On the Shortest Separating Cycle

Adrian Dumitrescu*

Abstract

According to a result of Arkin et al. (2016), given n point pairs in the plane, there exists a simple polygonal cycle that separates the two points in each pair to different sides; moreover, a $O(\sqrt{n})$-factor approximation with respect to the minimum length can be computed in polynomial time. Here we extend the problem to geometric hypergraphs, and obtain the following characterization of feasibility. Given a geometric hypergraph on points in the plane with hyperedges of size at least 2, there exists a simple polygonal cycle that separates each hyperedge if and only if the hypergraph is 2-colorable.

We extend the $O(\sqrt{n})$-factor approximation in the length measure as follows: Given a geometric graph $G = (V, E)$, a separating cycle (if it exists) can be computed in $O(m + n \log n)$ time, where $|V| = n$, $|E| = m$. Moreover, a $O(\sqrt{n})$-approximation of the shortest separating cycle can be found in polynomial time. Given a geometric graph $G = (V, E)$ in $\mathbb{R}^3$, a separating polyhedron (if it exists) can be found in $O(m + n \log n)$ time, where $|V| = n$, $|E| = m$. Moreover, a $O(n^{2/3})$-approximation of a separating polyhedron of minimum perimeter can be found in polynomial time.

Keywords: Minimum separating cycle, traveling salesman problem, geometric hypergraph, 2-colorability.

1 Introduction

Given a set of n pairs of points in the plane with no common elements, $\{(p_i, q_i) \mid i = 1, \ldots, n\}$, a Shortest Separating Cycle is a plane cycle (a closed curve, a.k.a. *tour*) of minimum length that contains inside exactly one point from each of the n pairs. The problem was introduced by Arkin et al. [3] motivated by applications in data storage and retrieval in a distributed sensor network. They gave a $O(\sqrt{n})$-factor approximation for the general case and better approximations for some special cases. On the other hand, using a reduction from Vertex Cover, they showed that the problem is hard to approximate for a factor of 1.36 unless P = NP, and is hard to approximate for a factor of 2 assuming the Unique Games Conjecture; see, e.g., [21, Ch. 16] for technical background.

*Department of Computer Science, University of Wisconsin–Milwaukee, USA. Email: dumitres@uwm.edu

The assumption that no point appears more than once, i.e., $|\{p_1, \ldots, p_n\} \cup \{q_1, \ldots, q_n\}| = 2n$, is sometimes necessary for the existence of a separating cycle; i.e., there are instances of sets of pairs with common elements and no separating cycle; see for instance Fig. 1 (left). For convenience, points on the boundary of the cycle are considered inside; it is easy to see that requiring points to lie strictly in the interior or also on the boundary are equivalent variants in regards to the existence of a separating cycle. Moreover, the equivalence is almost preserved in the length measure: given any positive $\varepsilon > 0$, and a separating cycle C for n pairs, enclosing $P = \{p_1, \ldots, p_n\}$ (say, after relabeling each pair, if needed), with some of the points of P on its boundary, a separating cycle of length at most $(1 + \varepsilon)\,\mathrm{len}(C)$ can be constructed, having all points of P in its interior.

In this paper we study the extension of the concept of separating cycle to arbitrary graphs and hypergraphs, and to higher dimensions; in the original version introduced by Arkin et al. [3], the input graph is a matching, i.e., it consists of n edges with no common endpoints. Two instances with 8 and respectively 3 point pairs that do not admit separating cycles are illustrated in Fig. 1.

Interestingly enough, even in instances with pairs where a solution exists, one cannot use the algorithm from [3]. Their algorithm (in [3, Subsec. 3.5]) starts by computing a minimum-size square Q containing at least one point from each pair, and then computes a constant-factor approximation of a shortest cycle (tour) of the points contained in Q, in the form of a simple polygon. In the end, this tour is refined to a separating cycle of the given set of point pairs with only a small increase in length. Here we note that there exist instances (like that in Fig. 1) for which there is no separating cycle confined to Q; moreover, the length of a shortest separating cycle can be arbitrarily larger than any function of $\mathrm{diam}(Q)$ and n, and so a new approach is needed for the general version with arbitrary input graphs, or its extension to hypergraphs; i.e., the current $O(\sqrt{n})$-factor approximation does not carry through to these settings.

