Geometric multicut

Abrahamsen, Mikkel; Giannopoulos, Panos; Löffler, Maarten; Rote, Günter

Published in:
46th International Colloquium on Automata, Languages, and Programming, ICALP 2019

DOI:
10.4230/LIPIcs.ICALP.2019.9

Publication date:
2019

Document version
Publisher's PDF, also known as Version of record

Citation for published version (APA):
Abrahamsen, M., Giannopoulos, P., Löffler, M., & Rote, G. (2019). Geometric multicut. In C. Baier, I. Chatzigiannakis, P. Flocchini, & S. Leonardi (Eds.), *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019* [9] Schloss Dagstuhl- Leibniz-Zentrum fur Informatik GmbH, Dagstuhl Publishing. Leibniz International Proceedings in Informatics, LIPIcs, Vol.. 132 https://doi.org/10.4230/LIPIcs.ICALP.2019.9

Geometric Multicut

Mikkel Abrahamsen
BARC, University of Copenhagen, Universitetsparken 1, DK-2100 Copenhagen Ø, Denmark
miab@di.ku.dk

Panos Giannopoulos
giCenter, Department of Computer Science, City University of London, EC1V 0HB, London, UK
panos.giannopoulos@city.ac.uk

Maarten Löffler
Department of Information and Computing Sciences, Utrecht University, The Netherlands
m.loffler@uu.nl

Günter Rote
Institut für Informatik, Freie Universität Berlin, Takustraße 9, 14195 Berlin, Germany
rote@inf.fu-berlin.de

Abstract

We study the following separation problem: Given a collection of colored objects in the plane, compute a shortest "fence" F, i.e., a union of curves of minimum total length, that separates every two objects of different colors. Two objects are separated if F contains a simple closed curve that has one object in the interior and the other in the exterior. We refer to the problem as GEOMETRIC k-CUT, where k is the number of different colors, as it can be seen as a geometric analogue to the well-studied multicut problem on graphs. We first give an $O(n^4 \log^3 n)$-time algorithm that computes an optimal fence for the case where the input consists of polygons of two colors and n corners in total. We then show that the problem is NP-hard for the case of three colors. Finally, we give a $(2 - 4/3k)$-approximation algorithm.

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

Keywords and phrases multicut, clustering, Steiner tree

Digital Object Identifier 10.4230/LIPIcs.ICALP.2019.9

Category Track A: Algorithms, Complexity and Games

Related Version A full version of the paper is available at `https://arxiv.org/abs/1902.04045`.

Funding *Mikkel Abrahamsen*: Supported by the Innovation Fund Denmark through the DABAI project. MA is also a part of BARC, Basic Algorithms Research Copenhagen, supported by the VILLUM Foundation grant 16582.
Maarten Löffler: Partially supported by the Netherlands Organisation for Scientific Research (NWO); 614.001.504.

Acknowledgements This work was initiated at the workshop on *Fixed-Parameter Computational Geometry* at the Lorentz Center in Leiden in May 2018. We thank the organizers and the Lorentz Center for a nice workshop and Michael Hoffmann for useful discussions during the workshop.

1 Introduction

Problem Definition. We are given k pairwise interior-disjoint, not necessarily connected, sets $B_1, B_2, \ldots, B_k$ in the plane. We want to find a covering of the plane $\mathbb{R}^2 = \bar{B}_1 \cup \bar{B}_2 \cup \cdots \cup \bar{B}_k$ such that the sets $\bar{B}_i$ are closed and interior-disjoint, $B_i \subseteq \bar{B}_i$ and the total length of the boundary $F = \bigcup_{i=1}^{k} \partial \bar{B}_i$ between the different sets $\bar{B}_i$ is minimized.

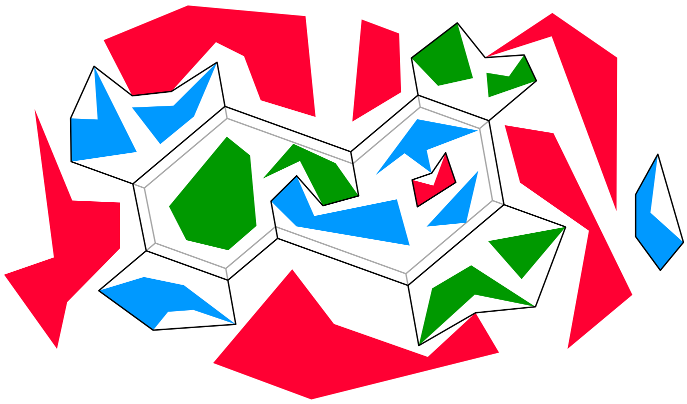

Figure 1 An instance of GEOMETRIC 3-CUT and an optimal fence in black. The fence contains a cycle that does not touch any object. The grey fence shows how the cycle can be shrunk without changing the total length of the fence.

We think of the k sets B_i as having k different *colors* and each set B_i as a union of simple geometric objects like circular disks and simple polygons. An example is shown in Figure 1. We call $\bar{B}_i$ the *territory* of color i. The "fence" F is the set of points that separates the territories. (Alternatively, F is the set of points belonging to more than one territory.) As we can see, a territory can have more than one connected component.

An alternative view of the problem concentrates on the *fence*: A fence is defined as a union of curves F such that each connected component of $\mathbb{R}^2 \setminus F$ intersects at most one set B_i. An interior-disjoint covering as defined above gives, by definition, such a fence. Likewise, a fence F induces such a covering, by assigning each connected component of $\mathbb{R}^2 \setminus F$ to an appropriate territory $\bar{B}_i$. The total length of a fence F is also called the *cost* of F and is denoted as $|F|$.

In our paper, we will focus on the case where the input consists of simple polygons (with disjoint interiors). We refer to this problem as *GEOMETRIC k-CUT*. Each input polygon is called an *object*. We use n to denote the total number of corners of the input polygons, counted with multiplicity.

Even in this simple setting, the problem poses both geometric and combinatorial difficulties. A set B_i can consist of disconnected pieces, and the combinatorial challenge is to choose which of the pieces should be grouped into the same component of $\bar{B}_i$. The geometric task is to construct a network of curves that surrounds the given groups of objects and thus separates the groups from each other. For $k = 2$ colors, optimal fences consist of geodesic curves around obstacles, which are well understood. As soon as the number k of colors exceeds 2, the geometry becomes more complicated, and the problem acquires traits of the geometric Steiner tree problem, as shown by the example in Figure 1.

The problem of enclosing a set of objects by a shortest system of fences has been considered with a single set B_1 by Abrahamsen et al. [1]. The task is to "enclose" the components of B_1 by a shortest system of fences. This can be formulated as a special case of our problem with $k = 2$ colors: We add an additional set B_2, far away from B_1 and large enough so that it is never optimal to enclose B_2. Thus, we have to enclose all components of B_1 and separate them from the unbounded region. In this setting, there will be no nested fences. Abrahamsen et al. gave an algorithm with running time $O(n \text{ polylog } n)$ for the case where the input consists of n unit disks.

