Minimum Monotone Spanning Trees*

Emilio Di Giacomo[1], Walter Didimo[1], Eleni Katsanou[3], Lena Schlipf[2], Antonios Symvonis[3], Alexander Wolff[4]

[1] Università degli Studi di Perugia, Italy
[2] Universität Tübingen, Tübingen, Germany
[3] National Technical University of Athens, Greece
[4] Universität Würzburg, Würzburg, Germany

Abstract. Computing a Euclidean minimum spanning tree of a set of points is a seminal problem in computational geometry and geometric graph theory. We combine it with another classical problem in graph drawing, namely computing a monotone geometric representation of a given graph. More formally, given a finite set S of points in the plane and a finite set $\mathcal{D}$ of directions, a geometric spanning tree T with vertex set S is *$\mathcal{D}$-monotone* if, for every pair $\{u, v\}$ of vertices of T, there exists a direction $d \in \mathcal{D}$ for which the unique path from u to v in T is monotone with respect to d. We provide a characterization of $\mathcal{D}$-monotone spanning trees. Based on it, we show that a $\mathcal{D}$-monotone spanning tree of minimum length can be computed in polynomial time if the number $k = |\mathcal{D}|$ of directions is fixed, both when (i) the set $\mathcal{D}$ of directions is prescribed and when (ii) the objective is to find a minimum-length $\mathcal{D}$-monotone spanning tree over all sets $\mathcal{D}$ of k directions. For $k = 2$, we describe algorithms that are much faster than those for the general case. Furthermore, in contrast to the classical Euclidean minimum spanning tree, whose vertex degree is at most six, we show that for every even integer k, there exists a point set S_k and a set $\mathcal{D}$ of k directions such that any minimum-length $\mathcal{D}$-monotone spanning tree of S_k has maximum vertex degree $2k$.

1 Introduction

We study a problem that combines the notion of minimum spanning tree of a set of points in the plane with the notion of monotone drawings of graphs.

The problem of computing a (Euclidean) *minimum spanning tree (MST)* of a set of points in the plane is a well-established topic with a long history in computational geometry [27]. An MST of a finite set S of points is a geometric tree T such that: (i) T *spans* S, i.e., the vertices of T are the points of S, and (ii) T has minimum length subject to property (i), where the length of T is the sum of the lengths of its edges and the length of an edge is the Euclidean distance of its endpoints. Equivalently, the MST is the minimum spanning tree of the complete graph on S where the weight of each edge is the Euclidean distance of its incident vertices. It is known that an MST is a subgraph of a Delaunay

* This work was initiated at the GNV Workshop 2022 at Heiligkreuztal.

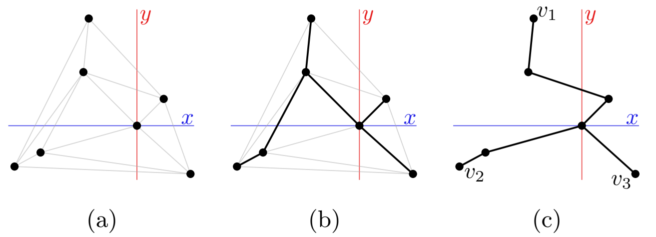


Fig. 1: (a) A point set S with its Delaunay triangulation, (b) MST of S, (c) MMST of S w.r.t. $\{\binom{1}{0}, \binom{0}{1}\}$. The v_2–v_3 path is x-monotone; the v_1–v_2 and v_1–v_3 paths are y-monotone.

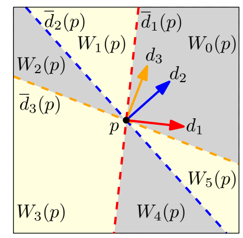


Fig. 2: The set $\mathcal{W}_{\mathcal{D}}(p)$ for the point p and the set $\mathcal{D} = \{d_1, d_2, d_3\}$.

triangulation [49] (see Figs. 1a and 1b). Given a set S of n points, its Delaunay triangulation has at most $3n - 6$ edges, hence an MST of S can be computed in $O(n \log n)$ time (in the real RAM model of computation) via standard MST algorithms. Eppstein [19] has a survey on MSTs.

Monotone drawings of graphs have been introduced by the authors of [4] and have received considerable attention in recent years. They are related to other types of drawings of graphs, such as angle-monotone [10,11,12,15,36], upward [17,24], greedy [3,6,15,16,45,47], self-approaching [1,9,42], and increasing-chord drawings [8,15,39,42]. Computing monotone drawings is also related to the geometric problem of finding monotone trajectories between two given points in the plane avoiding convex obstacles [7]. A plane path is *monotone with respect to a direction d* if the order of its vertices along the path coincides with the order of their projections on a line parallel to d. Any monotone path is necessarily crossing-free [4]. A straight-line drawing of a graph G in the plane is *monotone* if there exists a monotone path (with respect to some direction) between any two vertices of G; the direction of monotonicity may be different for each path. If the directions of monotonicity for the paths are restricted to a set $\mathcal{D}$ of directions, then the drawing is *$\mathcal{D}$-monotone*. Results about monotone drawings include algorithms for different graph classes [2,4,5,20] and the study of the area requirement of such drawings (see [30,33,43] for monotone drawings of trees and [31,32,44] for different classes of planar graphs).

Our setting. In this paper, we study a natural setting that combines the benefits of spanning trees of minimum length with the benefits of monotone drawings. Namely, given a set S of n points in the plane and a prescribed set $\mathcal{D}$ of directions, we study the problem MMST$(S, \mathcal{D})$ of computing a $\mathcal{D}$-monotone spanning tree of S of minimum length (see Fig. 1c). We call such a tree a *minimum $\mathcal{D}$-monotone spanning tree*. For a point set S and an integer $k \geq 1$, we also address the problem MMST(S, k) of computing a minimum *k-directional monotone spanning tree* of S, i.e., a $\mathcal{D}$-monotone spanning tree of minimum length among all possible sets $\mathcal{D}$ of k directions. In this variant, the choice of the directions of monotonicity adjusts to the given point set, which can lead to shorter monotone spanning trees.

We remark that there are other prominent attempts in the literature to couple the MST problem with an additional property. For example, the *Euclidean* *degree-Δ MST* asks for an MST whose maximum degree is bounded by a given integer Δ [21,46]. Seo, Lee, and Lin [48] studied MSTs of smallest diameter or smallest radius. Finding the k smallest spanning trees [18,22,23] or dynamic MSTs [14,50] are further problems related to spanning trees.

Particularly relevant to our study is the *Rooted Monotone MST problem* introduced by Mastakas and Symvonis [40] and further studied by Mastakas [38]. In that problem, given a set S of n points with a designated root $r \in S$, the task is to compute an MST such that the path from r to any other point of S is monotone. Mastakas [37] extended this setting to multiple roots.

Contribution. The main results in this paper are as follows:

- We provide a characterization of $\mathcal{D}$-monotone spanning trees; see Section 4. Based on it, we show how to solve MMST$(S, \mathcal{D})$ in $O(f(|\mathcal{D}|)n^{2|\mathcal{D}|-1} \log n)$ time for some function f of $|\mathcal{D}|$; see Section 5. In other words, MMST$(S, \mathcal{D})$ is in XP (that is, slicewise polynomial) when parameterized by $|\mathcal{D}|$. For $|\mathcal{D}| = 2$, we show how to solve MMST$(S, \mathcal{D})$ in $O(n^2)$ time.
- Regarding MMST(S, k), we describe $O(n^2 \log n)$- and $O(n^6)$-time algorithms for $k = 1$ and $k = 2$, respectively. For $k \geq 3$, we present an XP-algorithm that runs in $O(f(k)n^{2k(2k-1)} \log n)$ time; see Section 5 and Appendix C.
- We show that, in contrast to the MST, whose vertex degree is at most six [46], for every even integer $k \geq 2$, there exists a point set S_k and a set $\mathcal{D}$ of k directions such that any minimum-length $\mathcal{D}$-monotone spanning tree of S_k has maximum vertex degree $2k$; see Section 6.

The proofs of statements with a (clickable) "$\star$" appear in the appendix.

2 Basic Definitions

Let C denote the unit circle centered at the origin o of $\mathbb{R}^2$. Any segment oriented from the center of C to a point of C defines a *direction vector* or simply a *direction*. Two directions are *opposite* if the two segments that define them belong to the same line and lie on opposite sides of the origin. Given a direction d and a set S of points in the plane, we say that S is in *d-general position* if no two points in S lie on a line orthogonal to d. If S is in d-general position, let ord(S, d) be the linear ordering of the orthogonal projections of the points of S on any line parallel to d and directed as d; note that ord(S, d) is uniquely defined. Given a direction d and a point set $S = \{p_1, \ldots, p_n\}$ in d-general position, we say that the geometric path $\langle p_1, \ldots, p_n \rangle$ is *d-monotone* if ord$(S, d) = \langle p_1, \ldots, p_n \rangle$ or ord$(S, d) = \langle p_n, \ldots, p_1 \rangle$; in this case, all projections of the oriented segments $\overrightarrow{p_i p_{i+1}}$ (for $i \in \{1, \ldots, n-1\}$) on a line parallel to d point towards the same direction. A path is *monotone* if it is d-monotone with respect to some direction d.

Let S be a finite set of points, and let $\mathcal{D}$ be a finite set of directions such that no two of them are opposite. A spanning tree T of S is *$\mathcal{D}$-monotone* if, for every pair of vertices $\{u, v\}$ of T, there is a $d \in \mathcal{D}$ such that the unique geometric path from u to v in T is d-monotone (which requires that the subset of points on the path from u to v is in d-general position).

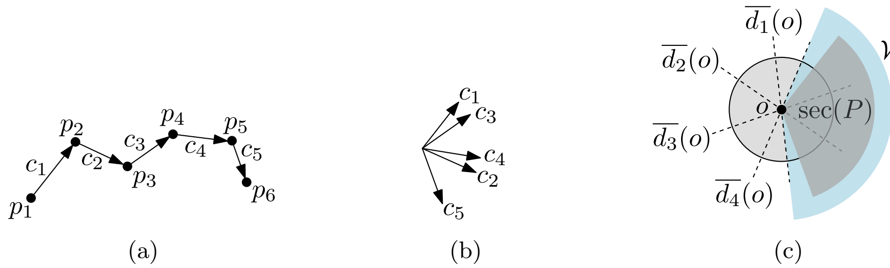


Fig. 3: (a) A directed geometric path P, (b) its sector of directions sec(P) (in dark gray) and (c) the wedge set $\mathcal{W}_P$ of path P (in blue).

A *minimum $\mathcal{D}$-monotone spanning tree* of S is a $\mathcal{D}$-monotone spanning tree of S of minimum length among all $\mathcal{D}$-monotone spanning trees of S; we call MMST$(S, \mathcal{D})$ the problem of computing such a tree. For a positive integer k, we say that a spanning tree T of S is *k-directional monotone* if there exists a set $\mathcal{D}$ of k directions such that T is $\mathcal{D}$-monotone. A *minimum k-directional monotone spanning tree* of S is a k-directional monotone spanning tree of S of minimum length among all k-directional monotone spanning trees of S; we call MMST(S, k) the problem of computing such a tree. To solve this problem, it turns out that it is sufficient to consider only sets $\mathcal{D}$ of directions such that S is in *$\mathcal{D}$-general position*, i.e., S is in d-general position for every $d \in \mathcal{D}$.

Given two points u and v, let $l_{u,v}$ be the line passing through u and v. Given a direction d and a point x, let $d(x)$ be the line parallel to d passing through x and let $\overline{d}$ be the direction orthogonal to d obtained by rotating d counterclockwise (ccw.) by an angle of 90°. Accordingly, $\overline{d(x)}$ is the line orthogonal to $d(x)$ and $\overline{l_{u,v}(x)}$ is the line orthogonal to $l_{u,v}$ passing through x. Given two vertices u and v of a geometric tree T, let $P_{u,v}$ denote the path of T from u to v.

Given a sorted set $\mathcal{D} = \{d_1, d_2, \ldots, d_k\}$ of $k \geq 1$ pairwise non-opposite directions (assumed to be sorted with respect to the directions' slopes) and a point p in the plane, let $\mathcal{W}_\mathcal{D}(p) = \{W_0(p), W_1(p), \ldots, W_{2k-1}(p)\}$ be the set of $2k$ wedges determined by the lines $\overline{d_1}(p), \overline{d_2}(p), \ldots, \overline{d_k}(p)$. See Fig. 2 on page 2 for an illustration where $k = 3$. We fix the numbering of the wedges by starting with an arbitrary wedge $W_0(p)$ and then continue with $W_1(p), W_2(p), \ldots, W_{2k-1}(p)$ in ccw. order around p. Whenever we refer to a wedge $W_i(p)$ for some integer i, we assume that i is taken modulo $2k$. If p coincides with the origin o, we just write $\mathcal{W}_\mathcal{D} = \{W_0, W_1, \ldots, W_{2k-1}\}$ instead of $\mathcal{W}_\mathcal{D}(o) = \{W_0(o), W_1(o), \ldots, W_{2k-1}(o)\}$.

For a directed geometric path $P = \langle p_1, \ldots, p_r \rangle$ and $i \in [r-1]$, let $c_i(P)$ be the oriented segment starting from the origin o that is parallel to and has the same orientation as $\overrightarrow{p_i p_{i+1}}$. Define sec$(P)$, the *sector of directions* of path P, to be the smallest sector of the unit circle that includes the oriented segment $c_i(P)$ for every $i \in [r-1]$; see Figs. 3a and 3b. Moreover, let $\mathcal{W}_P \subseteq \mathcal{W}_\mathcal{D}$ be the *wedge set of the directed path P*, i.e., the smallest set of consecutive wedges in counterclockwise (ccw.) order whose union contains sec(P); see Fig. 3c. For a point p, let $\mathcal{W}_P(p)$ be the region of the plane determined by $\mathcal{W}_P$ translated such

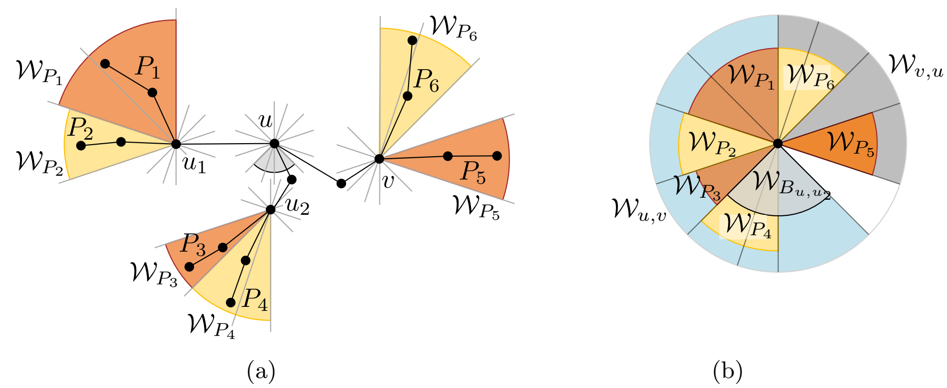


Fig. 4: (a) A monotone tree and its sets of utilized wedges for each leaf path. (b) All sets of utilized wedges drawn on the same unit circle. Set $\mathcal{W}_{u\setminus v}$ (resp. $\mathcal{W}_{v\setminus u}$) consists of all wedges in the blue (resp. gray) region.

that p is its apex. If $\overleftarrow{P}$ is the reverse path of P, then $\mathcal{W}_{\overleftarrow{P}}$ consists of the wedges opposite to those in $\mathcal{W}_P$. We say that path P *utilizes* wedge set $\mathcal{W}_P$.