We first show that a planar geometric graph $G = (V, E)$ admits a separating cycle (for all its edge-pairs) if and only if it is bipartite. This result can be extended to hypergraphs in $\mathbb{R}^d$. Given a geometric hypergraph on points in $\mathbb{R}^d$ with no singleton edges, there exists a simple polyhedron that separates each hyperedge if and only if the hypergraph is 2-colorable.

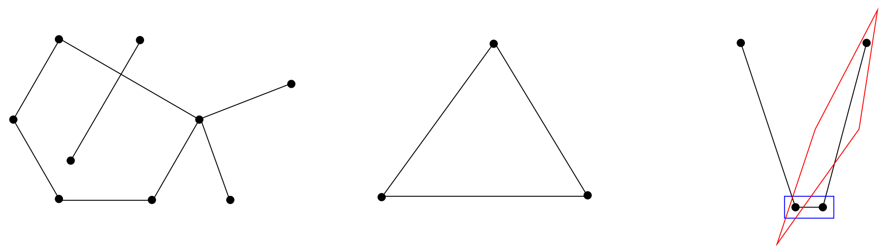

Figure 1: Left and center: instances with no separating cycle. Right: instance where the minimum axis-parallel square (or rectangle) that contains at least one point from each pair does not lead to a solution; a solution is indicated by the red cycle.

Definitions and notations. A *hypergraph* is a pair $H = (V, E)$, where V is finite set of *vertices*, and E is a family of subsets of V, called *edges*. H is said to have *property B*, or be 2-*colorable*, if there is a 2-coloring of V such that no edge is monochromatic; see, e.g., [2, Ch. 1.3].

If C is a (polygonal) cycle, let $\mathring{C}$ and $\overline{C}$ denote the interior and exterior of C, respectively; let ∂C denote its boundary. Consider a geometric hypergraph $H = (V, E)$ on points in the plane with no singleton edges. A polygonal cycle C is said to be a *separating cycle* for H if (i) C is simple; and (ii) each edge of H has points inside C (in its interior or on its boundary) and points in the exterior of C; that is, for each edge $A \in E$, both $A \cap (\mathring{C} \cup \partial C)$ and $A \cap \overline{C}$ are nonempty.

A simple polygonal cycle is said to have *zero area*, if Area$(C) \leq \varepsilon$, for a sufficiently small given $\varepsilon > 0$. Similarly, a polyhedron P is is said to have *zero volume*, if Vol$(P) \leq \varepsilon$, for a sufficiently small given $\varepsilon > 0$.

Preliminaries and related work. Let S be a finite set of points in the plane. According to an old result of Few [11], the length of a minimum spanning path (resp., minimum spanning tree) of any n points in the unit square is at most $\sqrt{2n} + 7/4$ (resp., $\sqrt{n} + 7/4$). Both upper bounds are constructive; for example, the construction of a short spanning path works as follows. Lay out about $\sqrt{n}$ equidistant horizontal lines, and then visit the points layer by layer, with the path alternating directions along the horizontal strips. In particular, the length of the minimum spanning tree of any n points in the unit square is bounded from above by the same expression. An upper bound with a slightly better multiplicative constant for a path was derived by Karloff [18]. L. Fejes Tóth [10] had observed earlier that for n points of a regular hexagonal lattice in the unit square, the length of the minimum spanning path is asymptotically equal to $(4/3)^{1/4}\sqrt{n}$, where $(4/3)^{1/4} = 1.0745\ldots$. As such, the maximum length of the minimum spanning tree of any n points in the unit square is $\Theta(\sqrt{n})$, for a small constant (close to 1). The bound also holds for points in a convex polygon of diameter $O(1)$, in particular for n points in a rectangle of diameter $O(1)$. In every dimension $d \geq 3$, Few showed that the maximum length of a shortest path (or tree) through n points in the unit cube is $\Theta(n^{1-1/d})$; this upper bound is again constructive and extends to rectangular boxes of diameter $O(1)$.