Applications. Besides being a natural problem in its own right, the geometric multicut problem may well find applications in image processing and computer vision. As we describe in Section 3, a problem closely related to the case $k = 2$ has been studied from the perspective of image segmentation. Simplified slightly, we are given a picture with some pixels known to be black or white, and we have to choose colors for the remaining pixels so as to minimize the boundary between black and white regions. The problem for $k > 2$ is equally well-motivated in this context, although we have not found any explicit references to it (perhaps because of the NP-hardness that we will prove in this case).

Our Results. In Section 3, we show how to solve the case with $k = 2$ colors in time $O(n^4 \log^3 n)$. The algorithm works by reducing the problem to the multiple-source multiple-sink maximum flow problem in a planar graph. In Section 4, we show that already the case with $k = 3$ colors is NP-hard by a reduction from PLANAR POSITIVE 1-IN-3-SAT.

In Section 5, we discuss approximation algorithms. We first compare the optimal fence $F_{\mathcal{A}}$ consisting of line segments between corners of input polygons to the unrestricted optimal fence F^*. We show that $|F_{\mathcal{A}}| \leq 4/3 \cdot |F^*|$. After applying a $(3/2 - 1/k)$-approximation algorithm for the k-terminal multiway cut problem [6], we obtain a polynomial-time $(2 - \frac{4}{3k})$-approximation algorithm for GEOMETRIC k-CUT (Theorem 11).

Due to restricted space, many details and proofs have been removed and can be found in the full version [2].

2 Structure of Optimal Fences

▶ **Lemma 1.** *An optimal fence F^* is a union of (not necessarily disjoint) closed curves, disjoint from the interior of the objects. Furthermore, F^* is the union of straight line segments of positive length. Consider two non-collinear line segments $\ell_1, \ell_2 \subset F^*$ with a common endpoint p. If p is not a corner of an object, then exactly three line segments meet at p and form angles of $2\pi/3$.*

Proof. It is clear that an optimal fence F^* never enters the interior of an object.

We next show that F^* is the union of a set of closed curves. Suppose not. Let $F' \subset F^*$ be the union of all closed curves contained in F^* and let π be a connected component in $F^* \setminus F'$. Then π is the (not necessarily disjoint) union of a set of open curves, which do not contribute to the separation of any objects. Hence, $F^* \setminus \pi$ is a fence of smaller length than F^*, so F^* is not optimal.

In a similar way, one can consider the union L of all line segments of positive length contained in F^*, and if $F^* \setminus L$ is non-empty, a curve π in $F^* \setminus L$ can be replaced by a shortest path homotopic to it, which consists of a sequence of line segments. (See the proof of Lemma 13 in the full version.)

The last claimed property is shared with the Euclidean Steiner minimal tree on a set of points in the plane, and it can be proved in the same easy way by local optimality arguments, see for example Gilbert and Pollak [10]. ◀

As it can be seen in Figure 1, optimal fences may contain cycles that do not touch any object. As is also indicated in the figure, such a cycle can be shrunk until it eventually hits an object and is eliminated. This does not increase the length, so there is always an optimal fence with no cycle disjoint from all objects. See the full version for the details.

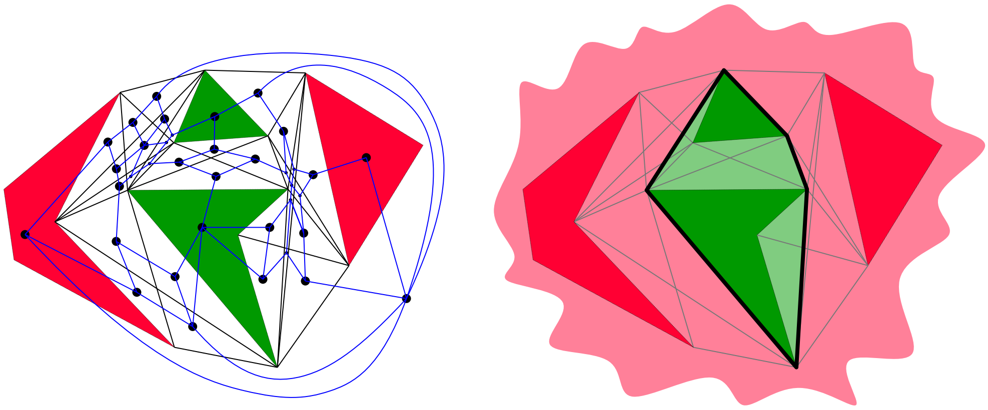

Figure 2 Left: The arrangement $\mathcal{A}$ induced by an instance of GEOMETRIC 2-CUT with two green and two red objects. The edges of the dual graph G are blue. Right: The optimal solution.

3 The Bicolored Case

In this section we consider the case of $k = 2$ different colors. Let N be the set of all corners of the objects. A line segment is said to be *free* if it is disjoint from the interior of every object. A vertex v of an optimal fence cannot have degree 3 or more unless $v \in N$, as otherwise two of the regions meeting at v would be part of the same territory and could be merged, thus reducing the length. We therefore get the following consequence of Lemma 1.

▶ **Lemma 2.** *An optimal fence consists of free line segments with endpoints in N.* ◀

Let S be the set of all free segments with endpoints in N. S includes all edges of the objects. Let $\mathcal{A}$ be the arrangement induced by S, see Figure 2. Consider an optimal fence F^* and the associated territories $\bar{B}_1$ and $\bar{B}_2$. Lemma 2 implies that F^* is contained in $\mathcal{A}$. Thus, each cell of $\mathcal{A}$ belongs entirely either to $\bar{B}_1$ or $\bar{B}_2$. The objects are cells of $\mathcal{A}$ whose classification (i.e., membership of $\bar{B}_1$ versus $\bar{B}_2$) is fixed. In order to find F^*, we need to select the territory that each of the other cells belongs to. Since $|S| = O(n^2)$, $\mathcal{A}$ has size $O(|S|^2) = O(n^4)$ and can be computed in $O(|\mathcal{A}|) = O(n^4)$ time [7]. For simplicity, we stick with the worst-case bounds. In practice, set S can be pruned by observing that the edges of an optimal fence must be *bitangents* that touch the objects in a certain way, because the curves of the fence are locally shortest.