In a $\mathcal{D}$-monotone spanning tree T, a *branching vertex* is a vertex of degree at least 3 and a *leaf path* is a path of degree-2 vertices from a branching vertex to a leaf. Given two adjacent branching vertices u and v in T, the *branch* $B_{u,v}$ is the unique path that connects u and v via a sequence of degree-2 vertices. Both a leaf path and a branch may consist of a single edge. Further, for any pair of (not necessarily adjacent) vertices u and v, let $T_{u\setminus v}$ be the subtree of T consisting of u and all subtrees hanging from u except for the one containing v. Let $\mathcal{W}_{u\setminus v} \subseteq \mathcal{W}_{\mathcal{D}}$, the wedge set of $T_{u\setminus v}$, be the smallest set of consecutive wedges that contains all wedges utilized by either leaf paths or branches oriented away from u in $T_{u\setminus v}$ and that does not contain the wedge utilized by the edge out of u that leads to vertex v; see Fig. 4. Note that if u and/or v is a leaf, then $\mathcal{W}_{u\setminus v} = \emptyset$ and/or $\mathcal{W}_{v\setminus u} = \emptyset$. Let $\mathcal{W}_{u\setminus v}(u)$ be the region defined by the wedges in $\mathcal{W}_{u\setminus v}$ translated such that u is their apex.

3 Properties of Monotone Paths and Trees

We describe basic properties of monotone paths and $\mathcal{D}$-monotone trees, which we use in Section 4. Unless otherwise stated, we assume that $\mathcal{D}$ consists of pairwise non-opposite directions and that the point set S is always in $\mathcal{D}$-general position.

Lemma 1 ($\star$). *Let S be a set of points, and let $P = \langle u, x, v\rangle$ be a geometric path on S. Let d be a direction such that S is in d-general position. If u and v lie in the same half-plane determined by $\overline{d}(x)$, then the path P is not d-monotone.*

The next lemma generalizes Lemma 1. It concerns the wedges formed by a set of $k > 1$ directions (in contrast to the half-plane formed by the perpendicular to a single direction) and two arbitrary points in the same wedge.

Lemma 2 (⋆). *Let S be a set of points, let T be a spanning tree of S, and let $\mathcal{D}$ be a set of k directions. Let x, u, and v be points in S such that $x \in P_{u,v}$. If u and v lie in the same wedge in $\mathcal{W}_{\mathcal{D}}(x)$, then the path $P_{u,v}$ is not $\mathcal{D}$-monotone.*

For any vertex x of T, the set of lines $\{\overline{d}(x) \colon d \in \mathcal{D}\}$ partitions the plane into $2k$ wedges with apex x, each wedge containing at most one neighbor of x.

Lemma 3 (⋆). *Let S be a set of points, let $\mathcal{D}$ be a set of k directions, and let T be a $\mathcal{D}$-monotone spanning tree of S. Let $\Delta(T)$ denote the maximum degree of tree T. Then, $\Delta(T) \leq 2k$.*

The authors of [4] gave the following characterization.

Lemma 4 ([4]). *Let P be a directed geometric path. Then, P is monotone if and only if the angle of its sector of directions $\sec(P)$ is smaller than π.*

While Lemma 4 can be used to recognize monotone paths, it does not specify a direction of monotonicity. This is rectified by Lemma 5.

Lemma 5 (⋆). *Given a direction d, a monotone directed geometric path P is d-monotone if and only if $\overline{d}(o)$ does not intersect $\sec(P)$, where o is the origin.*

The following corollary is an immediate consequence of Lemma 5.

Corollary 1. *Let S be a set of points, $\mathcal{D}$ be a set of k directions, and T be a $\mathcal{D}$-monotone spanning tree of S. Let P be a directed path in T. Given a direction $d \in \mathcal{D}$, P is d-monotone if and only if $\overline{d}(o)$ does not intersect the interior of $\mathcal{W}_P$, where o is the origin.*

Additional properties concerning paths of $\mathcal{D}$-monotone spanning trees and their corresponding wedge sets are presented in the following lemma.

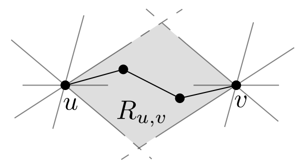
Lemma 6 (⋆). *Let S be a set of points, let $\mathcal{D}$ be a set of k directions, and let T be a $\mathcal{D}$-monotone spanning tree of S. Then, T has the following properties: (i) Let P be a directed path originating at vertex u of T. Then, P lies in $\mathcal{W}_P(u)$. (ii) Let P_1 and P_2 be two edge-disjoint directed paths originating at internal vertices u and v of T and terminating at leaves of T. Then, sets $\mathcal{W}_{P_1}$ and $\mathcal{W}_{P_2}$ are disjoint and regions $\mathcal{W}_{P_1}(u)$ and $\mathcal{W}_{P_2}(v)$ are disjoint.*

Lemma 6 immediately implies the next bound on the number of leaves of $\mathcal{D}$-monotone trees.

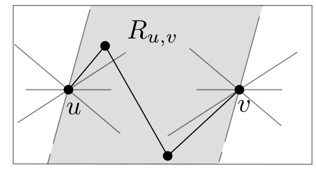
Lemma 7. *Let S be a set of points, and let $\mathcal{D}$ be a set of k directions. If T is a $\mathcal{D}$-monotone spanning tree of S, then T has at most $2k$ leaves.*

The following lemma generalizes Lemma 6 (which concerns paths) for subtrees of a $\mathcal{D}$-monotone spanning tree T.

Lemma 8 (⋆). *Let S be a set of points, let $\mathcal{D}$ be a set of k directions, let T be a $\mathcal{D}$-monotone spanning tree of S, and let u and v be two vertices of T. Then, it holds that: (i) Subtree $T_{u \setminus v}$ of T lies in $\mathcal{W}_{u \setminus v}(u)$. (ii) Sets $\mathcal{W}_{u \setminus v}$ and $\mathcal{W}_{v \setminus u}$ are disjoint, and regions $\mathcal{W}_{u \setminus v}(u)$ and $\mathcal{W}_{v \setminus u}(v)$ are disjoint.*



(a) $R_{u,v}$ is a parallelogram if $|\mathcal{W}_{B_{u,v}}| < k$.



(b) $R_{u,v}$ is a strip if $|\mathcal{W}_{B_{u,v}}| < k$.

Fig. 5: The different shapes of $R_{u,v}$ depending on $|\mathcal{W}_{B_{u,v}}|$.

Let $B_{u,v}$ be a branch of a $\mathcal{D}$-monotone tree T connecting branching vertices u and v. Recall that $|\mathcal{W}_{B_{u,v}}| \leq k$, due to monotonicity of $B_{u,v}$. Let $R_{u,v} = \mathcal{W}_{B_{u,v}}(u) \cap \mathcal{W}_{B_{v,u}}(v)$. If $|\mathcal{W}_{B_{u,v}}| < k$, then $R_{u,v}$ is a parallelogram; see Fig. 5a. Otherwise (i.e., if $|\mathcal{W}_{B_{u,v}}| = k$), $R_{u,v}$ is a strip bounded by the parallel lines $\overline{d}(u)$ and $\overline{d}(v)$, where d is the direction of monotonicity of $B_{u,v}$; see Fig. 5b. We call $R_{u,v}$ the *region of branch* $B_{u,v}$. Similarly, if $P_{u,\lambda}$ is a leaf path from u to λ, then we define the *region of the leaf path* $R_{u,\lambda} = \mathcal{W}_{P_{u,\lambda}}(u) \cap \mathcal{W}_{P_{\lambda,u}}(\lambda)$.

Lemma 9 ($\star$). *Let S be a set of points, let $\mathcal{D}$ be a set of k (pairwise non-opposite) directions such that S is in $\mathcal{D}$-general position, and let T be a $\mathcal{D}$-monotone spanning tree of S. If $P_{u,v}$ is either a branch or a leaf path of T, then $\mathcal{W}_{P_{u,v}} \cap \mathcal{W}_{u\setminus v} = \emptyset$ and $R_{u,v} \cap \mathcal{W}_{u\setminus v}(u) = \emptyset$.*

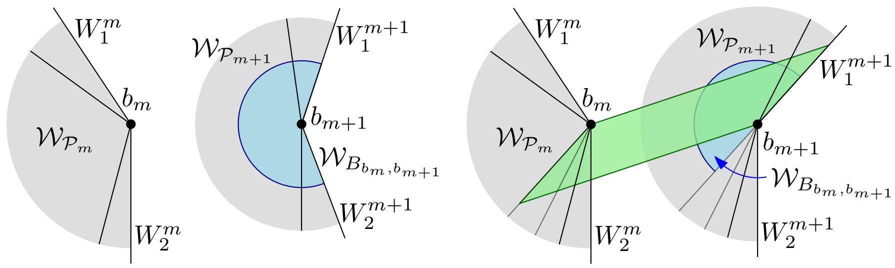
4 A Characterization of $\mathcal{D}$-Monotone Spanning Trees

In this section we provide a characterization of $\mathcal{D}$-monotone spanning trees. It is the basis for our algorithm that solves MMST($S, \mathcal{D}$); see Section 5.

Theorem 1. *Let S be a set of points, let $\mathcal{D}$ be a set of k (pairwise non-opposite) directions such that S is in $\mathcal{D}$-general position, and let T be a spanning tree of S. Then, T is $\mathcal{D}$-monotone if and only if:*

(a) Every leaf path and every branch P in T is $\mathcal{D}$-monotone.
(b) For every two leaf paths P_1 and P_2 incident to branching vertices u and v, respectively, $\mathcal{W}_{P_1}$ and $\mathcal{W}_{P_2}$ are disjoint.
(c) For every branch or leaf path $P_{u,v}$ of T it holds that $R_{u,v} \cap \mathcal{W}_{u\setminus v}(u) = \emptyset$.

Proof. ($\Rightarrow$) Since T is a $\mathcal{D}$-monotone tree, any subtree of T is also $\mathcal{D}$-monotone and, hence statement (a) holds. Statement (b) follows from Lemma 6(ii) since any two leaf paths are edge-disjoint. Statement (c) follows from Lemma 9.
($\Leftarrow$) For the monotonicity of T, it suffices to show that, for any two leaves λ and μ, the path $P_{\lambda,\mu}$ is $\mathcal{D}$-monotone. Let $P_{u,\lambda}$ and $P_{v,\mu}$ be the leaf paths to λ and μ where u and v are the branching vertices they are incident to, respectively. Suppose first that $P_{u,\lambda}$ and $P_{v,\mu}$ are incident to the same vertex, i.e., $u = v$. Due to (a), both leaf paths are $\mathcal{D}$-monotone; hence, $|\mathcal{W}_{P_{u,\lambda}}| \leq k$ and $|\mathcal{W}_{P_{v,\mu}}| \leq k$. Also, due to (b), $\mathcal{W}_{P_{u,\lambda}}$ and $\mathcal{W}_{P_{v,\mu}}$ are disjoint. Hence, there exists a direction d



(a) $W_1^m = W_1^{m+1}$ or $W_2^m = W_2^{m+1}$. (b) $|\mathcal{W}_{\mathcal{P}_{m+1}}| > k$ contradicts condition (c).

Fig. 6: Different cases examined in the proof of Theorem 1

in $\mathcal{D}$ such that $\overline{d}(u)$ separates $\mathcal{W}_{P_{u,\lambda}}(u)$ and $\mathcal{W}_{P_{v,\mu}}(v)$ and does not intersect the interior of either of them. By Corollary 1, $P_{u,\lambda}$ and $P_{v,\mu}$ are both d-monotone and, additionally, they lie in different halfplanes with respect to $\overline{d}(u)$. Hence, the path from λ to μ is d-monotone, and thus $\mathcal{D}$-monotone.

Suppose now that $u \neq v$. Let $\mathcal{B} = \{u = b_1, \ldots, b_r = v, b_{r+1} = \mu\}$ be the sequence of the branching vertices on $P_{\lambda,\mu}$ in order of appearance where, for convenience, μ is treated as a branching vertex. By Corollary 1, it suffices to show that there is a direction d such that $\overline{d}(\mu)$ does not intersect the interior of $\mathcal{W}_{P_{\mu,\lambda}}(\mu)$. Let $\mathcal{P}_i = P_{b_i,\lambda}$ denote the subpath of $P_{\mu,\lambda}$ from vertex b_i to leaf λ. We show by induction on the size of $\mathcal{B}$ that *for every* $i \in \{1, \ldots, r+1\}$, $|\mathcal{W}_{\mathcal{P}_i}| \leq k$. Since $\mathcal{P}_{r+1}$ is by definition the oriented path from μ to λ, the fact that $|\mathcal{W}_{\mathcal{P}_{r+1}}| \leq k$ together with Corollary 1 guarantee that there exists a direction $d \in \mathcal{D}$ such that the path from λ to μ is d-monotone. For the base of the induction, observe that $\mathcal{P}_1$ is the leaf path $P_{u,\lambda}$, which is $\mathcal{D}$-monotone by (a). For the induction hypothesis, assume that $|\mathcal{W}_{\mathcal{P}_i}| \leq k$ for $i \leq m$. We show that $|\mathcal{W}_{\mathcal{P}_{m+1}}| \leq k$. Assume, for a contradiction, that $|\mathcal{W}_{\mathcal{P}_{m+1}}| > k$. Since $\mathcal{P}_{m+1}$ consists of $\mathcal{P}_m$ and of the branch $B_{b_m,b_{m+1}}$, the wedges of $\mathcal{W}_{\mathcal{P}_{m+1}} \setminus \mathcal{W}_{\mathcal{P}_m}$ are due to branch $B_{b_m,b_{m+1}}$. Let W_1^m and W_2^m be the leading and the trailing wedges (in ccw. order) of $\mathcal{W}_{\mathcal{P}_m}$ and let W_1^{m+1} and W_2^{m+1} be the leading and the trailing wedges (in ccw. order) of $\mathcal{W}_{\mathcal{P}_{m+1}}$. Observe first that either $W_1^m = W_1^{m+1}$ or $W_2^m = W_2^{m+1}$. If this was not the case, then $|\mathcal{W}_{B_{b_{m+1},b_m}}| > k$ which contradicts the fact that all branches are $\mathcal{D}$-monotone (refer to Fig. 6a).