The topic of "separation" has appeared in multiple interpretations; here we only give a few examples: [1, 6, 7, 12, 14, 15, 16]. Some results on watchman tours relying on Few's bounds can be found in [8]; others can be be found in [4]. For instance, in the problem of finding a separating cycle for a given set of segment pairs, that we study here, it is clear that the edges of the cycle must hit all of the given segments. As such, this problem is related to the classic problem of hitting a set of segments by straight lines [15]. Coloring of geometric hypergraphs has been studied, e.g., in [20].

2 Separating Cycles for Graphs and Hypergraphs

By adapting results on hypergraph 2-colorability to a geometric setting, we obtain the following.

Theorem 1 *Let $H = (V, E)$ be a geometric hypergraph on points in the plane with no singleton edges. Then H admits a separating cycle if and only if H is 2-colorable.*

Proof. For the direct implication, assume that C is a separating cycle: then for each $A \in E$, both $A \cap \mathring{C}$ and $A \cap \overline{C}$ are nonempty. Color the points in the interior of C by red and those in its exterior by blue. As such, the hypergraph H is 2-colorable.

We now prove the converse implication. Let $V = R \cup B$ be a partition of the points into red and blue points, such that no edge in E is monochromatic. We construct a simple polygonal cycle containing only the red points in its interior. To this end, we first compute a minimum spanning tree T for the points in R; T is non-crossing [19, Ch. 6], however there could be blue points contained in edges of T. Replace each such edge s with a two-segment polygonal path $\widetilde{s}$ connecting the same pair of points and lying very close to the original segment, and so that $\widetilde{s}$ is not incident to any other point.

The resulting tree, $\widetilde{T}$ is still non-crossing and spans all points in R. By doubling the edges of $\widetilde{T}$ and adding short connection edges, if needed, construct a simple polygonal cycle C of *zero area* that contains it and lies very close to it; as such, C contains all red points and none of the blue points, as required. □

Since hypergraph 2-colorability is NP-complete [13], Theorem 1 yields the following.

Corollary 1 *Given a geometric hypergraph $H = (V, E)$ on points in the plane with no singleton edges, the problem of deciding whether H admits a separating cycle is NP-complete.*

A key fact in our algorithm is the following observation.

Lemma 2 *Let G be connected bipartite graph. Then (apart from a color flip), G admits a unique 2-coloring.*

Proof. Recall that a graph is bipartite if and only if it contains no odd cycle [17, Ch. 3.3]. Consider an arbitrary vertex s and color it red. Then the color of any other vertex, say v, is uniquely determined by the parity of the length of the shortest path from s to v in G: red for even length and blue for odd length. Indeed, the vertices are colored alternately on any path, and since any cycle has odd length, all lengths of paths from s to v have the same parity, as required. □

Let $G = (V, E)$ be the input geometric graph, where $|V| = n$, $|E| = m$, and G has no isolated vertices. Let $G_1, \ldots, G_k$ denote the connected components of G, where $G_i = (V_i, E_i)$, for $i = 1, \ldots, k$.

Theorem 3 (i) *Given a geometric graph $G = (V, E)$, a separating cycle (if it exists) can be computed in $O(m + n \log n)$ time, where $|V| = n$, $|E| = m$.* (ii) *Further, a $O(\sqrt{n})$-approximation of the shortest separating cycle can be found in polynomial time.*