Finding an optimal fence amounts to minimizing the boundary between $\bar{B}_1$ and $\bar{B}_2$. This can be formulated as a minimum-cut problem in the dual graph $G(V, E)$ of the arrangement $\mathcal{A}$. There is a node in V for each cell and a weighted edge in E for each pair of adjacent cells: the weight of the edge is the length of the cells' common boundary. Let $S_1, S_2 \subset V$ be the sets of cells that contain the objects of B_1, B_2, respectively. We need to find the minimum cut that separates S_1 from S_2. This can be obtained by finding the maximum flow in G from the sources S_1 to the sinks S_2, where the capacities are the weights. As G is a planar graph, we can use the algorithm by Borradaile et al. [5] with running time $O(|V| \log^3 |V|)$. The running time has since then been improved to $O(\frac{|V| \log^3 |V|}{\log^2 \log |V|})$ [9]. As $|V| = O(|S|^2) = O(n^4)$, we obtain the following theorem.

▶ **Theorem 3.** *GEOMETRIC 2-CUT can be solved in time $O(\frac{n^4 \log^3 n}{\log^2 \log n})$, where n is the total number of corners of the objects.*

A similar algorithm has been described before in a slightly different context: image segmentation [11], see also [5]. Here, we have a rectangular grid of pixels, each having a given gray-scale value. Some pixels are known to be either black or white. The remaining pixels have to be assigned either the black or the white color. Each pixel has edges to its (at most four) neighbors. The weights of these edges can be chosen in such a way that the minimum cut problem corresponds to minimizing a cost function consisting of two parts: One part, the *data component*, has a term for each pixel, and it measures the discrepancy between the gray-value of the pixel and the assigned value. The other part, the *smoothing component*, penalizes neighboring pixels with similar gray-values that are assigned different colors.

4 Hardness of the Tricolored Case

We show how to construct an instance I of GEOMETRIC 3-CUT from an instance Φ of PLANAR POSITIVE 1-IN-3-SAT. For ease of presentation, we first describe the reduction geometrically, allowing irrational coordinates. We prove that if Φ is satisfiable, then I has a fence of cost M^*, whereas if Φ is not satisfiable, then the cost is at least $M^* + 1/50$. We then argue that the corners can be slightly moved to make a new instance I' with rational coordinates while still being able to distinguish whether Φ is satisfiable or not, based on the cost of an optimal fence.

In order to make the proof as simple as possible, we introduce a new specialized problem COLORED TRIGRID POSITIVE 1-IN-3-SAT in the following.

4.1 Auxiliary NP-complete problems

▶ **Definition 4.** *In the POSITIVE 1-IN-3-SAT problem, we are given a collection Φ of clauses containing exactly three distinct variables (none of which are negated). The problem is to decide whether there exists an assignment of truth values to the variables of Φ such that exactly one variable in each clause is true.*

▶ **Definition 5.** *In the TRIGRID POSITIVE 1-IN-3-SAT problem, we are given an instance Φ of POSITIVE 1-IN-3-SAT together with a planar embedding of an associated graph $G(\Phi)$ with the following properties:*

- *$G(\Phi)$ is a subgraph of a regular triangular grid,*
- *for each variable x, there is a simple cycle v_x,*
- *for each clause $C = \{x, y, z\}$, there is a path c_C and three vertical paths $\ell_x^C, \ell_y^C, \ell_z^C$ with one endpoint at a vertex of c_C and one at a vertex of each of v_x, v_y, v_z,*
- *except for the described incidences, no edges share a vertex,*
- *all vertices have degree 2 or 3,*
- *any two adjacent edges form an angle of π or $2\pi/3$,*
- *the number of vertices is bounded by a quadratic function of the size of Φ.*

The problem is to decide whether Φ has a satisfying assignment (see Definition 4).

Mulzer and Rote [13] showed that another problem, PLANAR POSITIVE 1-IN-3-SAT, is NP-complete, which is similar but uses a slightly different embedding with axis-parallel segments. It trivially follows that TRIGRID POSITIVE 1-IN-3-SAT is also NP-complete, see Figure 3.

Consider an instance $(\Phi, G(\Phi))$ of TRIGRID POSITIVE 1-IN-3-SAT. There are some vertices of degree three on the cycles v_x corresponding to each variable x in Φ, and these we denote as *branch vertices* of $G(\Phi)$. There is also one vertex of degree three on the path c_C corresponding to each clause C in Φ, which we denote as a *clause vertex*. Except for branch and clause vertices, at most two edges meet at each vertex.

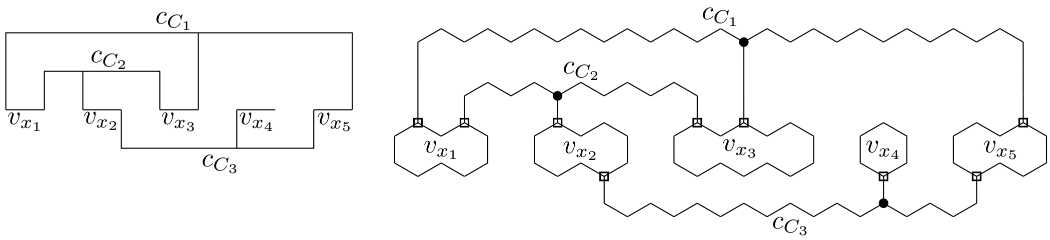

Figure 3 Left: An instance of PLANAR POSITIVE 1-IN-3-SAT for the formula $\Phi = C_1 \wedge C_2 \wedge C_3$ for $C_1 = x_1 \vee x_3 \vee x_5$, $C_2 = x_1 \vee x_2 \vee x_3$, and $C_3 = x_2 \vee x_4 \vee x_5$. Right: A corresponding instance of TRIGRID POSITIVE 1-IN-3-SAT. Clause vertices are drawn as dots and branch vertices as boxes.

Let $\mathcal{C}$ be the set of all clause vertices (considered as geometric points). Removing $\mathcal{C}$ from $G(\Phi)$ (considered as a subset of $\mathbb{R}^2$) splits $G(\Phi)$ into one connected component E_x for each variable x of Φ. The idea of our reduction to GEOMETRIC 3-CUT is to build a *channel* on top of E_x for each variable x. The channel has constant width $1/2$ and contains E_x in the center. The channel contains small *inner* objects and is bounded by larger *outer objects* of another color. There will be two equally good ways to separate the inner and outer objects, namely taking an individual fence around each inner object and taking long fences along the boundaries of the channel that enclose as many inner objects as possible. As it will turn out, any other way of separating the inner from the outer objects will require more fence. These two optimal fences play the roles of x being true and false, respectively.

At the clause vertices where three regions E_x, E_y, E_z meet, we make a clause gadget that connects the three channels corresponding to x, y, z. The objects in the clause gadget can be separated using the least amount of fence if and only if one of the channels is in the state corresponding to true and the other two are in the false state. Therefore, this corresponds to the clause in Φ being satisfied.

