observed by the following lemma, originally proven in [1].

This is an abstract of a presentation given at CG:YRF 2016. It has been made public for the benefit of the community and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear in a conference with formal proceedings and/or in a journal.

**Lemma 1 (Homotopy Area Inequality)** *For any closed curve $C : [0,1] \longrightarrow \mathbb{R}^2$, we have $\sigma(C) \geq W(C)$.*

A consequence of Lemma 1 is the following corollary:

**Corollary 2 (Minimum Area Realization)** *If there exists a homotopy $H$ between $C$ and $p_0$ such that $Area(H) = W(C)$, then $\sigma(C) = Area(H) = W(C)$.*

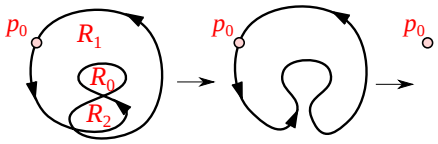



Figure 1: A self-overlapping curve with a minimum homotopy; $\sigma(C) = Area(H) = 2Area(R_2) + Area(R_1)$.

## 4 Self-Overlapping Curves

A normal curve $C : [0,1] \to \mathbb{R}^2$ is called *self-overlapping*, if there exists an immersion of the disk $F : D^2 \to \mathbb{R}^2$ such that $F|_{\partial D^2} = [C]$. Figures 1 and 2 show examples. Detecting whether a given normal curve is self-overlapping can be done in polynomial time [4].

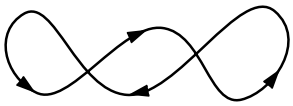


Figure 2: A curve that is not self-overlapping. This curve is not the projection of the boundary of a disk embedded in $\mathbb{R}^3$.

Let $M$ be a surface and let $F : M \to \mathbb{R}^2$ be an immersion. We define the *thickness* of $F$ at point $x \in \mathbb{R}^2$ as the cardinality of the set $F^{-1}(x)$ which is always a finite number. A *lifting* of the immersion $F$ is a smooth embedding $E : M \to \mathbb{R}^3$ that projects $F$, that is, for each point $p \in M$, we have $F(p) = \pi_z(E(p))$, where $\pi_z(x,y,z) = (x,y)$. We obtain the following theorem.

**Theorem 3 (Self-Overlapping Curve Homotopy)** *If $C$ is a self-overlapping curve, then $\sigma(C) = W(C)$.*

**Proof.** Let $F : D^2 \to \mathbb{R}^2$ be an immersion such that $F|_{\partial D^2} = [C]$. Consider a lifting $E : D^2 \to \mathbb{R}^3$ of $F$. Define $\gamma = E|_{\partial D^2}$. Hence, $\gamma$ is a Jordan curve in $\mathbb{R}^3$ with an interior $E|D^2$ homeomorphic to the open disk $D^2$ in $\mathbb{R}^2$. A standard homotopy can be defined for Jordan curves, and by projecting this homotopy to $\mathbb{R}^2$ we get a homotopy $H$ between $C$ and $C(0)$ such that for each point $p \in \mathbb{R}^2$, $E_H(p)$ is equal to the thickness of $F$ at $p$. But, the thickness at each point is also equal to the winding number of the curve at that point [2]. In other words, for each $p \in \mathbb{R}^2$ we have $E_H(p) = wn(p, C)$ and $Area(H) = W(C)$. By Corollary 2, $H$ is a minimum homotopy and $\sigma(C) = W(C)$ as desired. ∎

Let $C$ be a normal curve and $p$ be an intersection point. Assume $C^{-1}(p) = \{t_1, t_2\}$ where $t_1 < t_2$, and let $\gamma_p = C|_{[t_1,t_2]}$. We call $p$ an *anchor point* of $C$ if $\gamma_p$ is a self-overlapping curve.

If a homotopy $H$ is stable, then the set of anchor points of $H$ is contained in the set of anchor points of $C$. In this work we show that a stable minimum homotopy exists. Furthermore, to find the minimum homotopy, we divide the curves into simple subcurves, that are self-overlapping and calculate all possible areas obtained this way. The minimum homotopy will be the one that has the minimum area among the homotopies obtained from this construction.

**Theorem 4 (Minimum Area Homotopy)** *Let $C$ be a normal curve. Then there exists a stable minimum homotopy $H$. Furthermore, $H$ defines a sequence of curves $C = C_0 \to C_1 \to \ldots \to C_m = p_0$ such that each $C_i \to C_{i+1}$ is a contraction of a self-overlapping subcurve of $C_i$ based at an intersection point of $C_i$.*

A proof of the theorem can be sketched as follows: We show that for each normal curve there exists a minimum area homotopy that does not require $\mathrm{I}_b$ moves, and any $\mathrm{II}_b$ move does not create anchor points. Furthermore, if these homotopies are carefully constructed, the anchor points will be a subset of the simple crossing points of the curve. And since a minimum homotopy is locally sense-preserving, these anchor points define self-overlapping pieces of the curve.

Testing all possible sequences of intersection points as anchor points yields an exponential-time algorithm. We are currently working on using dynamic programming to obtain a polynomial-time algorithm.

## References

[1] Chambers, E., and Wang, Y. Measuring similarity between curves on 2-manifolds via homotopy area. In *SoCG* (2013), pp. 425–434.

[2] Chinn, W., and Steenrood, N. *First Concepts of Topology*. MAA, 1966.

[3] Erickson, J. Generic and regular curves. Lecture notes for CS 589 at UIUC, Chapter 5, 2013.

[4] Shor, P., and Van Wyk, C. Detecting and decomposing self-overlapping curves. *CGTA 2*, 1 (1992), 31–50.

[5] Whitney, H. On regular closed curves in the plane. *Compositio Math. 4* (1937), 267–284.