Proof. (i) The graph is first tested for bipartiteness and the input instance is declared infeasible if the test fails (by Theorem 1). This test takes $O(m + n)$ time; see, e.g., [17, Ch. 3.3]. We subsequently assume that G is bipartite, with vertices colored by red and blue. Then the algorithm constructs a plane spanning tree T of the red points (for instance, a minimum spanning tree), and outputs a simple cycle by doubling its edges and avoiding the blue points on its edges by bending those edges as indicated in the proof of Theorem 1. To this end, the following parameters are computed: $\delta_1 > 0$ is the minimum pairwise distance among points in V, found in $O(n \log n)$ time [19, Ch. 5]. For each edge e of T, $\delta_2(e) \geq 0$ is the minimum distance from some blue point to e ($\delta_2(e) = 0$ if e is incident to at least one blue point); and $\delta_3(e) > 0$ is the minimum nonzero distance from a blue point to e ($\delta_3(e) = \infty$ if no blue point is close to e, as described next). The set of values $\delta_2(e), \delta_3(e)$ can be determined using point location for the blue points (as query points) in a planar triangulated subdivision containing the edges of T, all in $O(n \log n)$ time [5, Ch. 6]. The overall time complexity of the algorithm is $O(m + n \log n)$.

(ii) The algorithm above is modified as follows; the first step is the same bipartiteness test. The algorithm 2-colors the vertices in each connected component by red and blue: $V_i = R_i \cup B_i$, for $i = 1, \ldots, k$. By Lemma 2, the 2-coloring of each component is unique (apart from a color flip). The initial coloring of a component may be subsequently subject to a color flip if the algorithm so later decides. Obviously, the coloring of each component is done independently of the others.

Then, the algorithm guesses the diameter of OPT, as determined by one of the $\binom{n}{2}$ pairs of points in V (by trying all such pairs). In each iteration, the algorithm may compute a separating cycle and record its length; the shortest cycle will be output by the algorithm; some iterations may be abandoned earlier, without the need of this calculation.

Consider the iteration in which the guess is correct, with pair $a, b \in V$; we may assume for concreteness that ab is a horizontal segment of unit length; refer to Fig 2. As such, we have that $\text{len(OPT)} \geq 2|ab| = 2$. In this iteration, the algorithm computes a separating cycle whose length is bounded from above by $O(\sqrt{n})$. First, the algorithm computes a rectangle Q of unit width and height $\sqrt{3}$ centered at the midpoint of ab. By the diameter assumption, OPT is contained in Q. In the next step the algorithm computes a separating cycle C containing only red points in Q in its interior (however, the initial coloring of some of the components may be flipped, as needed). By Lemma 2, the coloring of each component is unique (modulo a color flip) and so for each of the components at least one of its color classes is entirely contained in Q. As such, all points in V not contained in Q can be discarded from further consideration.

Each of the components G_i, $i = 1, \ldots, k$ is checked against this containment condition: if a component is found where neither of its two color classes lies in Q, the algorithm abandons this iteration (and assumed diameter pair, $ab = \text{diam(OPT)}$). For each component G_i: (i) if $R_i \subset Q$, then the coloring of this component remains unchanged, regardless of whether $B_i \subset Q$ or $B_i \not\subset Q$. (ii) if $R_i \not\subset Q$ and $B_i \subset Q$, then the coloring of this component is flipped: $R_i \leftrightarrow B_i$, so that $R_i \subset Q$ after the color flip.

Once the recoloring of components is complete, the algorithm computes a minimum spanning tree T of the red points in Q. Its length is bounded from above by the length of the spanning tree computed by Few's algorithm. Since the number of red points does not exceed n, we have $\text{len}(T) = O(\sqrt{n})$. Finally, T is converted

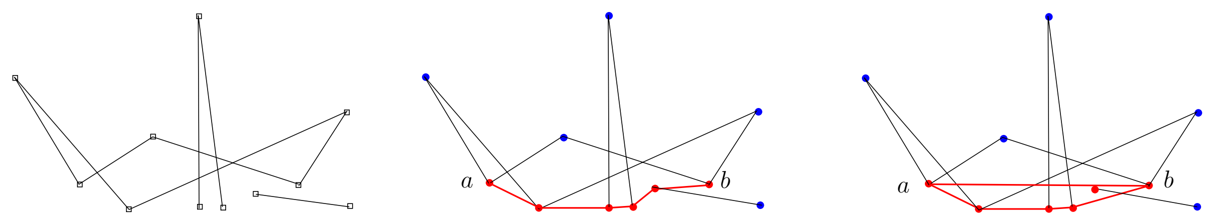

Figure 2: Left: input bipartite graph. Center: a separating cycle can be computed from the MST of the red points (after color flips). Right: a shortest separating cycle.