In order to make this idea work, we first assign every edge of $G(\Phi)$ an *inner* and an *outer* color among $\{\text{red}, \text{green}, \text{blue}\}$. These will be used as the colors of the inner and outer objects of the channel later on. We require the following of the coloring:

1. The inner and outer colors of any edge are distinct.
2. Any two adjacent collinear edges have the same inner or outer color.
3. Any two adjacent edges that meet at an angle of $2\pi/3$ at a non-clause vertex have the same inner and the same outer color.
4. The inner colors of the three edges meeting at a clause vertex are red, green, blue in clockwise order, while the outer colors of the same edges are blue, red, green, respectively.

We now introduce the problem COLORED TRIGRID POSITIVE 1-IN-3-SAT, which we will reduce to GEOMETRIC 3-CUT, see Figure 4. The problem is NP-complete, as shown in the full version.

▶ **Definition 6.** *In COLORED TRIGRID POSITIVE 1-IN-3-SAT, we are given an instance $(\Phi, G(\Phi))$ of TRIGRID POSITIVE 1-IN-3-SAT together with a coloring of the edges of $G(\Phi)$ satisfying the above requirements. We want to decide whether Φ has a satisfying assignment.*

4.2 Building a GEOMETRIC 3-SAT instance from tiles

Consider an instance $(\Phi, G(\Phi))$ of COLORED TRIGRID POSITIVE 1-IN-3-SAT that we will reduce to GEOMETRIC 3-CUT. We make the construction using hexagonal *tiles* of six different types, namely *straight*, *inner color change*, *outer color change*, *bend*, *branch*, and *clause* tiles. Each tile is a regular hexagon with side length $1/\sqrt{3}$ and hence has width 1. The tiles are rotated such that they have two horizontal edges.

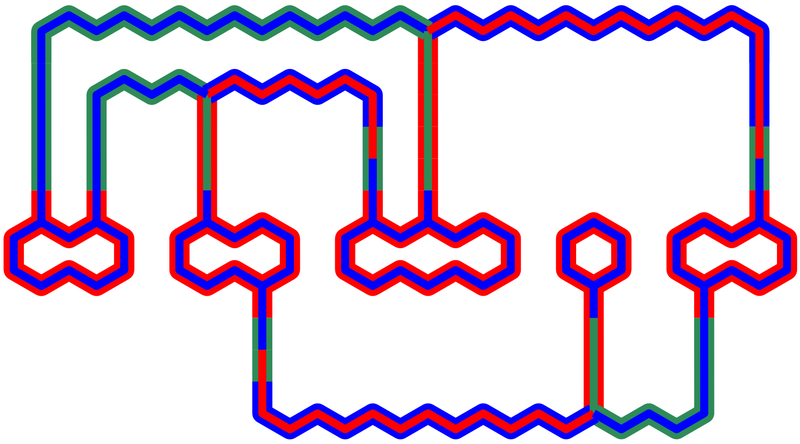

Figure 4 An instance of COLORED TRIGRID POSITIVE 1-IN-3-SAT based on the instance from Figure 3.

The tiles are placed so that each tile is centered at a vertex p of $G(\Phi)$. Let G_p be the part of $G(\Phi)$ within distance $1/2$ from p (recall that each edge of $G(\Phi)$ has length 1). Figure 5 shows the tiles and how they are placed according to the shape and colors of G_p.

In order to define the outer objects of a tile, we consider the straight skeleton offset [3, 4] of G_p at distance $1/4$. With the exception of the bend tile, this offset is the same as the Euclidean offset. By the *outer* and *inner region*, we mean the region of the tile outside, resp. inside, this offset. The outer objects cover the outer region, and every point is colored with the outer color of a closest edge in G_p. The inner region is empty except for the inner objects described in each case below. We suppose that $p = (0, 0)$.

The straight tile. If two collinear edges meet at p with the same inner and outer color, we use a straight tile. Suppose in this and the following two cases that G_p is the vertical line segment from $(0, -1/2)$ to $(0, 1/2)$ – tiles for edges of other slopes are obtained by rotation of the ones described here. There are four axis-parallel squares of the inner color of G_p with side length $1/8$ centered at $(\pm(1/4 - 1/16), \pm 1/4)$. This size is chosen so their total perimeter is 2, which is the length of the common boundary of the inner and outer regions.

The inner color change tile. If two collinear edges meet at p with different inner colors, we use an inner color change tile. There are again four squares colored in the inner color of the closest point in G_p. There are also four smaller axis-parallel squares with side length $1/28$ centered at $(\pm(1/4 - 1/56), \pm 1/56)$, likewise colored in the inner color of the closest point in G_p. The size of these small squares is chosen so that they can be individually enclosed using fences of total length $14 \cdot 1/28 = 1/2$, which is the width of the inner region.

The outer color change tile. If two collinear edges meet at p with different outer colors, we use an outer color change tile. There are four axis-parallel squares of the inner color of G_p with side length $3/32$. Their centers are $(\pm(1/4 - 3/64), \pm 1/4)$. The size of these squares is chosen so that their total perimeter is $2 - 1/2 = 3/2$.

The bend tile. If two non-collinear edges meet at p, we use a bend tile. Consider the case where G_p is the vertical line segment from p to $(0, 1/2)$ and the segment of length $1/2$ from p with direction $(\cos \pi/6, -\sin \pi/6)$. The other cases are obtained by a suitable rotation of this