Now assume, w.l.o.g., that $W_2^m = W_2^{m+1}$ (see Fig. 6b). The leading wedge of $\mathcal{W}_{B_{b_{m+1},b_m}}$ is W_1^{m+1}, and the branch B_{b_{m+1},b_m} uses at most k wedges as it is $\mathcal{D}$-monotone. Also, $\mathcal{W}_{B_{b_{m+1},b_m}}(b_{m+1})$ contains vertex b_m as otherwise it would not be $\mathcal{D}$-monotone. Consider now the utilized wedge set $\mathcal{W}_{B_{b_m,b_{m+1}}}$ of $B_{b_m,b_{m+1}}$ consisting of the opposite of $\mathcal{W}_{B_{b_{m+1},b_m}}$. Its leading wedge is the opposite of W_1^{m+1}, it is located before W_2^m (in ccw. order), and its trailing wedge is located after W_2^m (in ccw. order). Thus, $\mathcal{W}_{\mathcal{P}_m}(b_m)$ intersects the region of the branch (the green parallelogram in Fig. 6b). This is a contradiction, as $\mathcal{W}_{\mathcal{P}_m}(b_m) \subset \mathcal{W}_{b_m,b_{m+1}}(b_m)$ due to (c). Note that considering μ as a branching vertex does not affect the correctness of the proof. □

5 Algorithms for MMST($S, \mathcal{D}$)

In this section we prove that the problem MMST($S, \mathcal{D}$) is in XP with respect to $|\mathcal{D}|$, that is, it can be solved in polynomial time for any fixed value of $|\mathcal{D}|$. An *embedding* of a tree is prescribed by the clockwise circular order of the edges incident to each vertex of the tree. A tree with a given embedding is an *embedded tree*. A *homeomorphically irreducible tree (HIT)*, is an embedded tree without vertices of degree two [28]. Let T_1 and T_2 be two trees; we say that T_1 and T_2 have the same *topology* if they are (possibly different) subdivisions of the same HIT H. Two trees with the same topology have the same *embedding* if the circular order of the edges around the vertices is the same in both trees. Given a HIT H and any embedded tree T that is a subdivision of H, we say that H *corresponds to* T. Since for a vertex of degree two the circular order of its incident edges is unique, the embedding of a tree T uniquely defines the embedding of the corresponding HIT. Note that, given an embedded tree T and the corresponding HIT H, an internal vertex of H corresponds to a branching vertex of T, a leaf of H to a leaf path of T, and an edge between two internal vertices of H to a branch of T.

Let n_ℓ be the numbers of HITs with at most ℓ leaves. We can use a result of Harary, Robinson, and Schwenk [29] concerning the number of (non-embedded) trees with $2\ell - 2$ vertices to derive a bound for n_ℓ. However, this does not yield an algorithm to generate all different HITs with at most ℓ leaves. For this reason we give an upper bound that is based on a generation scheme. Note that our scheme may generate the same HIT several times.

Lemma 10 ($\star$). *The number of different HITs with at most ℓ leaves is $O(7^\ell \cdot \ell!)$, and these HITs can be enumerated in $O(7^\ell \cdot \ell!)$ time.*

We now present an overview of the algorithm for solving the MMST($S, \mathcal{D}$) problem. It examines every HIT with at most $2k$ leaves. Since there are many ($\mathcal{D}$-monotone) spanning trees that are subdivisions of the same HIT, the algorithm examines for each HIT all of its $\mathcal{D}$-monotone spanning trees on S. Let H be the HIT under consideration, and let ℓ and b be the numbers of leaves and branching vertices of H, respectively. Let M be one of the $O(n^b)$ possible mappings of the b branching vertices to points in S. Let A be an assignment of the wedges of $\mathcal{W}_\mathcal{D}$ to the leaves of H so that each leaf receives a distinct set of consecutive wedges. Assigning (as part of A) the set of consecutive wedges $\mathcal{W}^A$ to a leaf λ incident to a branching vertex v of H can be interpreted as our intention to cover all points in region $\mathcal{W}^A(v)$ by the monotone leaf path P that ends at λ. As shown in Fig. 7, the monotone leaf path may utilize a set of consecutive wedges $\mathcal{W}_P \subseteq \mathcal{W}^A$, i.e., some of the leading and/or trailing wedges of $\mathcal{W}^A$ may not be utilized by P.

The point set S, the set $\mathcal{D}$ of k (pairwise non-opposite) directions, the HIT H, together with mapping M and assignment A, form an instance of a restricted problem that asks for a minimum $\mathcal{D}$-monotone spanning tree having H as its HIT and respecting M and A. Let MMST($S, \mathcal{D}, H, M, A$) denote this problem instance. Note that such a monotone spanning tree may not exist. If it exists, it turns out that it is unique (see Lemma 11). The algorithm for solving instances

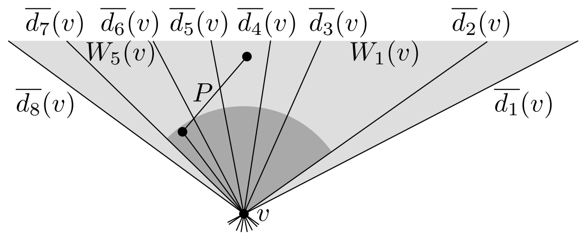


Fig. 7: A leaf path P that is assigned seven wedges but utilizes only five of them (shaded darkgray): It is not monotone with respect to $\{d_3, d_4, d_5, d_6\}$.

of type $\text{MMST}(S, \mathcal{D}, H, M, A)$ is repeatedly used by the algorithm that proves Theorem 2.

Lemma 11 ($\star$). *Let S be a set of n points, let $\mathcal{D}$ be a set of k (pairwise non-opposite) directions, let H be a HIT, let M be a mapping of the internal vertices of H to points of S, and let A be an assignment of $\mathcal{W}_\mathcal{D}$ to the leaves of H so that each leaf receives a distinct set of consecutive wedges. Then, $\text{MMST}(S, \mathcal{D}, H, M, A)$ can be solved in $O(n \log n + nk + k)$ time. Moreover, if a solution to the $\text{MMST}(S, \mathcal{D}, H, M, A)$ exists, then it is unique.*

Proof (sketch). Based on the characterization in Theorem 1, the algorithm checks whether point set S admits a $\mathcal{D}$-monotone spanning tree whose associated HIT is H, respecting mapping M and assignment A. Condition (b) of Theorem 1 is satisfied by definition since A is a valid assignment. For condition (c), we first compute the set $\mathcal{R}$ that consists of all path regions and branch regions and for every branch $B_{u,v}$, we compute $\mathcal{W}_{u \setminus v}$ and $\mathcal{W}_{v \setminus u}$. These computations take $O(k)$ time since HIT H has size $O(k)$. Then, the algorithm verifies, for every edge (u, v) of H, whether regions $\mathcal{W}_{u \setminus v}(u)$ and $R_{u,v}$ are disjoint, in $O(k)$ time. For condition (a), we compute, for every remaining point p in S, the region of $\mathcal{R}$ that contains p, in $O(nk)$ time. We then check, for every region in $\mathcal{R}$, whether there exists a path that (i) is monotone with respect to the two directions that are orthogonal to its boundaries and (ii) spans all points in the region. This can be done in $O(n \log n)$ time by sorting the points according to both directions. If the spanning tree exists, its uniqueness follows from the fact that each region in $\mathcal{R}$ contains a unique $\mathcal{D}$-monotone path. □

Theorem 2 ($\star$). *Let S be a set of n points, and let $\mathcal{D}$ be a set of k (pairwise non-opposite) distinct directions. There exists a function $f \colon \mathbb{N} \to \mathbb{N}$ such that, if S is in $\mathcal{D}$-general position, then we can compute a minimum $\mathcal{D}$-monotone spanning tree of S in $O(f(k) \cdot n^{2k-1} \log n)$ time. In other words, the problem $\text{MMST}(S, \mathcal{D})$ is in XP when parameterized by k.*

Proof (sketch). The given set $\mathcal{D}$ of k directions yields a set of $2k$ wedges. Hence, a $\mathcal{D}$-monotone spanning tree has at most $2k$ leaves and at most $2k-2$ branching vertices. We enumerate the at most $7^{2k} \cdot (2k)!$ HITs according to Lemma 10.

Let H be the current HIT, let ℓ be the number of leaves, and let $b \leq \ell - 2$ be the number of branching vertices of H. We go through each of the $O(n^b) = O(n^{2k-2})$ subsets of cardinality $b \leq 2k-2$ of S. Let M be the mapping of the branching vertices of H to points in S. Let A be the assignment of a set of consecutive wedges in $\mathcal{W}_\mathcal{D}$ to the leaves of H. There are at most $2k \cdot \binom{2k-1}{\ell-1} \leq 2k \cdot 2^{2k}$ many such assignments since we have $2k$ choices for mapping the first leaf to some wedge, and then we select $\ell - 1$ out of the $2k - 1$ remaining wedges that we attribute to a different leaf than the preceding wedge (in circular order). For each of the $n^{2k-2} \cdot f_0(k)$, (with $f_0(k) = 7^{2k} \cdot (2k)! \cdot 2k \cdot 4^k \in 2^{O(k \log k)}$) choices of a HIT H, mapping M and assignment A, we run the algorithm presented in the proof of Lemma 11 for the MMST$(S, \mathcal{D}, H, M, A)$, which terminates in $O(n \log n + nk + k)$ time. Finally, we return the shortest tree that we have found (if any). The total runtime is $O(f(k) \cdot n^{2k-1} \log n)$, where $f(k) = f_0(k) \cdot k \in 2^{O(k \log k)}$. We argue the correctness of the algorithm in Appendix B. ◻

Speed-Up for $|\mathcal{D}| = 2$: For $|\mathcal{D}| = 2$, the algorithm from Section 5 computes a $\mathcal{D}$-monotone spanning tree of a set of n points in the plane in $O(n^3 \log n)$ time. In Appendix B.1, we show how to speed this up to $O(n^2)$ time.

Solving MMST(S, k): When the set of directions $\mathcal{D}$ is not prescribed and we are asked to search over all possible sets of k directions, a minimum k-directional monotone spanning tree of a point set S can be identified in $O(n^2 \log n)$ and in $O(n^6)$ time for $k = 1$ and $k = 2$, respectively. For $k \geq 3$, we describe an XP algorithm that runs in $2^{O(k \log k)} \cdot n^{2k(2k-1)} \log n$ time w.r.t. k; see Appendix C.

6 Maximum Degree of the Minimum k-Directional MST

Since the (Euclidean) MST has maximum degree at most six [25], it is natural to ask whether this upper bound carries over to minimum k-directional monotone spanning trees. We prove that this is not the case by presenting a set $\mathcal{D}$ of k specific directions and a set S_k of $2k + 1$ points such that the unique monotone k-directional spanning tree of S_k has degree $2k$.

Let k be an even positive integer, and let $\mathcal{D} = \{d_1, d_2, \dots, d_k\}$ be the set of k distinct (pairwise non-opposite) directions (in ccw. order) such that d_1 is defined by the vector $(1, 0)$ and, for $1 \leq i < k$, $\angle d_i d_{i+1} = \frac{\pi}{k}$. Since k is even, it holds that $\mathcal{W}_\mathcal{D} = \mathcal{W}_{\overline{\mathcal{D}}}$ where $\overline{\mathcal{D}} = \{\overline{d_1}, \overline{d_2}, \dots, \overline{d_k}\}$. For simplicity, we consider W_0 to be the wedge defined by d_1 and d_2. We define $S_k = \{o\} \cup \{v_0, v_1, \dots, v_{2k-1}\}$ to be the set of $2k + 1$ points, where o is the origin and, for $i \in [k-1]$, v_i is placed on the unit circle in the (ccw.) second angle-trisection of wedge W_i of $\mathcal{W}_\mathcal{D}$; see Fig. 8a. By construction $S_k \setminus \{o\}$ is the vertex set of a regular $2k$-gon centered at o and the star with edges $ov_0, \dots, ov_{2k-1}$ is a valid monotone spanning tree for S_k of length $2k$. Thus, any solution of MMST$(S_k, \mathcal{D})$ has length at most $2k$.

Let T be a tree that spans S_k. We call *polygon vertices* the vertices of T distinct from o. We refer to edges of T connecting adjacent polygon vertices as *external*, to edges incident to o as *rays* and to all other edges as *chords*. To show

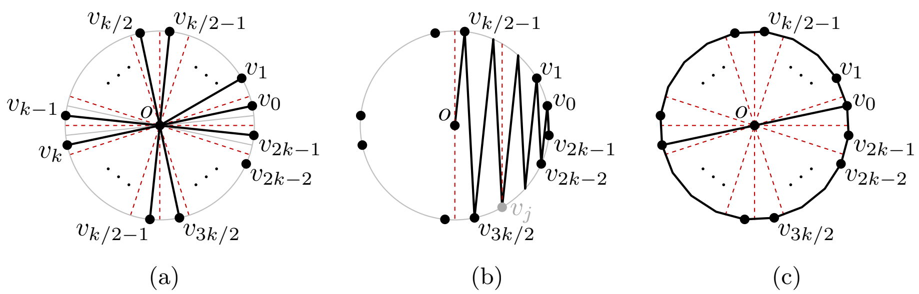


Fig. 8: (a) The point set S_k is defined based on the set $\mathcal{W}_{\mathcal{D}}$ of wedges (red dashed). (b) The path setting exploited in the proof of Theorem 3. (c) A monotone spanning graph of the point set in Fig. 8a whose length is much smaller than the $2k$-star in (a).

that the unique solution to the instance $\mathrm{MMST}(S_k, \mathcal{D})$ is the $2k$-star centered at o, we first establish that polygon vertices have degree at most 2.

Lemma 12 ($\star$). *Let T be a solution to the* $\mathrm{MMST}(S_k, \mathcal{D})$ *problem, and let x be a polygon vertex. Then,* $\deg_T(x) \leq 2$.

Theorem 3 ($\star$). *The only solution to the* $\mathrm{MMST}(S_k, \mathcal{D})$ *problem is the star $T^\star$ with center o and* $\deg_{T^\star}(o) = 2k \in \Omega(|S_k|)$.

Proof (sketch). Let tree T be a solution to the instance $\mathrm{MMST}(S_k, \mathcal{D})$ and assume that T is not the $2k$-star with o at its center. It is easy to show that a leaf of T cannot be the endpoint of a chord. By using this property together with the fact that all polygon vertices of T have degree at most 2 (Lemma 12), we can show that T must contain the path $P = \langle v_{2k-1}, v_0, v_{2k-2}, \ldots, v_{2k-1-i}, v_i, \ldots, v_{\frac{k}{2}-1}, o\rangle$, (Fig. 8b). Consider the tree T' formed by replacing the edges of P by rays from o to the path vertices. Clearly, T' is also monotone. To show that T is not optimal, it suffices to show that the length $\|P\|$ of P is greater than the total length of the rays that replaced the edges of P in T' or, equivalently, that $\|P\| > k$. Indeed, using geometry, we show that $\|P\| = 1 + \sum_{i=1}^{k-1} 2\sin\left(\frac{\pi}{2k} i\right) = \cot\left(\frac{\pi}{4k}\right) > k$. □

7 Open Problems

We have presented an XP algorithm for solving $\mathrm{MMST}(S, k)$. It is natural to ask whether this problem is NP-hard if k is part of the input (rather than a fixed constant).