to a separating cycle C by a factor of at most $2+\varepsilon$ increase in length, for any given $\varepsilon > 0$, as in the proof of part (i). Recalling that $\text{len(OPT)} \geq 2$, it follows that C is a $O(\sqrt{n})$-factor approximation of a shortest separating cycle. □

3 Remarks

1. If the input consists of a set of pairs so that the corresponding graph is bipartite, then by Theorem 1, it admits a separating cycle. (If the corresponding graph is not bipartite, no separating cycle exists.) Similarly, if the input is a 2-colorable hypergraph, it admits a separating cycle. For illustration, we recall some common instances of 2-colorable hypergraphs. A hypergraph $H = (V, E)$ is called k-uniform if all $A \in E$ have $|A| = k$. A random 2-coloring argument gives that any k-uniform hypergraph with fewer than 2^{k-1} edges is 2-colorable [2, Ch. 1.3]; as such, by Theorem 1, it admits a separating cycle. Slightly better bounds have been recently obtained; see [2, Ch. 3.5]. Similarly, let $H = (V, E)$ be a hypergraph in which every edge has size at least k and assume that every edge $A \in E$ intersects at most Δ other edges, i.e., the maximum degree in H is at most Δ. If $e(\Delta+1) \leq 2^{k-1}$ (here $e = \sum_{i=0}^{\infty} 1/i!$ is the base of the natural logarithm), then by the Lovász Local Lemma, H can be 2-colored [2, Ch. 5.2] and so by Theorem 1, it admits a separating cycle; moreover, if a 2-coloring is given, it can be used to obtain a separating cycle. While testing for 2-colorability can be computationally expensive in a general setting (for certain problem instances), it can be always achieved in exponential time; recall that hypergraph 2-colorability is NP-complete [13].

2. Theorem 3 generalizes to 3-dimensional polyhedra. A polyhedron in 3-space is a simply connected solid bounded by piecewise linear 2-dimensional manifolds. The *perimeter* $\text{per}(P)$ of a polyhedron P is the total length of the edges of P (as in [8]).

For part (i), a method similar to that used in the planar case can be used to construct a separating polyhedron in $\mathbb{R}^3$ (or $\mathbb{R}^d$). However, since computing minimum spanning trees in $\mathbb{R}^3$ is more expensive [9, Ch. 9], we employ a slightly different approach. We may assume a coordinate system so that no pair of points have the same x-coordinate. First, the points in V are colored by red or blue as a result of the bipartiteness test, in $O(m+n)$ time. The algorithm then computes a (spanning tree of the red points in the form of a) x-monotone polygonal path P spanning all the red points; this step takes $O(n \log n)$ time. From P, it then obtains a x-monotone polygonal path $\widetilde{P}$ spanning all the red points and not incident to any blue point ($P = \widetilde{P}$ if no blue points are incident to edges of P); $\widetilde{P}$ is constructed in $O(n \log n)$ time.

To this end, P and all blue points are projected onto the xoy plane. Let $\sigma(\cdot)$ denote the projection function. Note that $\sigma(P)$ is x-monotone and that the projection $\sigma(b)$ of a blue point b can be incident to at most one edge of $\sigma(P)$. Checking the projection points $\sigma(b)$ against corresponding edges of $\sigma(P)$ allows for testing whether the original edges of P are incident to the respective blue points. Further, this test allows replacing each such edge s with a two-segment polygonal path $\widetilde{s}$ connecting the same pair of points and lying very close to the original segment, and so that $\widetilde{s}$ is not incident to any other point. Finally the algorithm computes a polyhedron of *zero volume* that contains $\widetilde{P}$; as such, the polyhedron contains all red points but no blue points; this step takes $O(n \log n)$ time. Some details are omitted.

For part (ii), instead of a rectangle based on segment ab as an assumed diameter pair, the algorithm works with a rectangular box where ab is parallel to a side of the box and is incident to its center. The upper bound on the perimeter of the separating polyhedron follows from Few's bound mentioned in the preliminaries.