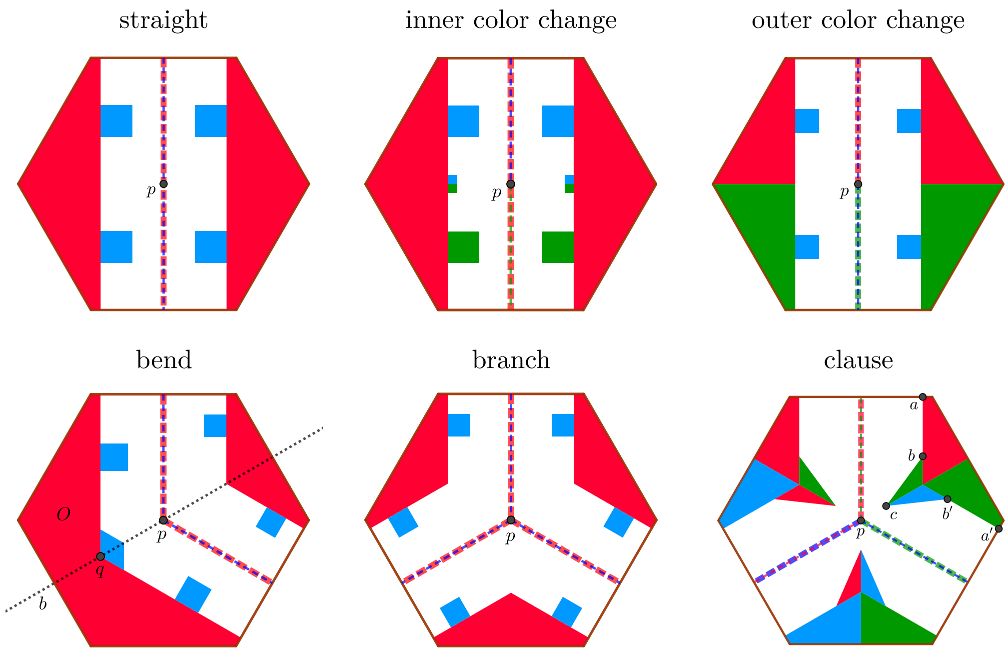

Figure 5 Different kinds of tiles used in the reduction to GEOMETRIC 3-CUT. The dashed colored segments show G_p and the inner and outer color of G_p. The tiles are colored accordingly. The points in the clause tile are defined so that $\|ab\| = \|a'b'\| = 6/25 = 0.24$ and $\|bc\| = \|b'c\| = 1/4 = 0.25$. Point c has coordinates $(x, x/\sqrt{3})$, where $x = \frac{13\sqrt{3}}{200} + 3/16 - \frac{\sqrt{-459+3900\sqrt{3}}}{400}$ is a solution to $10000x^2 + (-1300\sqrt{3} - 3750)x + 507 = 0$. The remaining points in the tile are given by rotations by angles $2\pi/3$ and $4\pi/3$ around p.

tile. There is an axis parallel square of side length $x = \frac{6+\sqrt{3}}{72}$ with center $(-(1/4 - x/2), 1/4)$ and another with side length $y = \frac{6-\sqrt{3}}{48}$ centered at $(1/4 - y/2, 3/8)$. The tile is symmetric with respect to the angular bisector b of G_p, and so the reflections of the described squares with respect to b are also inner objects. Note that there are two outer objects, one of which, O, has a concave corner q with exterior angle $2\pi/3$. We place a parallelogram with side length x, a corner at q, and two edges contained in the edges of O incident at q. It is easy to verify that the common boundary of the inner and outer regions has a total length of 2; the inner objects are chosen such that their total perimeter is also 2.

The branch tile. If p is a branch vertex, we use the branch tile. There are two cases: G_p either contains the vertical segment from p to $(0, 1/2)$ or that from p to $(0, -1/2)$. We specify the tile in the first case – the other can be obtained by a rotation of π. There are axis-parallel squares of side length $y = \frac{6-\sqrt{3}}{48}$ centered at $(\pm(1/4 - y/2), 3/8)$ and their rotations around p by angles $2\pi/3$ and $4\pi/3$. The common boundary of the inner and outer regions has a total length of $\frac{6-\sqrt{3}}{2}$, and the total perimeter of the inner objects is also $\frac{6-\sqrt{3}}{2}$.

The clause tile. If p is a clause vertex, we use the clause tile (defined in Figure 5). The other clause tiles are given by rotations of the described tile by angles $k\pi/3$ for $k = 1, \ldots, 5$.

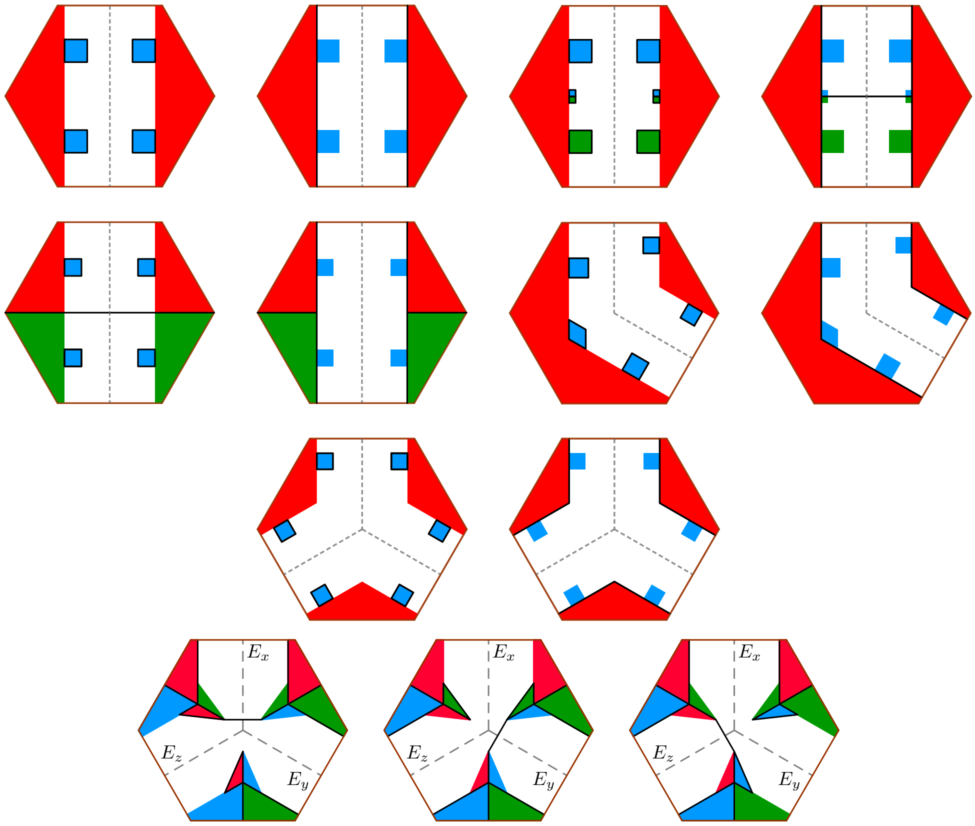

Figure 6 The optimal solutions to each type of tile. The edges in G_p are shown in dashed grey. We denote the left solution of each of the first five types of tiles as the *outer* solution and the other as the *inner* solution. For the clause tile, we define the solution as the *z-outer*, *x-outer*, and *y-outer* solution in order from left to right, respectively.

4.3 Solving the tiles

Let an instance I of GEOMETRIC 3-SAT be given together with an associated fence $\mathcal{F}$. Consider the restriction of I to a convex polygon P and the part of the fence $\mathcal{F} \cap P$ inside P. Note that $\mathcal{F} \cap P$ consists of (not necessarily disjoint) closed curves and open curves with endpoints on the boundary ∂P, such that no two objects in P of different color can be connected by a path $\pi \subset P$ unless π intersects $\mathcal{F}$. (An open curve is a subset of a larger closed curve of $\mathcal{F}$ that continues outside P.) We say that a set of closed and open curves in P with that property is a *solution* to $I \cap P$. In the following, we analyze the solutions to the tiles defined in Section 4.2 in order to characterize the solutions of minimum cost. We say that two closed curves (disjoint from the interiors of the objects) are *homotopic* if one can be continuously deformed into the other without entering the interiors of the objects. Two open curves with endpoints on the boundary of the tile are homotopic if they are subsets of two homotopic closed curves (that extend outside the tile).