Another research direction is to study, for a given point set S and a set $\mathcal{D}$ of directions, the problem of computing a minimum $\mathcal{D}$-monotone spanning *graph* for S. Note that such a graph can have much smaller total length than a solution to $\mathrm{MMST}(S, \mathcal{D})$. Indeed, Theorem 3 shows that there is a point set S_k (Fig. 8a) and a set $\mathcal{D}$ of k directions such that the only solution to $\mathrm{MMST}(S_k, \mathcal{D})$ is the $2k$-star, which has a total length of $2k$. A monotone spanning *graph* of S_k (see Fig. 8c) has a total length of at most $2(\pi + 1)$.

References

1. Soroush Alamdari, Timothy M. Chan, Elyot Grant, Anna Lubiw, and Vinayak Pathak. Self-approaching graphs. In Walter Didimo and Maurizio Patrignani, editors, *Proc. 20th Int. Symp. Graph Drawing (GD'12)*, volume 7704 of *LNCS*, pages 260–271. Springer, 2012. doi:10.1007/978-3-642-36763-2_23.
2. Patrizio Angelini. Monotone drawings of graphs with few directions. *Inform. Process. Lett.*, 120:16–22, 2017. doi:10.1016/j.ipl.2016.12.004.
3. Patrizio Angelini, Michael A. Bekos, Walter Didimo, Luca Grilli, Philipp Kindermann, Tamara Mchedlidze, Roman Prutkin, Antonios Symvonis, and Alessandra Tappini. Greedy rectilinear drawings. *Theor. Comput. Sci.*, 795:375–397, 2019. doi:10.1016/J.TCS.2019.07.019.
4. Patrizio Angelini, Enrico Colasante, Giuseppe Di Battista, Fabrizio Frati, and Maurizio Patrignani. Monotone drawings of graphs. *J. Graph Algorithms Appl.*, 16(1):5–35, 2012. doi:10.7155/jgaa.00249.
5. Patrizio Angelini, Walter Didimo, Stephen Kobourov, Tamara Mchedlidze, Vincenzo Roselli, Antonios Symvonis, and Stephen Wismath. Monotone drawings of graphs with fixed embedding. *Algorithmica*, 71:233–257, 2015. doi:10.1007/s00453-013-9790-3.
6. Patrizio Angelini, Fabrizio Frati, and Luca Grilli. An algorithm to construct greedy drawings of triangulations. *J. Graph Algorithms Appl.*, 14(1):19–51, 2010. doi:10.7155/jgaa.00197.
7. Esther M. Arkin, Robert Connelly, and Joseph S. B. Mitchell. On monotone paths among obstacles with applications to planning assemblies. In *Proc. 5th Ann. ACM Symp. Comput. Geom. (SoCG)*, pages 334–343, 1989. doi:10.1145/73833.73870.
8. Yeganeh Bahoo, Stephane Durocher, Sahar Mehrpour, and Debajyoti Mondal. Exploring increasing-chord paths and trees. In Joachim Gudmundsson and Michiel Smid, editors, *Proc. 29th Canadian Conf. Comput. Geometry (CCCG)*, pages 19–24, 2017.
9. Davood Bakhshesh and Mohammad Farshi. (Weakly) Self-approaching geometric graphs and spanners. *Comput. Geom.*, 78:20–36, 2019. doi:10.1016/j.comgeo.2018.10.002.
10. Davood Bakhshesh and Mohammad Farshi. Angle-monotonicity of Delaunay triangulation. *Comput. Geom.*, 94:101711, 2021. doi:10.1016/j.comgeo.2020.101711.
11. Davood Bakhshesh and Mohammad Farshi. On the plane angle-monotone graphs. *Comput. Geom.*, 100:101818, 2022. doi:10.1016/j.comgeo.2021.101818.
12. Nicolas Bonichon, Prosenjit Bose, Paz Carmi, Irina Kostitsyna, Anna Lubiw, and Sander Verdonschot. Gabriel triangulations and angle-monotone graphs: Local routing and recognition. In Yifan Hu and Martin Nöllenburg, editors, *Proc. Int. Symp Graph Drawing & Network Vis. (GD)*, volume 9801 of *LNCS*, pages 519–531. Springer, 2016. doi:10.1007/978-3-319-50106-2_40.
13. Bernard Chazelle and David P. Dobkin. Intersection of convex objects in two and three dimensions. *J. ACM*, 34(1):1–27, 1987. doi:10.1145/7531.24036.
14. Francis Chin and David Houck. Algorithms for updating minimal spanning trees. *J. Comput. Syst. Sci.*, 16(3):333–344, 1978. doi:10.1016/0022-0000(78)90022-3.
15. Hooman Reisi Dehkordi, Fabrizio Frati, and Joachim Gudmundsson. Increasing-chord graphs on point sets. In Christian Duncan and Antonios Symvonis, editors, *Proc. Int. Symp. Graph Drawing (GD)*, volume 8871 of *LNCS*, pages 464–475. Springer, 2014. doi:10.1007/978-3-662-45803-7_39.

16. Raghavan Dhandapani. Greedy drawings of triangulations. *Discrete. Comput. Geom.*, 43:375–392, 2010. doi:10.1007/s00454-009-9235-6.
17. Walter Didimo. Upward graph drawing. In Ming-Yang Kao, editor, *Encyclopedia of Algorithms*, pages 2308–2312. Springer, 2016. doi:10.1007/978-1-4939-2864-4_653.
18. David Eppstein. Finding the k smallest spanning trees. *BIT*, 32:237–248, 1992. doi:10.1007/BF01994879.
19. David Eppstein. Spanning trees and spanners. In J.-R. Sack and J. Urrutia, editors, *Handbook of Computational Geometry*, pages 425–461. North-Holland, Amsterdam, 2000. doi:10.1016/B978-044482537-7/50010-3.
20. Stefan Felsner, Alexander Igamberdiev, Philipp Kindermann, Boris Klemz, Tamara Mchedlidze, and Manfred Scheucher. Strongly monotone drawings of planar graphs. In Sándor Fekete and Anna Lubiw, editors, *Proc. 32nd Int. Symp. Comput. Geom. (SoCG)*, volume 51 of *LIPIcs*, pages 37:1–37:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2016. doi:10.4230/LIPIcs.SoCG.2016.37.
21. Andrea Francke and Michael Hoffmann. The Euclidean degree-4 minimum spanning tree problem is NP-hard. In *Proc. 25th Ann. ACM Symp. Comput. Geom. (SoCG)*, pages 179–188, 2009. doi:10.1145/1542362.1542399.
22. Greg N. Frederickson. Ambivalent data structures for dynamic 2-edge-connectivity and k smallest spanning trees. *SIAM J. Comput.*, 26(2):484–538, 1997. doi:10.1137/S0097539792226825.
23. Harold N. Gabow. Two algorithms for generating weighted spanning trees in order. *SIAM J. Comput.*, 6:139–150, 1977. doi:10.1137/0206011.
24. Ashim Garg and Roberto Tamassia. Upward planarity testing. *Order*, 12(2):109–133, 1995. doi:10.1007/BF01108622.
25. George Georgakopoulos and Christos H. Papadimitriou. The 1-Steiner tree problem. *J. Algorithms*, 8(1):122–130, 1987. doi:10.1016/0196-6774(87)90032-0.
26. Jacob E. Goodman and Richard Pollack. On the combinatorial classification of nondegenerate configurations in the plane. *J. Combin. Theory Ser. A*, 29(2):220–235, 1980. doi:10.1016/0097-3165(80)90011-4.
27. Ronald L. Graham and Pavol Hell. On the history of the minimum spanning tree problem. *Ann. Hist. Comput.*, 7(1):43–57, 1985. doi:10.1109/MAHC.1985.10011.
28. Frank Harary and Geert Prins. The number of homeomorphically irreducible trees, and other species. *Acta Math.*, 101(1–2):141–162, 1959. doi:10.1007/BF02559543.
29. Frank Harary, Robert W. Robinson, and Allen J. Schwenk. Twenty-step algorithm for determining the asymptotic number of trees of various species. *J. Austral. Math. Soc.*, 20(4):483–503, 1975. doi:10.1017/S1446788700016190.
30. Dayu He and Xin He. Optimal monotone drawings of trees. *SIAM J. Discrete Math.*, 31(3):1867–1877, 2017. doi:10.1137/16M1080045.
31. Xin He and Dayu He. Monotone drawings of 3-connected plane graphs. In Nikhil Bansal and Irene Finocchi, editors, *Proc. Europ. Symp. Algorithms (ESA)*, volume 9294 of *LNCS*, pages 729–741. Springer, 2015. doi:10.1007/978-3-662-48350-3_61.
32. Md. Iqbal Hossain and Md. Saidur Rahman. Good spanning trees in graph drawing. *Theoret. Comput. Sci.*, 607:149–165, 2015. doi:10.1016/j.tcs.2015.09.004.
33. Philipp Kindermann, André Schulz, Joachim Spoerhase, and Alexander Wolff. On monotone drawings of trees. In Christian Duncan and Antonios Symvonis, editors, *Proc. Int. Symp. Graph Drawing (GD)*, volume 8871 of *LNCS*, pages 488–500. Springer, 2014. doi:10.1007/978-3-662-45803-7_41.
34. Michael P. Knapp. Sines and cosines of angles in arithmetic progression. *Mathematics Magazine*, 82(5):371–372, 2009. doi:10.4169/002557009X478436.

35. Hsiang-Tsung Kung, Fabrizio L. Luccio, and Franco P. Preparata. On finding the maxima of a set of vectors. *J. ACM*, 22(4):469–476, 1975. doi:10.1145/321906.321910.
36. Anna Lubiw and Debajyoti Mondal. Construction and local routing for angle-monotone graphs. *J. Graph Algorithms Appl.*, 23(2):345–369, 2019. doi:10.7155/jgaa.00494.
37. Konstantinos Mastakas. Uniform 2d-monotone minimum spanning graphs. In Stephane Durocher and Shahin Kamali, editors, *Proc 30th Canadian Conf. Comput. Geom. (CCCG)*, pages 318–325, 2018. URL: https://arxiv.org/abs/1806.08770.
38. Konstantinos Mastakas. Drawing a rooted tree as a rooted y-monotone minimum spanning tree. *Inform. Process. Lett.*, 166:106035, 2021. doi:10.1016/j.ipl.2020.106035.
39. Konstantinos Mastakas and Antonios Symvonis. On the construction of increasing-chord graphs on convex point sets. In *Proc. 6th Int. Conf. Inform. Intell. Syst. Appl. (IISA)*, pages 1–6, 2015. doi:10.1109/IISA.2015.7388028.
40. Konstantinos Mastakas and Antonios Symvonis. Rooted uniform monotone minimum spanning trees. In Dimitris Fotakis, Aris Pagourtzis, and Vangelis Th. Paschos, editors, *Proc. Int. Conf. Algorithms & Complexity (CIAC)*, volume 10236 of *LNCS*, pages 405–417. Springer, 2017. doi:10.1007/978-3-319-57586-5_34.
41. Dragoslav S. Mitrinović. *Analytic Inequalities*. Springer, 1970. doi:10.1007/978-3-642-99970-3.
42. Martin Nöllenburg, Roman Prutkin, and Ignaz Rutter. On self-approaching and increasing-chord drawings of 3-connected planar graphs. *J. Comput. Geom.*, 7(1):47–69, 2016. doi:10.20382/jocg.v7i1a3.
43. Anargyros Oikonomou and Antonios Symvonis. Simple compact monotone tree drawings. In Fabrizio Frati and Kwan-Liu Ma, editors, *Proc. 25th Int. Symp. Graph Drawing & Netw. Vis. (GD)*, volume 10692 of *LNCS*, pages 326–333. Springer, 2017. doi:10.1007/978-3-319-73915-1_26.
44. Anargyros Oikonomou and Antonios Symvonis. Monotone drawings of k-inner planar graphs. In Therese Biedl and Andreas Kerren, editors, *Proc. 26th Int. Symp. Graph Drawing & Netw. Vis. (GD)*, volume 11282 of *LNCS*, pages 347–353. Springer, 2018. doi:10.1007/978-3-030-04414-5_24.
45. Christos H. Papadimitriou and David Ratajczak. On a conjecture related to geometric routing. *Theor. Comput. Sci.*, 344(1):3–14, 2005. doi:10.1016/j.tcs.2005.06.022.
46. Christos H. Papadimitriou and Umesh V. Vazirani. On two geometric problems related to the travelling salesman problem. *J. Algorithms*, 5(2):231–246, 1984. doi:10.1016/0196-6774(84)90029-4.
47. Ananth Rao, Sylvia Ratnasamy, Christos Papadimitriou, Scott Shenker, and Ion Stoica. Geographic routing without location information. In *Proc. 9th Ann. ACM Conf. Mobile Comput. Network. (MobiCom)*, pages 96–108, 2003. doi:10.1145/938985.938996.
48. Dae Young Seo, Der-Tsai Lee, and Tien-Ching Lin. Geometric minimum diameter minimum cost spanning tree problem. In *Proc. 20th Int. Symp. Algorithms & Comput. (ISAAC)*, page 283–292. Springer, 2009. doi:10.1007/978-3-642-10631-6_30.
49. Michael Ian Shamos and Dan Hoey. Closest-point problems. In *Proc. 16th Ann. IEEE Symp. Foundat. Comput. Sci. (FOCS)*, pages 151–162, 1975. doi:10.1109/SFCS.1975.8.

50. P. M. Spira and A. Pan. On finding and updating spanning trees and shortest paths. *SIAM J. Comput.*, 4(3):375–380, 1975. doi:10.1137/0204032.

A Additional Material for Section 3

Lemma 1 (⋆). *Let S be a set of points, and let $P = \langle u, x, v\rangle$ be a geometric path on S. Let d be a direction such that S is in d-general position. If u and v lie in the same half-plane determined by $\overline{d}(x)$, then the path P is not d-monotone.*

Proof. It is immediate to see that in the linear ordering $\mathrm{ord}(S, d)$, the projection of v either precedes or follows both the projections of u and x. Hence, the path $\langle u, x, v\rangle$ is not d-monotone (see Fig. 9). □

Lemma 2 (⋆). *Let S be a set of points, let T be a spanning tree of S, and let $\mathcal{D}$ be a set of k directions. Let x, u, and v be points in S such that $x \in P_{u,v}$. If u and v lie in the same wedge in $\mathcal{W}_{\mathcal{D}}(x)$, then the path $P_{u,v}$ is not $\mathcal{D}$-monotone.*

Proof. Let $S' = V(P_{u,v})$. For the path $P_{u,v}$ to be monotone with respect to some direction $d \in \mathcal{D}$, it must hold that u and v are located in different half-planes of $\overline{d}(x)$ so that x appears between u and v in $\mathrm{ord}(S', d)$. This is not possible, however, since u and v lie in the same wedge in $\mathcal{W}_{\mathcal{D}}(x)$. □

Lemma 3 (⋆). *Let S be a set of points, let $\mathcal{D}$ be a set of k directions, and let T be a $\mathcal{D}$-monotone spanning tree of S. Let $\Delta(T)$ denote the maximum degree of tree T. Then, $\Delta(T) \le 2k$.*

Proof. Let x be an arbitrary vertex of T. The set of lines $\{\overline{d}(x) \colon d \in \mathcal{D}\}$ partitions the plane into $2k$ wedges with apex x. By Lemma 1, if two neighbors u and v of x lie in the same wedge, then they lie in the same halfplane with respect to every direction d in $\mathcal{D}$. Hence, the path $\langle u, x, v\rangle$ is not monotone with respect to any direction in $\mathcal{D}$. Since T is $\mathcal{D}$-monotone, it follows that no two neighbors of x lie in the same wedge with apex x, which implies that $\deg(x) \le 2k$. □

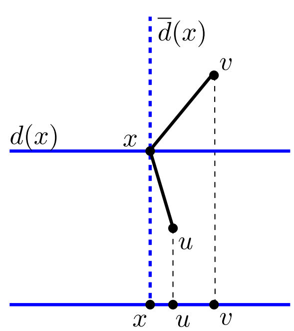


Fig. 9: The path $\langle u, x, v\rangle$ is not d-monotone.