Theorem 4 (i) *Given a geometric graph $G = (V, E)$ in $\mathbb{R}^3$, a separating polyhedron (if it exists) can be found in $O(m + n \log n)$ time, where $|V| = n$, $|E| = m$.* (ii) *Further, a $O(n^{2/3})$-approximation of a separating polyhedron of minimum perimeter can be found in polynomial time.*

Acknowledgment. The author is grateful to an anonymous reviewer for his careful reading of the manuscript and pertinent remarks.

References

[1] N. Alon, Z. Füredi, and M. Katchalski, Separating pairs of points by standard boxes, *European Journal of Combinatorics* **6(3)** (1985), 205–210.

[2] N. Alon and J. Spencer, *The Probabilistic Method*, 4th edition, Wiley, Tel Aviv and New York, 2015.

[3] E. Arkin, J. Gao, A. Hesterberg, J. S. B. Mitchell, and J. Zeng, The shortest separating cycle problem, *Proc. 14th International Workshop on Approximation and Online Algorithms* (WAOA 2016), vol. 10138 of Lecture Notes in Computer Science, 2016, pp. 1–13.

[4] E. M. Arkin, J. S. B. Mitchell and C. D. Piatko, Minimum-link watchman tours, *Information Processing Letters* **86** (2003), 203–207.

[5] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars, *Computational Geometry: Algorithms and Applications*, 3rd edition, Springer, 2008.

[6] S. Cabello and P. Giannopoulos, The complexity of separating points in the plane, *Algorithmica* **74(2)** (2016), 643–663.

[7] G. Călinescu, A. Dumitrescu, H. Karloff, and P.-J. Wan, Separating points by axis-parallel lines, *International Journal of Computational Geometry & Applications* **15(6)** (2005), 575–590.

[8] A. Dumitrescu and C. D. Tóth, Watchman tours for polygons with holes, *Computational Geometry: Theory and Applications*, **45** (2012), 326–333.

[9] D. Eppstein, Spanning trees and spanners, in J.-R. Sack and J. Urrutia (editors), *Handbook of Computational Geometry*, pages 425–461, Elsevier Science, Amsterdam, 2000.

[10] L. Fejes Tóth, Über einen geometrischen Satz, *Mathematische Zeitschrift* **46** (1940), 83–85.

[11] L. Few, The shortest path and shortest road through n points, *Mathematika* **2** (1955), 141–144.

[12] R. Freimer, J.S.B. Mitchell, and C. Piatko, On the complexity of shattering using arrangements. Extended abstract in *Proc. of the 2nd Canadian Conference on Computational Geometry* (CCCG 1990), 1990, pp. 218–222. Technical Report 91-1197, Dept. of Comp. Sci., Cornell University, April, 1991.

[13] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman and Co., New York, 1979.

[14] D. R. Gaur, T. Ibaraki, and R. Krishnamurti, Constant ratio approximation algorithm for the rectangle stabbing problem and the rectilinear partitioning problem, *Journal of Algorithms* **43** (2002), 138–152.

[15] R. Hassin and N. Megiddo, Approximation algorithms for hitting objects with straight lines, *Discrete Applied Mathematics* **30(1)** (1991), 29–42.

[16] F. Hurtado, M. Noy, P. A. Ramos, and C. Seara, Separating objects in the plane with wedges and strips, *Discrete Applied Mathematics* **109** (2001), 109–138.

[17] D. Jungnickel, *Graphs, Networks and Algorithms*, Springer Verlag, Berlin, 1999.

[18] H. J. Karloff, How long can a Euclidean traveling salesman tour be? *SIAM Journal on Discrete Mathematics* **2** (1989), 91–99.

[19] F. P. Preparata and M. I. Shamos, *Computational Geometry*, Springer-Verlag, New York, 1985.

[20] S. Smorodinsky, On the chromatic number of geometric hypergraphs, *SIAM Journal on Discrete Mathematics* **21(3)** (2007), 676–687.

[21] D. Williamson and D. Shmoys, *The Design of Approximation Algorithms*, Cambridge University Press, 2011.