The following lemma characterizes the optimal solutions to each type of tile. The statement is that if a solution is not too much more expensive than the solutions shown in Figure 6, then it will contain curves homotopic to each curve in one of the solutions in the figure. The proof is deferred to the full version.

▶ **Lemma 7.** *Figure 6 shows optimal solutions to each kind of tile. The cost in each case is: Straight tile:* 2*. Inner color change tile:* $5/2$*. Outer color change tile:* $\left(\frac{2}{\sqrt{3}} - \frac{1}{2}\right) + 2 \approx 2.65$*. Bend tile:* 2*. Branch tile:* $\frac{6-\sqrt{3}}{2} \approx 2.13$*. Clause tile:* ≈ 3.51 *(the exact value is complicated due to the coordinates and of no use).*

If the cost of a solution $\mathcal{F}$ to a tile T exceeds the optimum by less than $1/50$, then $\mathcal{F}$ is homotopic to one of the optimal solutions $\mathcal{F}^*$ of T in the following sense: For each curve π^* in $\mathcal{F}^*$, there is a curve π in $\mathcal{F}$ homotopic to π^*. If π is closed, the distance from any point on π to the closest point on π^* is less than $\sqrt{(1/8 + 1/100)^2 - (1/8)^2} < 0.06$. If π is open and π^* has an endpoint f^*, there is a corresponding endpoint f of π with $\|f^* f\| < 1/10$.

▶ **Theorem 8.** *The problem GEOMETRIC* 3*-CUT is NP-hard.*

Proof. Let an instance $(\Phi, G(\Phi))$ of COLORED TRIGRID POSITIVE 1-IN-3-SAT be given and construct the tiles on top of $G(\Phi)$ as described. Let $\mathcal{T}$ be the set of tiles and $\mathcal{A}$ the area that the tiles cover (i.e., $\mathcal{A}$ is a union of the hexagons). We will cover any holes in $\mathcal{A}$ with completely red tiles, and place red tiles all the way along the exterior boundary of $\mathcal{A}$. Let $\mathcal{R}$ be the set of these added red tiles and let I be the resulting instance of GEOMETRIC 3-CUT. It is now trivial how to place the fences in I everywhere except in the interior of $\mathcal{A}$.

Consider a fence $\mathcal{F}$ to the obtained instance with cost M. Let M^* be the sum of the cost of an optimal solution to each tile in $\mathcal{T}$ plus the cost of the fence that must be placed along the boundaries of the added red tiles in $\mathcal{R}$. We claim that if Φ is satisfiable, then a solution realizing the minimum M^* exists. Furthermore, if $M < M^* + 1/50$, then Φ is satisfiable.

Suppose that Φ is satisfiable and fix a satisfying assignment. Consider a clause tile where E_x, E_y, and E_z meet. Now, we choose the v-outer state, where $v \in \{x, y, z\}$ is the variable that is satisfied. For each non-clause tile that covers a part of E_w for a variable w of Φ, we choose the outer state if w is true and the inner otherwise. It is now easy to see that the curves form a fence of the desired cost.

On the other hand, suppose that $M < M^* + 1/50$. It follows that in each tile in $\mathcal{T}$, the cost exceeds the optimum by at most $1/50$. Hence, the solution in each tile is homotopic to one of the optimal states as described in Lemma 7. We now claim that the states of all tiles representing one variable must agree on either the inner or outer state. Consider two adjacent tiles where one is in the inner state. There are open curves with endpoints on the shared edge of the two tiles with a distance of more than $1/2 - 2 \cdot 1/10 = 3/10$. The other tile cannot be in the outer state, because then there would have to be an extra open curve of length at least $3/10$ to connect those endpoints. It follows that the other tile must also be in the inner state. Thus, both tiles are either in the inner or in the outer state, as desired.

We now describe how to obtain a satisfying assignment of Φ. Consider a clause tile where E_x, E_y, and E_z meet and suppose the tile is in the x-outer state. It follows from the above that each tile covering E_x is in the outer state or, in the case of the clause tile, in the x-outer state. Similarly, each non-clause tile covering only E_y (resp. E_z) is in the inner state and each clause tile covering a part of E_y (resp. E_z) is not in the y-outer (resp. z-outer) state. We now set x to true and y and z to false and do similarly with the other clause tiles, and it follows that we get a solution to Φ.

The proof that we can avoid the use of irrational corners is deferred to the full version. The basic idea is as follows. For each object O with corner v with an irrational coordinate, we choose a substitute $v' \in O$ with rational coordinates such that $\|vv'\| < \frac{1/50}{4n}$ and such that v' only requires polynomially many bits to represent. This results in a modified instance I', and we prove that I' has a solution of cost $M' := \frac{\lceil 100M^* \rceil}{100}$ if and only if Φ is satisfiable. ◀

5 Approximation

The approach for $k = 2$ from Section 3 does not extend to $k \geq 3$ because Lemma 2 does not apply: The arrangement $\mathcal{A}$ (formed by the free segments between the corners N of the input objects) is no longer guaranteed to contain an optimal fence, see Figure 1. However, we can still search for an approximate solution in $\mathcal{A}$: We show that the optimal fence $F_{\mathcal{A}}$ contained in $\mathcal{A}$ has a cost which is at most $4/3$ times higher than the true optimal fence $F^\star$ (Theorem 9). In the full version, we construct a corresponding lower-bound example with $|F_{\mathcal{A}}| > 1.15 \cdot |F^\star|$.

The graph-theoretic problem that we then have to solve in the weighted dual graph $G = (V, E)$ of $\mathcal{A}$ is the *colored multiterminal cut problem*: We have terminals of $k \geq 3$ different colors and want to make a cut that separates every pair of terminals of different colors. This problem is NP-hard, but we can use approximation algorithms, see Section 5.1.

▶ **Theorem 9.** $|F_{\mathcal{A}}| \leq 4/3 \cdot |F^\star|$.

Proof. From Section 2, we know that after cutting an optimal fence $F^\star$ at all points of N, the remaining components are Steiner minimal trees with leaves in N and internal *Steiner vertices* of degree 3, where three segments make angles of $2\pi/3$.