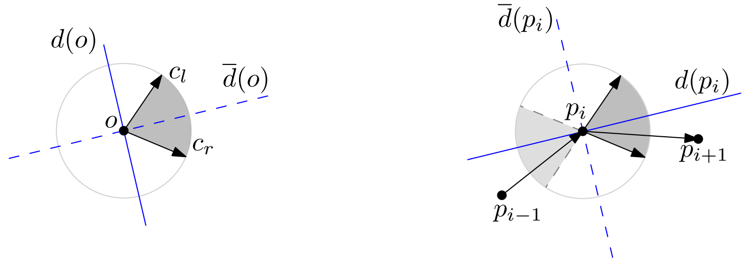


Fig. 10: (a) Line $\overline{d}(o)$ intersects $\sec(P)$. (b) Line $\overline{d}(p_i)$ does not intersect $\sec(P)$.

Lemma 5 (⋆). *Given a direction d, a monotone directed geometric path P is d-monotone if and only if $\overline{d}(o)$ does not intersect* $\sec(P)$, *where o is the origin.*

Proof. Assume first that $P = \langle p_1, \ldots, p_r \rangle$ is d-monotone. Suppose by contradiction that $\overline{d}(o)$ intersects $\sec(P)$; refer to Fig. 10a. Let c_f and c_l, $1 \leq f < r$ and $1 \leq l < r$, be the oriented segments of the unit circle that delimit the sector of directions $\sec(P)$. Then, the projections of the oriented segments $\overrightarrow{p_f p_{f+1}}$ and $\overrightarrow{p_l p_{l+1}}$ on line $d(o)$ point in opposite directions. This is a clear contradiction since all the projections of the oriented segments $\overrightarrow{p_i p_{i+1}}$, $1 \leq i < r$, of a monotone path point in the same direction. Also, note that in the boundary case where $\overline{d}(o)$ overlaps with c_f or c_l (or both), path P cannot be monotone since the projections of at least two of its points on $d(o)$ coincide; another contradiction.

Assume now that $\overline{d}(o)$ does not intersect $\sec(P)$. Consider three consecutive path points p_{i-1}, p_i, p_{i+1}, $1 < i < r$, and let the unit circle be centered at point p_i; refer to Fig. 10b. As points p_{i-1} and p_{i+1} are on opposite sides of line $\overline{d}(p_i)$, the projections of the oriented segments $\overrightarrow{p_{i-1} p_i}$ and $\overrightarrow{p_i p_{i+1}}$ on line $d(p_i)$ point in the same direction. Thus, path P is monotone. □

Lemma 6 (⋆). *Let S be a set of points, let $\mathcal{D}$ be a set of k directions, and let T be a $\mathcal{D}$-monotone spanning tree of S. Then, T has the following properties: (i) Let P be a directed path originating at vertex u of T. Then, P lies in $\mathcal{W}_P(u)$. (ii) Let P_1 and P_2 be two edge-disjoint directed paths originating at internal vertices u and v of T and terminating at leaves of T. Then, sets $\mathcal{W}_{P_1}$ and $\mathcal{W}_{P_2}$ are disjoint and regions $\mathcal{W}_{P_1}(u)$ and $\mathcal{W}_{P_2}(v)$ are disjoint.*

Proof. (i) Let d_1 and d_2 be the two directions in $\mathcal{D}$ orthogonal to the boundaries of $\mathcal{W}_P$. Then, due to Corollary 1, path P is both d_1- and d_2-monotone. Let u_1 be the vertex incident to u on P. Then, by definition of $\mathcal{W}_P$ we have that u_1 lies in $\mathcal{W}_P(u)$. Now let, for the sake of contradiction, x be a vertex of P that lies outside the region $\mathcal{W}_P(u)$. Observe that vertices x and u lie in the same halfplane with respect either to $\overline{d_1}(u_1)$ or $\overline{d_2}(u_1)$. Therefore, due to Lemma 1, path P is not monotone with respect to both d_1 and d_2. A contradiction.

(ii) Let $P_1 = P_{u,\lambda}$ and $P_2 = P_{v,\mu}$ where u and v are internal vertices of T and λ and μ are the corresponding leaves. Since P_1 and P_2 are edge-disjoint, path $P = P_{\lambda,\mu}$ from λ to μ is composed of $\overleftarrow{P_1}$, $P_{u,v}$ and P_2. Since T is $\mathcal{D}$-monotone, P must be $\mathcal{D}$-monotone with respect to at least one direction, say $d \in \mathcal{D}$. It follows that for any internal vertex w in the path the oriented subpaths $P_{\lambda,w}$ and $P_{w,\mu}$ lie in different halfplanes with respect to $\overline{d}(w)$.

(a) For the sake of contradiction assume that $\mathcal{W}_{P_1}$ and $\mathcal{W}_{P_2}$ overlap. Then, for $w = u$ we get that all vertices of P_1 must lie behind $\overline{d}(u)$. At the same time, for $w = v$ we get that all vertices of path P_2 must lie ahead of $\overline{d}(v)$. However, due to the fact that $\mathcal{W}_{P_1}$ and $\mathcal{W}_{P_2}$ overlap, no such direction d exists; a contradiction to the monotonicity of path P.

(b) As shown in (a), subpaths $\overleftarrow{P_1} = P_{\lambda,u}$ and $P_2 = P_{v,\mu}$ lie in different halfplanes with respect to $\overline{d}(u)$ (or $\overline{d}(v)$). Given that $\mathcal{W}_{P_1}$ and $\mathcal{W}_{P_2}$ are disjoint, we conclude that $\mathcal{W}_{P_1}(u) \cap \mathcal{W}_{P_2}(v) = \emptyset$.

□

Lemma 8 (⋆). *Let S be a set of points, let $\mathcal{D}$ be a set of k directions, let T be a $\mathcal{D}$-monotone spanning tree of S, and let u and v be two vertices of T. Then, it holds that: (i) Subtree $T_{u\setminus v}$ of T lies in $\mathcal{W}_{u\setminus v}(u)$. (ii) Sets $\mathcal{W}_{u\setminus v}$ and $\mathcal{W}_{v\setminus u}$ are disjoint, and regions $\mathcal{W}_{u\setminus v}(u)$ and $\mathcal{W}_{v\setminus u}(v)$ are disjoint.*

Proof. (i) Consider first a path P from vertex u oriented towards an arbitrary leaf λ of $T_{u\setminus v}$. By Lemma 6(i), we have that path P lies in region $\mathcal{W}_P(u)$. Since path P is composed of branches (zero or more) and a single leaf path in $T_{u\setminus v}$ it follows that $\mathcal{W}_P \subseteq \mathcal{W}_{u\setminus v}$ and, in turn, that path P lies in region $\mathcal{W}_{u\setminus v}(u)$. Since the union of all paths from u to the leaves of $T_{u\setminus v}$ covers all branches and leaf paths in $T_{u\setminus v}$ is follows that $T_{u\setminus v}$ lies in $\mathcal{W}_{u\setminus v}(u)$.

(ii) Observe that if one of the vertices u or v, say u, is a leaf, then $\mathcal{W}_{u\setminus v} = \emptyset$. Therefore, both statements of the lemma trivially hold. The same applies if u and v are distinct leaves. So, in the remainder of the proof we assume that u and v are internal tree vertices.

(a) For the sake of contradiction assume that $\mathcal{W}_{u\setminus v}$ and $\mathcal{W}_{v\setminus u}$ overlap. By Lemma 6(ii), we know that there do not exist directed paths P_1 and P_2 terminating at leaves of T that belong in $T_{u\setminus v}$ and $T_{v\setminus u}$, respectively, such that $\mathcal{W}_{P_1}$ overlaps with $\mathcal{W}_{P_2}$. So, without loss of generality, we assume that there exists a path $P = P_{w,\lambda}$ originating at an internal vertex w in $T_{v\setminus u}$ and terminating at leaf λ in $T_{v\setminus u}$ such that $\mathcal{W}_P \subset \mathcal{W}_{u\setminus v}$. Furthermore, let P_1 and P_2 be the paths terminating at leaves of $T_{u\setminus v}$ utilizing the leading and trailing wedges of $\mathcal{W}_{u\setminus v}$, respectively. Refer to Fig. 11. Since T is an embedded monotone tree, so is its subtree T' that consists of $T_{u\setminus v}$ and the path from u to λ (which passes from v and w) and uses the same embedding as T. Consider path P' from u to λ and let (u, u') be its first edge. Then, P' is a path terminating at leaf λ and its set of utilized wedges $\mathcal{W}_{P'}$ includes the wedge utilized by edge (u, u'), the wedges in $\mathcal{W}_P$ and the wedges in $\mathcal{W}_{P_{u',w}}$.

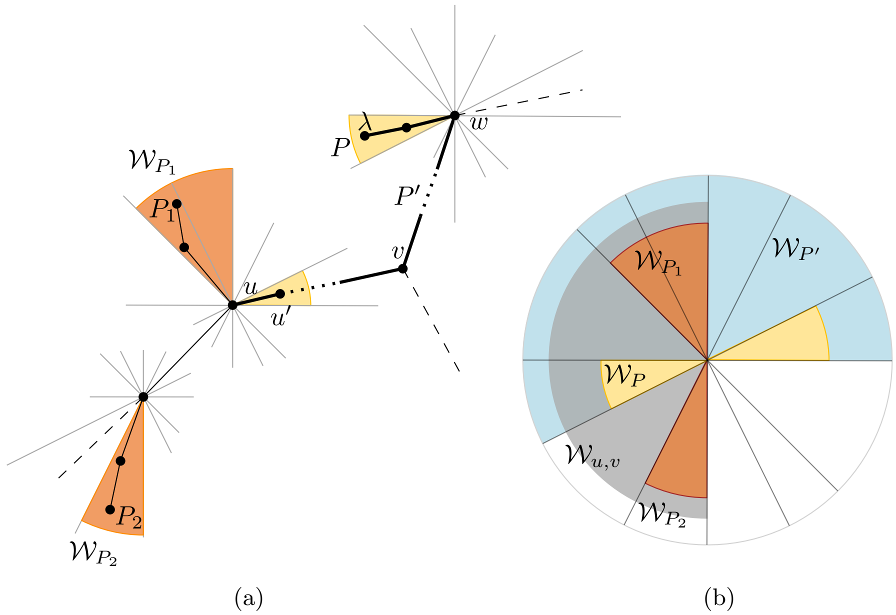


Fig. 11: Monotone embedded tree used in the proof of property Lemma 8(ii).

Thus, path P' is, at least, utilizing all wedges also utilized by either P_1 or P_2. Without loss of generality, assume that $\mathcal{W}_{P'}$ intersects with $\mathcal{W}_{P_1}$. However, given that both P' and P_1 are edge-disjoint paths terminating at leaves, by Lemma 6(ii), we have that $\mathcal{W}_{P'}$ and $\mathcal{W}_{P_1}$ are disjoint, a clear contradiction. We conclude that $\mathcal{W}_{u\setminus v}$ and $\mathcal{W}_{v\setminus u}$ are disjoint.

(b) Recall that due to (a) we have that $\mathcal{W}_{u\setminus v}$ and $\mathcal{W}_{v\setminus u}$ are disjoint. By the definition of $\mathcal{W}_{u\setminus v}$ and $\mathcal{W}_{v\setminus u}$ it follows that the leading and the trailing wedges of $\mathcal{W}_{u\setminus v}$ and $\mathcal{W}_{v\setminus u}$ are utilized. Let W_1 and W_1' be the leading and the trailing wedges of $\mathcal{W}_{u\setminus v}$ and let W_2 and W_2' be the leading and the trailing wedges of $\mathcal{W}_{v\setminus u}$ in ccw. order. Assume for the sake of contradiction that areas $\mathcal{W}_{u\setminus v}(u)$ and $\mathcal{W}_{v\setminus u}(v)$ intersect. Then, at least one of $W_1(u), W_1'(u)$ intersects with at least one of $W_2(v), W_2'(v)$. W.l.o.g., let $W_1(u)$ intersect with $W_2(v)$ and let $e_1 = (u_1, u_2)$ be the oriented edge away of u that utilizes wedge W_1 and $e_2 = (v_1, v_2)$ be the oriented edge away of v that utilizes wedge W_2. Let $P_1 = P_{u,u_2}$ be the path of T originating at u that utilizes wedge W_1 and let $P_2 = P_{v,v_2}$ be the path of T originating at v that utilizes wedge W_2. Since T is an embedded $\mathcal{D}$-monotone tree, so is its subtree $T' = P_{u_2,v_2}$ that uses the same embedding. But, in T', paths P_1 and P_2 are edge-disjoint and terminate at leaves of T'. Due to Lemma 6(ii), regions $\mathcal{W}_{P_1}(u)$ and $\mathcal{W}_{P_2}(v)$ are disjoint. A contradiction.

□

Lemma 9 (⋆). *Let S be a set of points, let $\mathcal{D}$ be a set of k (pairwise non-opposite) directions such that S is in $\mathcal{D}$-general position, and let T be a $\mathcal{D}$-monotone spanning tree of S. If $P_{u,v}$ is either a branch or a leaf path of T, then $\mathcal{W}_{P_{u,v}} \cap \mathcal{W}_{u\setminus v} = \emptyset$ and $R_{u,v} \cap \mathcal{W}_{u\setminus v}(u) = \emptyset$.*

Proof. Let T' be the subtree of T formed by $T_{u\setminus v}$ and $P_{u,v}$. Tree T' is $\mathcal{D}$-monotone since it is a subtree of T. Consider first the case where $P_{u,v}$ consists only of edge (u, v). Then, by definition, edge (u, v) utilizes a wedge which is not contained in $\mathcal{W}_{u\setminus v}$ and, thus, it immediately follows that $\mathcal{W}_{P_{u,v}} \cap \mathcal{W}_{u\setminus v} = \emptyset$. Consider now the case where path $P_{u,v}$ contains at least one intermediate vertex. Let (u, w) be the edge of $P_{u,v}$ incident to u. Edge (u, w) utilizes a wedge which is not contained in $\mathcal{W}_{u\setminus v}$. Consider now vertices u and w of T' and subtrees $T'_{u\setminus w}$ and $T'_{w\setminus u}$. By Lemma 8(ii) we have that $\mathcal{W}_{u\setminus w}$ and $\mathcal{W}_{w\setminus u}$ are disjoint with respect to T' and, therefore, they are also disjoint with respect to T. Since $T'_{u\setminus v} = T'_{u\setminus w}$ and $P_{u,v}$ is composed of edge (u, w) and $T'_{w\setminus u}$, we conclude that $\mathcal{W}_{P_{u,v}} \cap \mathcal{W}_{u\setminus v} = \emptyset$.

Now, observe that $R_{u,v} \subset \mathcal{W}_{P_{u,v}}(u)$. Since regions $R_{u,v}$ and $\mathcal{W}_{u\setminus v}(u)$ have the same apex and are contained in the disjoint sets of utilized wedges $\mathcal{W}_{P_{u,v}}$ and $\mathcal{W}_{u\setminus v}$, respectively, we also conclude that $R_{u,v} \cap \mathcal{W}_{u\setminus v}(u) = \emptyset$. □

B Additional Material for Section 5

Lemma 10 (⋆). *The number of different HITs with at most ℓ leaves is $O(7^\ell \cdot \ell!)$, and these HITs can be enumerated in $O(7^\ell \cdot \ell!)$ time.*

Proof. Denote by $\bar{n}_i$ the number of different HITs with *exactly* i leaves, for $i \geq 2$. We now prove, by induction on i, that $\bar{n}_i \leq 7^i \cdot i!$ for $i \geq 2$. For $i = 2$ there exists only one possible HIT, i.e., the tree consisting of a single edge. Suppose that $i > 2$. A HIT with i leaves can be obtained from a HIT with $i-1$ leaves by means of one of two operations: by attaching an edge (and a leaf) to an internal vertex (we call this Operation 1) or by subdividing an edge and attaching a new edge to the degree-two vertex created by the subdivision (we call this Operation 2). See Fig. 12 for an illustration.

Let T be an HIT with $i - 1$ leaves. Given T, let V be the set of vertices, let I be the set of internal vertices, let L be the set of leaves, and let m be the number of edges of T. If we perform Operation 1 on an internal vertex v of the tree T, we can obtain $\deg(v)$ different HITs, which have the same topology but different embedding depending on the position of the new edge in the circular order around v. Thus, the number of different HITs that can be generated starting from T by performing Operation 1 is $\sum_{v\in I} \deg(v)$. We have $\sum_{v\in V} \deg(v) = \sum_{v\in I} \deg(v) + \sum_{v\in L} \deg(v) = 2m$. The term $\sum_{v\in L} \deg(v)$ is equal to the number of leaves, that is $i-1$; moreover, since the number of leaves is $i - 1$, the number of vertices is at most $2i - 4$, and the number of edges m is at most $2i - 5$. Thus, we obtain $\sum_{v\in I} \deg(v) = 2(2i - 5) - i + 1 = 3i - 9$.

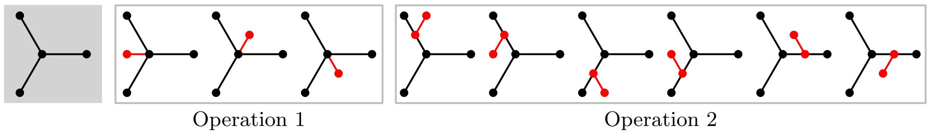


Fig. 12: A HIT with three leaves and all nine HITs that can be generated from it by means of Operations 1 and 2.

If we perform Operation 2 on an edge e of T, we can obtain 2 different HITs depending on the side of e where the new edge is added. Thus, from the tree T we can obtain at most $\sum_{v\in I}\deg(v)+2m = 3i-9+4i-10 = 7i-19$ different HITs, which implies $\bar{n}_i \leq 7i\bar{n}_{i-1}$. By induction, $\bar{n}_{i-1} \leq 7^{i-1}(i-1)!$ and therefore $\bar{n}_i \leq 7^i \cdot i!$.

The number of different HITs with *at most* ℓ leaves can now be computed as $\sum_{i=2}^{\ell} \bar{n}_i \leq \sum_{i=2}^{\ell} 7^i \cdot i! \leq \ell! \sum_{i=2}^{\ell} 7^i \leq \frac{7^\ell \cdot \ell!}{14}$. Clearly, all these HITs can be generated starting from the single tree with two leaves as described above by performing Operations 1 and 2. Since each operation can be executed in $O(1)$ time, the whole set can be generated in $O(7^\ell \cdot \ell!)$ time. □

Lemma 11 ($\star$). *Let S be a set of n points, let $\mathcal{D}$ be a set of k (pairwise non-opposite) directions, let H be a HIT, let M be a mapping of the internal vertices of H to points of S, and let A be an assignment of $\mathcal{W}_\mathcal{D}$ to the leaves of H so that each leaf receives a distinct set of consecutive wedges. Then,* MMST$(S,\mathcal{D},H,M,A)$ *can be solved in $O(n\log n + nk + k)$ time. Moreover, if a solution to the* MMST$(S,\mathcal{D},H,M,A)$ *exists, then it is unique.*

Proof. Let $\mathcal{B}^H$ be the internal vertices of H. As discussed, every internal vertex b_i^H of H corresponds to a branching vertex b_i^S in the solution of the problem MMST$(S,\mathcal{D},H,M,A)$. For each branch $B_{u,v}$ of H we compute $\mathcal{W}_{u\setminus v}$ and $\mathcal{W}_{v\setminus u}$ based on assignment A. This computation can be easily completed in total $O(k)$ time. Since $\mathcal{W}_{u\setminus v}$ and $\mathcal{W}_{v\setminus u}$ are complementary, the candidate region $R_{u,v}$ is uniquely defined. The same holds for $\mathcal{W}_{B_{u,v}}$ and $\mathcal{W}_{B_{v,u}}$. Let $\mathcal{B}^S$ be the set of points in S that correspond to internal vertices of $\mathcal{B}^H$ through mapping M. For each branch $B_{u,v}$ with $u,v \in \mathcal{B}^S$, our algorithm checks whether region $R_{u,v}$ is a valid area of the plane by verifying that v lies in $\mathcal{W}_{B_{u,v}}(u)$. If $R_{u,v}$ is valid, then $B_{u,v}$ must be contained in it; otherwise, the algorithm rejects the tuple $(S,\mathcal{D},H,M,A)$.

The assignment A of the wedges of $\mathcal{W}_\mathcal{D}$ to the leaves of H defines, for each leaf path, a reagion that must contain it. Let $\mathcal{R}$ denote the set of all leaf path regions and branch regions (that have already been computed). Observe that $|\mathcal{R}| = \ell + b - 1 \in O(k)$.

Due to Theorem 1, if $P_{u,v}$ is either a branch or a leaf path of a $\mathcal{D}$-monotone spanning tree, then areas $\mathcal{W}_{u\setminus v}(u)$ and $R_{u,v}$ must be disjoint. Since $\mathcal{W}_{u\setminus v}(u)$ is bounded by two semi-lines originating at u and $R_{u,v}$ is either a parallelogram or a strip between two parallel lines the test for their intersection can be completed in constant time [13]. In total, we can check in $O(k)$ time all intersections suggested by Theorem 1.

Now we compute, in total $O(nk)$ time, for each point p in $S \setminus \mathcal{B}^S$, the region in $\mathcal{R}$ that contains p. Due to Lemma 6(i) every leaf path P incident to a branching vertex v in the solution of the MMST$(S, \mathcal{D}, H, M, A)$ should be contained in $\mathcal{W}_P(v)$ and every branch $B_{u,v}$ between two branching vertices u and v should be contained in $R_{u,v}$. As a result, if there is a point p that does not lie in any region in $\mathcal{R}$ the algorithm rejects tuple $(S, \mathcal{D}, H, M, A)$. Additionally, if for a leaf λ_j in H incident to vertex b_i^H the corresponding region does not contain any points, then we also reject tuple $(S, \mathcal{D}, H, M, A)$ (because a missing leaf path induces a different HIT).

The last step of the algorithm is to go through every region $R \in \mathcal{R}$ and check whether there exists a spanning path of the points in R that is monotone with respect to the two directions d_1 and d_2 that are orthogonal to the boundaries of R. This can be achieved in $O(n \log n)$ time, by sorting the points according to d_1 and d_2 and compare whether both orderings coincide. If each region in $\mathcal{R}$ contains a $\mathcal{D}$-monotone path, then connecting all these paths yields a $\mathcal{D}$-monotone spanning tree T for S. Observe that T is unique, since in each region we have a unique $\mathcal{D}$-monotone path.

The algorithm for solving the MMST$(S, \mathcal{D}, H, M, A)$ terminates in $O(n \log n + nk + k)$ time. Its correctness is immediate from Theorem 1. □

Theorem 2 ($\star$). *Let S be a set of n points, and let $\mathcal{D}$ be a set of k (pairwise non-opposite) distinct directions. There exists a function $f \colon \mathbb{N} \to \mathbb{N}$ such that, if S is in $\mathcal{D}$-general position, then we can compute a minimum $\mathcal{D}$-monotone spanning tree of S in $O(f(k) \cdot n^{2k-1} \log n)$ time. In other words, the problem* MMST$(S, \mathcal{D})$ *is in XP when parameterized by k.*

Proof. The given set $\mathcal{D}$ of k directions yields a set of $2k$ wedges. Hence, a $\mathcal{D}$-monotone spanning tree has at most $2k$ leaves and at most $2k-2$ branching vertices. We enumerate the at most $7^{2k} \cdot (2k)!$ HITs with at most $2k$ leaves according to Lemma 10. Let H be the current HIT, and let $\ell \leq 2k$ be the number of leaves of T. Then T has at most $b = \ell - 2$ branching vertices. We go through each of the $O(n^b) = O(n^{2k-2})$ subsets of cardinality $b \leq 2k-2$ of S. Let M be the mapping of the internal vertices of H to points in S. Let A be the assignment of a set of consecutive wedges in $\mathcal{W}_\mathcal{D}$ to the leaves of H. There are at most $2k \cdot \binom{2k-1}{\ell-1} \leq 2k \cdot 2^{2k}$ many such assignments since we have $2k$ choices for mapping the first leaf to some wedge, and then we select $\ell - 1$ out of the $2k-1$ remaining wedges that we attribute to a different leaf than the preceding wedge (in circular order). For each of the $n^{2k-2} \cdot f_0(k)$, (with $f_0(k) = 7^{2k} \cdot (2k)! \cdot 2k \cdot 4^k \in 2^{O(k \log k)}$) choices of a HIT H, mapping M and assignment A, we run the algorithm presented in the proof of Lemma 11 for the MMST$(S, \mathcal{D}, H, M, A)$, which terminates in $O(n \log n + nk + k)$ time. Finally, we return the shortest tree that we found (if any). The total runtime is $O(f(k) \cdot n^{2k-1} \log n)$, where $f(k) = f_0(k) \cdot k \in 2^{O(k \log k)}$.

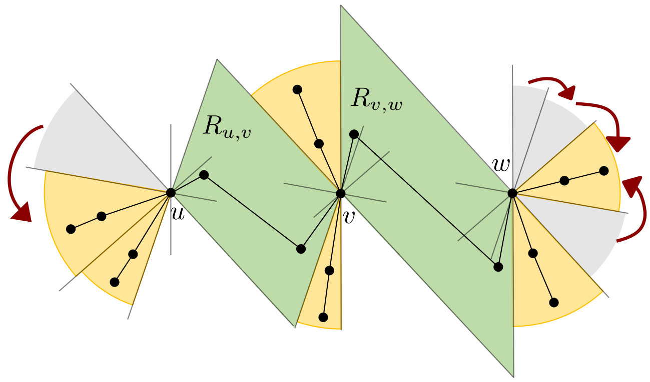


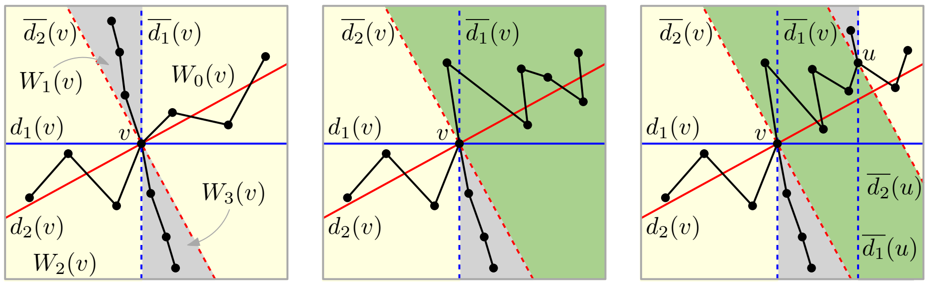
Fig. 13: Gray regions are the wedges not utilized by any leaf path. The red arrows indicate the leaf the wedge is assigned to as described in the proof of Theorem 2.

It remains to show the correctness of our approach. Towards that goal it is sufficient to show that for any $\mathcal{D}$-monotone spanning tree T there exists a HIT H, a mapping M and an assignment A, such that T is the solution to the problem $\mathrm{MMST}(S, \mathcal{D}, H, M, A)$. We fix H to be the unique HIT of tree T and M to be the corresponding mapping of the internal vertices of H to the branching vertices of T. We proceed to show how to specify an appropriate wedge assignment A.

We initialize our assignment A by adopting the actual wedge usage of the leaf paths of T. We describe now how to extend A by assigning the remaining wedges of $\mathcal{W}_\mathcal{D}$ to the existing leaf paths of T based on the branches of T. Since T is a $\mathcal{D}$-monotone spanning tree of S, for every directed branch $B_{v,u}$ of T we know $\mathcal{W}_{u \setminus v}$ and $\mathcal{W}_{B_{v,u}}$. If $\mathcal{W}_{B_{v,u}}$ contains wedges which are not included in $\mathcal{W}_{u \setminus v}$ then we assign these wedges to the leading and/or the trailing leaf path that utilizes wedges in $\mathcal{W}_{u \setminus v}$ (refer to Fig. 13). Note that these wedges are not utilized by any other leaf path. Also note that the same wedge W cannot receive contradicting assignment due to two different branches. If that was the case then these two branches would have to be oppositely facing in the path P that has them at its ends. Then, path P wouldn't be monotone since $\mathcal{W}_P$ would contain both wedge W and its opposite wedge. Any remaining unassigned wedges after the processing of all branches of T is assigned arbitrarily to a leaf path that utilizes the ccw. neighboring wedges. The resulting assignment A assigns all $2k$ wedges of $\mathcal{W}_\mathcal{D}$ to leaf paths of H and is consistent with the $\mathcal{D}$-monotone tree T. □

B.1 Speed-Up for $|\mathcal{D}| = 2$

For $|\mathcal{D}| = 2$, the algorithm from Section 5 computes a $\mathcal{D}$-monotone spanning tree of a set of n points in the plane in $O(n^3 \log n)$ time. We now speed this up to $O(n^2)$ time. Recall that, for $\mathcal{D} = \{d_1, d_2\}$ and a point p, $\mathcal{W}_\mathcal{D}(p) = \{W_0(p), W_1(p), W_2(p), W_3(p)\}$ denotes the set of wedges formed at p by the lines orthogonal to the directions in $\mathcal{D}$. By Lemma 7, a 2-directional spanning tree has



(a) *single-degree-4 $\mathcal{D}$-tree* (b) *single-degree-3 $\mathcal{D}$-tree* (c) *double-degree-3 $\mathcal{D}$-tree*

Fig. 14: Different topologies of a spanning tree that is monotone w.r.t. $\mathcal{D} = \{d_1, d_2\}$.

at most four leaves. Hence, by Lemma 10, there are only $O(1)$ different HITs, namely the *$\mathcal{D}$-path* and the topologies depicted in Fig. 14.