Consider such a Steiner tree T (Figure 7a). Since T is embedded in the plane, the leaves can be enumerated in cyclic order as $v_1, \ldots, v_m$. We will replace T by a connected system $\bar{T}$ of fences that connects the same set of leaves $v_1, \ldots, v_m$, but contains only segments from the arrangement $\mathcal{A}$. Furthermore, we prove that the total length of $\bar{T}$ is bounded as $|\bar{T}| \leq \frac{4}{3}|T|$. Thus, carrying out this replacement for every Steiner tree leads to the fence $F_{\mathcal{A}}$ of the desired cost. If T consists of a single segment, we define $\bar{T}$ to be the same segment, in which case trivially $|\bar{T}| \leq \frac{4}{3}|T|$. Assume therefore that T has at least one Steiner vertex.

Let T_{ij} be the path in T from v_i to v_j. For each pair $\{i, j\}$, we define the path $\bar{T}_{ij}$ as the shortest path with the properties that

a) $\bar{T}_{ij}$ has endpoints v_i and v_j, and

b) $\bar{T}_{ij}$ is *homotopic* to T_{ij}: this means that T_{ij} can be continuously deformed into $\bar{T}_{ij}$ while keeping the endpoints fixed at v_i and v_j, without entering the interiors of the objects.

It is clear that

c) $\bar{T}_{ij}$ is contained in the arrangement $\mathcal{A}$, and

d) $\bar{T}_{ij}$ is at most as long as T_{ij}.

We will construct $\bar{T}$ as the union of paths $\bar{T}_{ij}$ that are specified by a certain set S of leaf pairs $\{i, j\}$, and we will show that its total length is bounded $|\bar{T}| \leq \frac{4}{3}|T|$. The fact that $F_{\mathcal{A}}$ is a valid fence is ensured by our choice of the set S, which we will now discuss.

If we overlay all paths T_{ij} for $\{i, j\} \in S$, we get a multigraph $\widetilde{T}$, which has the same vertices as T and uses the edges of T, some of them multiple times. We require these three properties:

1. Every edge of T is used once or twice in $\widetilde{T}$.
2. Every Steiner vertex of T has even degree (4 or 6) in $\widetilde{T}$. (By contrast, the degree in T is always 3.)
3. Any two paths T_{ij} and $T_{i'j'}$ that have a point of T in common must *cross* in the following sense: If we assume, by relabeling if necessary, that $i < j$ and $i' < j'$, then $i \leq i' \leq j \leq j'$ or $i' \leq i \leq j' \leq j$.

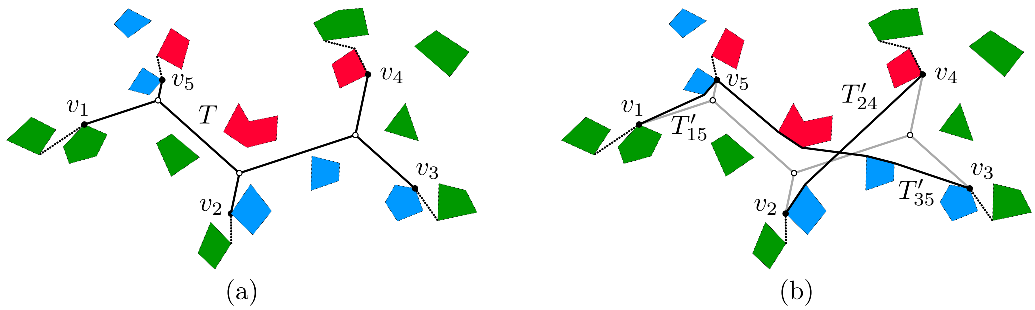

Figure 7 (a) a single Steiner tree T with 5 terminals $v_1, \ldots, v_5$, part of a larger fence system $F^\star$. Steiner vertices are white, leaves are black. (b) The transformed graph $\bar{T}$, formed as the union of three shortest homotopic paths $\bar{T}_{15}$, $\bar{T}_{24}$, and $\bar{T}_{35}$.

The last property is important to ensure that $\bar{T}$ is connected.

As we prove in the full version, Properties 1 and 3 imply that for any two leaves v_i and v_j (where the pair $\{i, j\}$ is not necessarily in S), the set $\bar{T}$ contains a path from v_i to v_j that is homotopic to the path T_{ij}. This means that after replacing T by $\bar{T}$ in $F^\star$, we get a system of fences F' that encloses and separates the same objects as $F^\star$, and thus we have indeed produced a valid fence.

To bound the length of $\bar{T}$, we bound each path $\bar{T}_{ij}$, $\{i, j\} \in S$, by the corresponding path T_{ij} in T. This upper estimate is simply the total length of T plus the length of the duplicated edges of T.

Our first task is to construct the multigraph $\widetilde{T}$. By Property 1, this boils down to selecting which edges of T to duplicate. In order to fulfill Property 2, we require that the degree of every inner vertex of $\widetilde{T}$ becomes even. (We show later that this is sufficient to ensure that the edges of $\widetilde{T}$ can be partitioned into paths T_{ij} subject to Property 3.)

▶ **Lemma 10.** *The edges that should be duplicated can be chosen such that their total length is at most* $|T|/3$.

Proof. For a particular tree, the optimum can be computed easily by dynamic programming, as follows. We root T at some arbitrary leaf. Consider a subtree U rooted at some vertex u of T such that u has one child v in U. We define U_1 and U_2 as the cost of the optimal set of duplicated edges in U, under the constraint that the multiplicity of the edge uv in $\widetilde{T}$ is 1 and 2, respectively.

By induction, we will establish that

$$2U_1 + U_2 \le |U|. \tag{1}$$

This gives $\min\{U_1, U_2\} \le |U|/3$ and proves the lemma, since this also holds for $U = T$. In the base case U has only one edge. Then $U_1 = 0$ and $U_2 = \|uv\| = |U|$, and (1) holds.

If U is larger, v has degree 3, and two subtrees L and R are attached there. If uv is not duplicated, then exactly one of the other edges incident to v has to be duplicated in order for v to get even degree in $\widetilde{T}$. On the other hand, if uv is duplicated, then either both or none of the other edges should be duplicated. Hence, we can compute U_1 and U_2 by the following recursion:

$$U_1 = \min\{L_1 + R_2, L_2 + R_1\} \tag{2}$$

$$U_2 = \min\{L_1 + R_1, L_2 + R_2\} + \|uv\| \tag{3}$$

We therefore get

$$U_1 \le L_2 + R_1 \tag{4}$$
$$U_1 \le L_1 + R_2 \tag{5}$$

from (2) and

$$U_2 \le L_1 + R_1 + \|uv\| \tag{6}$$

from (3).

Adding inequalities (4–6) and using the inductive hypothesis (1) for L and R gives

$$2U_1 + U_2 \le 2L_1 + L_2 + 2R_1 + R_2 + \|uv\| \le |L| + |R| + \|uv\| = |U|. \blacktriangleleft$$

We now have a multigraph $\widetilde{T}$ where every internal vertex has even degree. It follows that the edges of $\widetilde{T}$ can be partitioned into leaf-to-leaf paths, much like when creating an Eulerian tour in a graph where all vertices have even degree.