Observation 1. Let $\mathcal{D}$ be a set of two non-opposite directions, and let T be a $\mathcal{D}$-directional spanning tree. Then, T is either a $\mathcal{D}$-path, a single-degree-4 $\mathcal{D}$-tree, a single-degree-3 $\mathcal{D}$-tree, or a double-degree-3 $\mathcal{D}$-tree (defined below).

1. A *$\mathcal{D}$-path* is simply a path; clearly it must be d_1-monotone or d_2-monotone.
2. A *single-degree-4 $\mathcal{D}$-tree* consists of a degree-4 vertex v and four leaf paths emanating from v. By Theorem 1(ii), each leaf path lies in a distinct wedge of $\mathcal{W}_\mathcal{D}(v)$. Since every wedge is bounded by both $\overline{d_1}$ and $\overline{d_2}$, Corollary 1 ensures that each leaf path is both d_1- and d_2-monotone.
3. A *single-degree-3 $\mathcal{D}$-tree* consists of a degree-3 vertex v and three paths emanating from v such that, for some $i \in \{0, 1, 2, 3\}$, one path lies in the wedge $W_i(v)$ and one in $W_{i+1}(v)$, these two paths are both d_1- and d_2-monotone, and the third path connects all points in $W_{i+2}(v) \cup W_{i+3}(v)$ and is d-monotone, where $d \in \mathcal{D}$ is the direction orthogonal to the line that separates $W_i(v) \cup W_{i+1}(v)$ from $W_{i+2}(v) \cup W_{i+3}(v)$.
4. A *double-degree-3 $\mathcal{D}$-tree* consists of two degree-3 vertices u and v and five paths such that, for some $i \in \{0, 1, 2, 3\}$, one path lies in $W_i(u)$, one in $W_{i+1}(u)$, one in $W_{i+2}(v)$, and one in $W_{i+3}(v)$; these four paths are both d_1- and d_2-monotone, and the fifth path connects all points in the infinite strip $\mathbb{R}^2 \setminus (W_i(u) \cup W_{i+1}(u) \cup W_{i+2}(v) \cup W_{i+3}(v))$ and is d-monotone, where $d \in \mathcal{D}$ is the direction orthogonal to the two lines delimiting the strip.

In the above characterization of a $\mathcal{D}$-monotone spanning tree for the case $|\mathcal{D}| = 2$, we heavily exploit Corollary 1, which ensures that a leaf path or branch P must be d-monotone for every direction d such that $\overline{d}$ bounds $\mathcal{W}_P$. (Above, we argued this explicitly only for the single degree-4 $\mathcal{D}$-tree.)

The following two lemmas lead to the main result of this section, Theorem 4. The algorithm behind Lemma 13 is reminiscent of the sweep-line algorithm for computing the maxima of a set of points [35]. It is easy to implement, but its analysis is somewhat intricate.

Lemma 13. *Given a set $\mathcal{D} = \{d_1, d_2\}$ of two (non-opposite) directions and a point set S in $\mathcal{D}$-general position, we can compute a table $\mathcal{Q}(S, \mathcal{D})$ such that: (i) $\mathcal{Q}(S, \mathcal{D})$ reports in $O(1)$ time, for a query point p in S and $i \in \{0, 1, 2, 3\}$, whether the points in $W_i(p) \cap S$ form a path that is both d_1- and d_2-monotone. If yes, the length of the path is also reported. (ii) $\mathcal{Q}(S, \mathcal{D})$ has size $O(n)$ and can be computed in $O(n \log n)$ time, where $n = |S|$.*

Proof. Let $\mathcal{D} = \{d_1, d_2\}$. The table $\mathcal{Q}(S, \mathcal{D})$ simply stores, for each pair (p, i), with $p \in S$ and $i \in \{0, 1, 2, 3\}$, the following data: (1) a Boolean flag that is true if and only if the points in $W_i(p) \cap S$ form a path that is both d_1- and d_2-monotone and, if the flag is true, (2) the length $\ell(p, i)$ of the corresponding path. Observe that $\mathcal{Q}(S, \mathcal{D})$ has size $O(n)$.

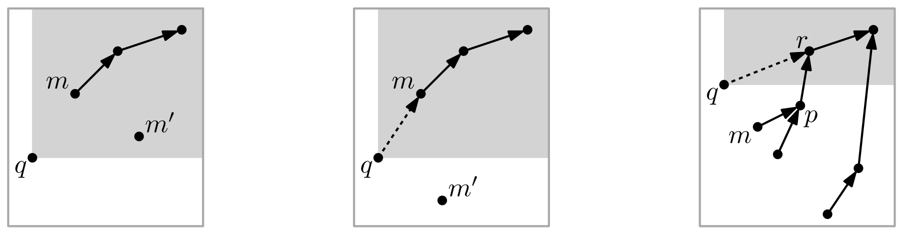
To construct $\mathcal{Q}(S, \mathcal{D})$ in $O(n \log n)$ time, we proceed as follows. For each wedge W_i in $\mathcal{W}_\mathcal{D}$ with $i \in \{0, 1, 2, 3\}$, we first transform the point set S by an affine transformation that maps the d_1-coordinates to x-coordinates and the d_2-coordinates to y-coordinates. Additionally, we make sure that the wedge W_i corresponds to the first quadrant. This can always be achieved by appropriately multiplying all coordinates of some type by $+1$ or by -1. Hence, after our transformation, for any point q in S, $W_i(q) = W_1(q)$ is the first quadrant with respect to q.

Sort the points by x-coordinate. This takes $O(n \log n)$ time. The rest of the algorithm is iterative; it takes only $O(n)$ time. For a point p in S, let p_y be its y-coordinate and let p_x be its x-coordinate. To simplify the description of the algorithm, we assume that no two points have the same x- or y-coordinate. We say that a point p in S *dominates* a point r if $p_x > r_x$ and $p_y > r_y$. We say that p *directly dominates* r if there is no point q in S such that p dominates q and q dominates r. In other words, given a point set S', there is an x- and y-monotone path through the points in S' if and only if no point in S' is directly dominated by two other points.

Scan the points in S in order of decreasing x-coordinates. For each point q, we do the simple test described below. If q passes the test, we set its flag to true, establish a pointer to the next point p on its x- and y-monotone path, and set $\ell(q, i) = \ell(p, i) + d(q, p)$, where $d(p, q)$ is the Euclidean distance of the (untransformed) points p and q. If a point in S has no edge directed into it, then we call it *minimal*. At any time, we maintain the minimal point m in S that currently has the largest y-coordinate. We also maintain the point m' that has the largest y-coordinate among the points in $S \cap W_4(m)$ (that is, among the points to the right and below m). Note that m' may or may not be minimal. In the first iteration, we set m to the rightmost point and set its flag to true. For simplicity, we initially set m' to a dummy point at $(\infty, -\infty)$.

For any further iteration, let q be the current point in S. There are three cases depending on the vertical position of q with respect to m and m'; see Fig. 15:

1. If $q_y < m'_y$, then q fails the test because the points m and m' both dominate it directly; see Fig. 15a. Set the flag of q to false.



(a) $q_y < m'_y\,(< m_y)$ (b) $m'_y < q_y < m_y$ (c) $(m'_y <)\, m_y < q_y$

Fig. 15: The three cases that occur in the iterative algorithm.

2. If $m'_y < q_y < m_y$, then set the flag of q to true, establish a pointer from q to m, and set $m = q$; see Fig. 15b.
3. If $m_y < q_y$, then we follow pointers from m to its successors as long as the current point is below q; see Fig. 15c. If the last such point p has a pointer to a point r in $W_1(q)$, establish a pointer from q to r. Independently of that, set the flag of q to true, set $m = q$, and set $m' = p$.

The algorithm maintains the following invariant throughout the algorithm: The point m is the starting point of a (possibly empty) x- and y-monotone path through all points in $W_1(m) \cap S$. Accordingly, the flag of m is always true.

Note that in case 1, neither m nor m' changes. In case 2, m' does not change, whereas m goes down (but stays above m'). Only in case 3 the point m' changes. In that case, m and m' go up (that is, their y-coordinates increase).

For the runtime analysis, note that every point has at most one pointer to any other point, and we traverse each pointer at most once. This is due to the fact that (a) the point m' never goes down (b) m is always above m', and (c) the pointers that we traverse in case 3 on the path from m to p (or r) originate in points that will be below m' after we update m' to p.

For the correctness, we consider the three possible types of wrong outcomes of the algorithm and show that each of them leads to a contradiction.

First assume that there is a point q in S such that the points in $W_1(q) \cap S$ form an x- and y-monotone path, but the algorithm set the flag of q to false. Suppose that q is the first (that is, rightmost) point where the algorithm makes this mistake. But then the flag of the successor q' of q on the path is true, and the points in $W_1(q) \cap S$ form an x- and y-monotone path starting in q'. When the algorithm reaches q, either q' is a minimal point, so $m = q'$ (case 2; note that m cannot be above q' because either q or q' would be directly dominated by two points), or m is below q (case 3). However, in both cases the algorithm would have added a pointer from q to q' and would have set the flag of q to true, contradicting our assumption.

Now assume that there is a point q in S that is directly dominated by two other points in S, but the algorithm set the flag of q to true. Suppose again that q is the first point where the algorithm makes this mistake, and let p and r with $p_y > r_y$ be the two points that directly dominate q. If p is minimal, then either $m = p$ and $m' = r$ and we are exactly in the situation of case 1, or m is above p and/or m' is above r, and we are still in case 1. So if p is minimal, the algorithm sets the flag of q to false, contradicting our assumption. If p is not minimal, then there is a point q' to the right of (and below) q that has a pointer to p. But q' would be directly dominated by p and r, contradicting our choice of q.

Finally, assume that there is a point q in S such that $W_1(q) \cap S$ contains a point q' directly dominated by two other points in S, but the algorithm set the flag of q to true. Let q' be the last such point in $W_1(q) \cap S$ treated by the algorithm. As we have argued above, the algorithm has correctly recognized q' (due to pints m and m' in $W_1(q')$). Until the algorithm treats q, the points m and m' may change, but since m' never goes down and m stays above m', both m and m' are contained in $W_1(q)$ (which contains $W_1(q')$). Hence, the algorithm would actually have set the flag of q to false when treating q, contradicting our assumption. □

Lemma 14. *Given a direction d and a point set S in d-general position, we can compute a table $\mathcal{Q}'(S, d)$ such that: (i) $\mathcal{Q}'(S, d)$ reports in $O(1)$ time, for a query pair of points $\{p, q\}$ in S, the length of the unique d-monotone path from p to q passing through all points in the infinite strip bounded by $\overline{d}(p)$ and $\overline{d}(q)$. (ii) $\mathcal{Q}'(S, d)$ has size $O(n)$ and can be computed in $O(n \log n)$ time, where $n = |S|$.*

Proof. Let $\operatorname{ord}(S, d) = p_1, p_2, \dots, p_n$. The table $\mathcal{Q}'(S, d)$ associates with each $i = 1, 2, \dots, n$ the length l_i of the path $p_1, p_2, \dots, p_i$. Given a pair of points $p = p_i$ and $q = p_j$, with $1 \leq i, j \leq n$, the length of the path from p to q passing through all points in the infinite strip bounded by $\overline{d}(p)$ and $\overline{d}(q)$ is $|l_j - l_i|$, which is computed in constant time using the values stored in the table at indices i and j. □

Theorem 4. *Let S be a set of n points, and let $\mathcal{D} = \{d_1, d_2\}$ be a set of two (non-opposite) distinct directions such that S is in $\mathcal{D}$-general position. There exists an $O(n^2)$-time algorithm that computes a minimum $\mathcal{D}$-monotone spanning tree of S.*

Proof. We give an algorithm that, for each of the four potential topologies listed at the beginning of this section, checks whether a spanning tree with that topology exists. If this is the case, the algorithm computes one of minimum length. Among the at most four resulting trees, the algorithm returns one of minimum length.

We first set up the data structure $\mathcal{Q}(S, \mathcal{D})$ mentioned in Lemma 13. Then, we set up the data structures $\mathcal{Q}'(S, d_1)$ and $\mathcal{Q}'(S, d_2)$ of Lemma 14. This preprocessing takes $O(n \log n)$ total time. $\mathcal{Q}'(S, d_1)$ and $\mathcal{Q}'(S, d_2)$ immediately give us the lengths of the unique d_1- and d_2-monotone spanning paths. The shorter of the two is a $\mathcal{D}$-path and is stored as a candidate for the minimum $\mathcal{D}$-monotone spanning tree of S.

Then we go through each point p in S and check whether p can be the unique degree-4 node of a single-degree-4 $\mathcal{D}$-tree. To this end, we query $\mathcal{Q}(S, \mathcal{D})$ with p

and with each of the four wedges in $\mathcal{W}_\mathcal{D}(p)$. If the data structure returns "yes" four times, that is, if the points in each of the four wedges form a d_1- and d_2-monotone path, we add up their lengths and compare their sum to the length of the shortest single-degree-4 $\mathcal{D}$-tree found so far (if any).

As for the previous case, for the single-degree-3 $\mathcal{D}$-tree we go through each point p in S and check whether p can be the unique degree-3 node. We query $\mathcal{Q}(S, \mathcal{D})$ with p and with each of the four wedges in $\mathcal{W}_\mathcal{D}(p)$. If the data structure returns "yes" for a pair of neighboring wedges $W_i(p)$ and $W_{i+1}(p)$, let l_1 and l_2 be the lengths of the paths in $W_i(p)$ and $W_{i+1}(p)$, respectively, that are both d_1- and d_2-monotone. Let $d \in \mathcal{D}$ be the direction orthogonal to the line separating $W_i(p)$ and $W_{i+1}(p)$. We query $\mathcal{Q}'(S, d)$ for the length l_3 of the d-monotone path that starts in p and goes through all points in $W_{i+2}(p) \cup W_{i+3}(p)$. Then we compare the sum $l_1 + l_2 + l_3$ to the length of the shortest single-degree-3 $\mathcal{D}$-tree found so far (if any).

Finally, we compute a minimum-length double-degree-3 $\mathcal{D}$-tree, if such a tree exists. We go through every pair $\{p, q\}$ of points in S and check whether S admits a double-degree-3 $\mathcal{D}$-tree whose only two degree-3 vertices are p and q. To this end, we query the table $\mathcal{Q}(S, \mathcal{D})$ with p and with each of the four wedges in $\mathcal{W}_\mathcal{D}(p)$. If the table has stored "true" for a pair of neighboring wedges $W_i(p)$ and $W_{i+1}(p)$, then we define l_1, l_2, and $d \in \mathcal{D}$ as in the case of the single-degree-3 $\mathcal{D}$-tree. Now we query $\mathcal{Q}(S, \mathcal{D})$ with q and with the two wedges $W_{i+2}(q)$ and $W_{i+3}(q)$ in $\mathcal{W}_\mathcal{D}(q)$. If the table has stored "true" for $W_{i+2}(q)$ and $W_{i+3}(q)$, then let l_3 and l_4 be the lengths of the corresponding paths in $W_{i+2}(q)$ and $W_{i+3}(q)$. We query $\mathcal{Q}'(S, d)$ for the length l_5 of the d-monotone path that starts in p, goes through all points in the strip delimited by $\overline{d}(p)$ and $\overline{d}(q)$, and ends in q. Then we compare the sum $l_1 + \cdots + l_5$ to the length of the shortest double-degree-3 $\mathcal{D}$-tree that we have found so far (if any).