We still need to satisfy Property 3. Whenever two paths P_1 and P_2 violate this property, we repair this by swapping parts of the paths, without changing the number of remaining violating pairs, as follows: The paths P_1 and P_2 must have a common vertex, and thus also a common edge uv, because the maximum degree in T is 3. Orient P_1 and P_2 so that they use this edge in the direction uv, and cut them at v into $P_1 = Q_1 \cdot R_1$ and $P_2 = Q_2 \cdot R_2$. We now make a cross-over at v, forming the new paths $Q_1 \cdot R_2$ and $Q_2 \cdot R_1$. These new paths satisfy Property 3. To check that we did not create any new violations, we observe that, by Property 1, no other path can use the edge uv, because the capacity of 2 is already taken by P_1 and P_2. Thus, all other paths can either interact with Q_1 and Q_2, or with R_1 and R_2. Thus, swapping the parts of P_1 and P_2 in the other half of the tree T does not affect Property 3.

We have thus established Theorem 9. ◀

5.1 Finding a good fence in $\mathcal{A}$

The problem of finding a small cut in a planar graph $G = (V, E)$ that separates k different classes $T_1, \dots, T_k \subset V$ of terminals was mentioned as a suggestion for future work by Dahlhaus et al. [8], but we have not found any subsequent work on that except for the case $k = 2$ [5]. We can, however, reduce the problem to the multiway cut problem in general graphs (also known as the multiterminal cut problem): For each class T_i, we add an “apex vertex” t_i which is connected to all vertices in T_i by edges of infinite weight. We then ask for the cut of minimum total weight that separates each pair t_i, t_j. Dahlhaus et al. gave a $(2 - 2/k)$-approximation algorithm for the problem. In our setup, the running time will be $O(kn^8 \log n)$. The approximation ratio was since then improved to $3/2 - 1/k$ by Călinescu et al. [6]. Finally, a randomized algorithm with approximation factor 1.3438 was given by Karger et al. [12], who also gave the best known bounds for various specific values of k. Together with Theorem 9, we obtain the following result.

▶ **Theorem 11.** *There is a randomized $4/3 \cdot 1.3438$-approximation algorithm and a deterministic $(2 - \frac{4}{3k})$-approximation algorithm for GEOMETRIC k-CUT, each of which runs in polynomial time.*

6 Concluding Remarks

We have initiated the study of the geometric multicut problem. As our NP-hardness reduction does not imply APX-hardness, an interesting open question is whether there exists a $(1+\varepsilon)$-approximation algorithm for any $\varepsilon > 0$.

There are other versions of the problem that could also be interesting to study. For example, apart from considering shortest paths in the plane, much attention has also been paid to minimum-link paths, i.e., paths connecting two points and consisting of a minimum number of line segments. The analogous problem in our setup is likewise interesting: Compute a simplest possible fence, i.e., one that is the union of as few line segments as possible. The fence can be required to be disjoint from the object interiors, or it can be allowed to pass through the objects, leading to two different problems.

References

1. Mikkel Abrahamsen, Anna Adamaszek, Karl Bringmann, Vincent Cohen-Addad, Mehran Mehr, Eva Rotenberg, Alan Roytman, and Mikkel Thorup. Fast fencing. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing (STOC 2018)*, pages 564–573, 2018. `doi:10.1145/3188745.3188878`.
2. Mikkel Abrahamsen, Panos Giannopoulos, Maarten Löffler, and Günter Rote. Geometric multicut, 2019. `arXiv:1902.04045`.
3. Oswin Aichholzer and Franz Aurenhammer. Straight skeletons for general polygonal figures in the plane. In Jin-Yi Cai and Chak Kuen Wong, editors, *Computing and Combinatorics (COCOON 1996)*, volume 1090 of *Lecture Notes in Computer Science*, pages 117–126. Springer-Verlag, 1996. `doi:10.1007/3-540-61332-3_144`.
4. Oswin Aichholzer and Franz Aurenhammer. Straight skeletons for general polygonal figures in the plane. In A. Samoilenko, editor, *Voronoi's Impact on Modern Sciences, Vol. II*, volume 21 of *Proceedings of the Institute of Mathematics of the National Academy of Sciences of Ukraine*, pages 7–21. Institute of Mathematics of the National Academy of Sciences of Ukraine, Kiew, 1998.
5. Glencora Borradaile, Philip N. Klein, Shay Mozes, Yahav Nussbaum, and Christian Wulff-Nilsen. Multiple-source multiple-sink maximum flow in directed planar graphs in near-linear time. *SIAM Journal on Computing*, 46(4):1280–1303, 2017. `doi:10.1137/15M1042929`.
6. Gruia Călinescu, Howard Karloff, and Yuval Rabani. An improved approximation algorithm for MULTIWAY CUT. *Journal of Computer and System Sciences*, 60(3):564–574, 2000. `doi:10.1006/jcss.1999.1687`.
7. Bernard Chazelle and Herbert Edelsbrunner. An optimal algorithm for intersecting line segments in the plane. *J. ACM*, 39(1):1–54, 1992. `doi:10.1145/147508.147511`.
8. Elias Dahlhaus, David S. Johnson, Christos H. Papadimitriou, Paul D. Seymour, and Mihalis Yannakakis. The complexity of multiterminal cuts. *SIAM Journal on Computing*, 23(4):864–894, 1994. `doi:10.1137/S0097539792225297`.
9. Paweł Gawrychowski and Adam Karczmarz. Improved bounds for shortest paths in dense distance graphs. In Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella, editors, *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*, volume 107 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 61:1–61:15, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum für Informatik. `doi:10.4230/LIPIcs.ICALP.2018.61`.
10. Edgar N. Gilbert and Henry O. Pollak. Steiner minimal trees. *SIAM Journal on Applied Mathematics*, 16(1):1–29, 1968. `doi:10.1137/0116001`.
11. Dorothy M. Greig, Bruce T. Porteous, and Allan H. Seheult. Exact maximum a posteriori estimation for binary images. *Journal of the Royal Statistical Society. Series B (Methodological)*, 51(2):271–279, 1989. `doi:10.1111/j.2517-6161.1989.tb01764.x`.

12 David R. Karger, Philip Klein, Cliff Stein, Mikkel Thorup, and Neal E. Young. Rounding algorithms for a geometric embedding of minimum multiway cut. *Mathematics of Operations Research*, 29(3):436–461, 2004. doi:10.1287/moor.1030.0086.

13 Wolfgang Mulzer and Günter Rote. Minimum-weight triangulation is NP-hard. *Journal of the ACM*, 55:Article 11, 29 pp., May 2008. doi:10.1145/1346330.1346336.