Clearly, after the $O(n \log n)$-time preprocessing, the running time of the algorithm is dominated by the time needed to compute the shortest double-degree-3 $\mathcal{D}$-tree (if any). This computation requires to iterate over all pairs of points in S, but, using $\mathcal{Q}(S, \mathcal{D})$, $\mathcal{Q}'(S, d_1)$, and $\mathcal{Q}'(S, d_2)$, we have only constant work for each pair, and hence $O(n^2)$ time in total. □

C Algorithms for MMST(S, k)

This section considers the MMST(S, k) problem, where S is a set of points and k is a positive integer. Recall that, in this case, we want to find a minimum $\mathcal{D}$-monotone spanning tree over all possible sets $\mathcal{D}$ of k directions. We first address the cases $k = 1$ (Theorem 5) and $k = 2$ (Theorem 6), and then we give a general result for any positive integer k (Theorem 7).

Theorem 5. *Given a set S of n points, a solution to the* MMST($S, 1$) *problem can be computed in $O(n^2 \log n)$ time.*

Proof. Based on Lemma 3, any 1-directional monotone spanning tree of S is necessarily a path. For any given direction d such that S is in d-general position, consider $\mathrm{ord}(S,d)$. If we connect every two points of S whose projections are consecutive in $\mathrm{ord}(S,d)$, we uniquely define a d-monotone spanning path of S. Note that, for two distinct directions d and d', the d-monotone spanning path might coincide with the d'-monotone spanning path. We describe an $O(n^2 \log n)$-time algorithm that solves $\mathrm{MMST}(S,1)$; it considers all (and only) the distinct 1-directional monotone spanning paths of S and returns one of minimum length.

Assume, for now, that the point set S does not contain three or more collinear points and, moreover, no two pairs of points are lying on parallel lines. Later on, we will describe how to deal with an arbitrary point set. Let o be a point in the plane such that $o \notin S$, and define set $\mathcal{L}$ to consist of the $h = \binom{n}{2}$ lines $\overline{l_{u,v}}(o)$, $u, v \in S$ with $u \neq v$, passing through the origin o (see the dashed lines in Fig. 16). Then, these h lines partition the unit circle into $2h$ sectors. Start from an arbitrary sector and let d_1 be the direction that bisects it. Consider then the next sector in ccw. order and let d_2 be the direction that bisects it. By continuing in this manner, we can define a circular sequence $\sigma = \langle d_1, d_2, \ldots, d_h \rangle$ of h pairwise non-opposite directions (see the red direction in Fig. 16). This construction of the direction set σ was outlined by Goodman and Pollack [26]. They showed that, for every $i = 1, \ldots, h-1$, the linear orderings $\mathrm{ord}(S, d_i)$ and $\mathrm{ord}(S, d_{i+1})$ differ exactly for the positions of two consecutive points p and p', namely p immediately precedes p' in $\mathrm{ord}(S, d_i)$, while p immediately follows p' in $\mathrm{ord}(S, d_{i+1})$. By construction, for each direction $d \in \sigma$, S is in d-general position. Also, σ can be computed in $O(n^2 \log n)$ time by sorting the distinct slopes of the $\binom{n}{2}$ lines that are defined by point pairs in S.

Obviously, each direction d in σ defines a distinct monotone spanning path of S; to form the path, simply connect the points in order of appearance (of their projections) in $\mathrm{ord}(S,d)$. Moreover, for each monotone spanning path of S there exists at least one direction d of σ such that this path is d-monotone. To see that, first consider the set of consecutive directions such that a path P is d-monotone. We call $\overline{\mathrm{sec}}(P)$ the monotonicity interval of P. By Lemma 5, $\overline{\mathrm{sec}}(P)$ is an open sector of the unit circle. Now, simply observe that the boundaries of the monotonicity interval of path P, $\overline{\mathrm{sec}}(P)$, are lines in $\mathcal{L}$ and, thus, at least one direction in σ is contained in the monotonicity interval. For $i \in [h]$, let P_i be the monotone path defined by $\mathrm{ord}(S, d_i)$, and let $\|P_i\|$ be the length of P_i. For implementation purposes, we assume that projections of points of S in $\mathrm{ord}(S, d_i)$ (and, consequently, path P_i) are stored in order of appearance in a doubly linked list of *projection objects*. Moreover, in order to be able to locate the projection of each point in the sorted list, we maintain with each point of S a pointer to its projection object. Path P_1 and its length $\|P_1\|$ are easily computed in $O(n \log n)$ time by sorting the points in S with respect to their projection on d_1. Then, for each $i \in \{2, \ldots, h\}$, path P_i and its length $\|P_i\|$ are computed in $O(1)$ time from P_{i-1} and $\|P_{i-1}\|$, by just updating the segments incident to the two points p and p' of S that exchange their position passing from P_{i-1} to P_i. Note that it is not hard to identify p and p', as they are the points that define the line $\overline{l_{p,p'}}(o)$ that separates d_{i-1} and d_i in our construction; simply associate with each line

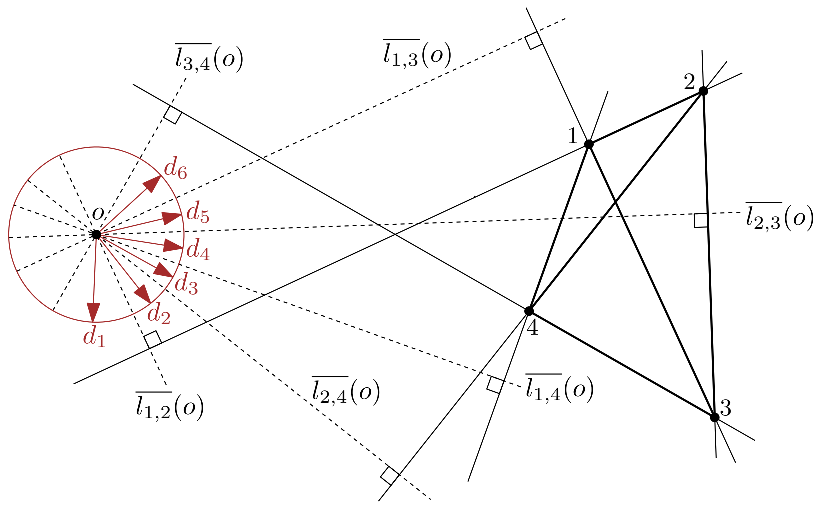


Fig. 16: Point set $S = \{1, 2, 3, 4\}$ and illustration of the computation of a set of distinct directions over which the minimum monotone spanning path of S is computed.

in our construction the two points that define it. Hence, the algorithm computes all paths defined by σ and the lengths of these paths in $O(n^2 \log n)$ time.

We now describe how to deal with the case where point set S contains pairs of points lying on parallel lines. Note that these pairs of points may be lying on the same line, resulting to having more that three collinear points. For an example point set, refer to Fig. 17.

We again compute set $\mathcal{L}$ consisting of the $h = \binom{n}{2}$ lines $\overline{l_{u,v}}(o)$ for every $u, v \in S$ with $u \neq v$, and sort the pairs of points with respect to the slopes of the corresponding lines in $\mathcal{L}$. In contrast with the "simple" point set examined in the previous paragraphs, we now end up with a smaller set of h' distinct slopes, where $h' < h$. In addition, these new distinct slopes partition the h pairs of points into h' disjoints sets $E_1, E_2, \ldots, E_{h'}$, each containing pairs of points having identical slopes for their corresponding lines in $\mathcal{L}$. From each set E_i, we can select an arbitrary pair of points, say (u_i, v_i), $1 \leq i \leq h'$, as the *representative pair*. We note that sets E_i, $1 \leq i \leq h'$, can be computed in $O(n^2 \log n)$ time by simply sorting the pairs of points with respect to the slopes of their corresponding lines in $\mathcal{L}$; pairs of identical slope end up consecutive after sorting. We also observe that each set E_i with $1 \leq i \leq h'$ induces a graph $G_i = (V_i, E_i)$ whose vertex set V_i contains exactly the points involved in the pairs of E_i. In addition, observe that each graph G_i consists of k_i connected components each of which is a clique and corresponds to points lying on the same line perpendicular to $\overline{l_{u_i,v_i}}(o)$ and, moreover, the projection of the points of each of these k_i connected components on $d_i(o)$ and $d_{i+1}(o)$ do not overlap. In the example of Fig. 17, we have that $V_i = \{1, \ldots, 9\}$ and the three connected components (cliques) of V_i are $V_{i,1} = \{1, 2, 3, 4\}$, $V_{i,2} = \{5, 6\}$ and $V_{i,3} = \{7, 8, 9\}$.

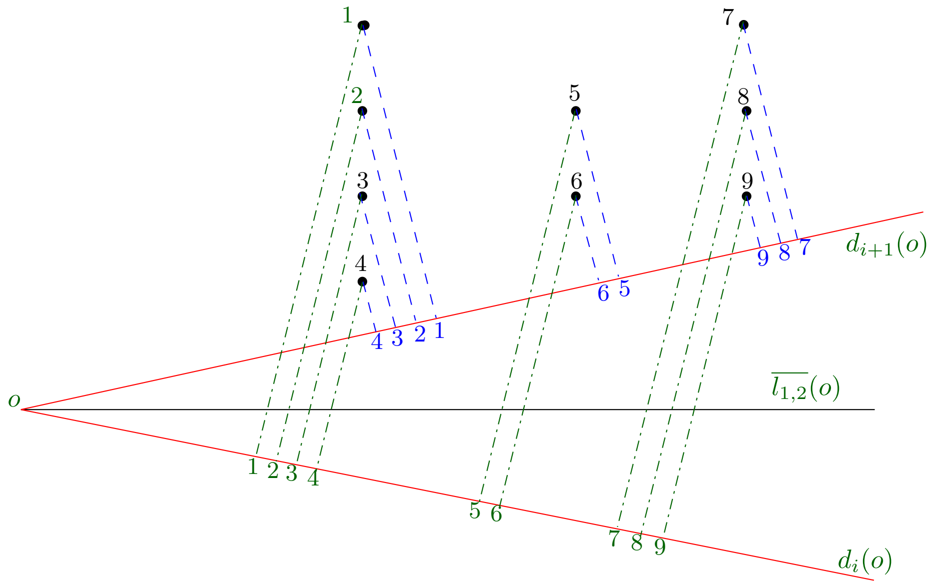


Fig. 17: Point set $S = \{1, 2, \ldots, 9\}$ consists of three sets of collinear points that form pairs having the line passing through them perpendicular to $\overline{l_{1,2}}(o)$ and its orderings $\text{ord}(S, d_i)$ and $\text{ord}(S, d_{i+1})$ on $d_i(o)$ and $d_{i+1}(o)$, respectively.

Note that the connected components of all graphs G_i with $1 \leq i \leq h'$ can be computed using depth first search in $O(n^2)$ total time since there are exactly $\binom{n}{2}$ edges in all graphs together.

Thus, for the case of a point set containing pairs of points that lie on parallel lines, we can define the set $\sigma' = \{d_1, d_2, \ldots, d_{h'}\}$ of directions by considering only the h' distinct slopes of the lines in $\mathcal{L}$. It remains, however, to describe how we compute for two consecutive arbitrary directions d_i and d_{i+1}, $1 \leq i < h'$, $\text{ord}(S, d_{i+1})$ from $\text{ord}(S, d_i)$. Let $\overline{l_{u_i,v_i}}(o)$ be the line separating d_i and d_{i+1} where (u_i, v_i) is the representative pair of E_i.

Observe that all collinear points lying on a line perpendicular to $\overline{l_{u_i,v_i}}(o)$ appear in reverse order in $d_{i+1}(o)$ compared to the order they appear in $d_i(o)$; see Fig. 17. Thus, in order to compute $\text{ord}(S, d_{i+1})$ from $\text{ord}(S, d_i)$, we have simply to identify these points. Of course, this has to be repeated for all different perpendicular lines to $\overline{l_{u_i,v_i}}(o)$ that contain at least a pair of points. However, we have already computed this information. The sets of collinear points perpendicular to $\overline{l_{u_i,v_i}}(o)$ correspond to the vertex sets $V_{i,j}$ with $1 \leq j \leq k_i$ of the k_i connected components of the graph G_i. Thus, the extra cost for computing $\text{ord}(S, d_{i+1})$ from $\text{ord}(S, d_i)$ is $O(V_i)$, which amounts to the reversion of the order of the points in the projections. We conclude that all such reversions can be computed in $O(n^2)$ total time since the total number of edges in all computed graphs equals $\binom{n}{2}$. $\square$

Theorem 6. *Given a set S of n points, a solution to the* MMST$(S, 2)$ *problem can be computed in $O(n^6)$ time.*

Proof. Let $\sigma = \langle d_1, d_2, \ldots, d_h \rangle$ be the circular sequence of directions with $h \leq \binom{n}{2}$ as defined in the proof of Theorem 5. Recall that we can compute σ in $O(n^2 \log n)$ time. By applying Theorem 4 to every pair of distinct directions $d, d' \in \sigma$, we consider a candidate $\mathcal{D}$-monotone tree for each set $\mathcal{D} = \{d, d'\}$ for which S is in $\mathcal{D}$-general position. Since there are $\binom{h}{2} \in O(n^4)$ pairs, this takes $O(n^6)$ time. To complete the proof we show that restricting to sets $\mathcal{D}$ for which S is in $\mathcal{D}$-general position is sufficient, namely we prove that if T is a $\mathcal{D}$-monotone spanning tree for a set $\mathcal{D} = \{d, d'\}$ and S is not in d-general position, then there is a set $\mathcal{D}' = \{d'', d'\}$ such that S is in d''-general position and T is $\mathcal{D}'$-monotone. This implies that if T is a $\mathcal{D}$-monotone spanning tree and S is not in $\mathcal{D}$-general position, then there is also a set $\mathcal{D}'$ such that S is in $\mathcal{D}'$-general position and T is $\mathcal{D}'$-monotone (with the previous reasoning for one or both the directions in $\mathcal{D}$).

Let T be a $\mathcal{D}$-monotone spanning tree, where $\mathcal{D} = \{d, d'\}$. If S is not in d-general position, the orthogonal projection of S on a line parallel to d defines a sequence $\alpha = \langle p_1, p_2, \ldots, p_{n'} \rangle$ of points with $n' < n$. Some points of α correspond to the projection of multiple points of S; each of these points is called a *multiple* point. By slightly rotating d, we can obtain a direction d'' such that: (i) the projections of the points of S that correspond to the same multiple point in α form a consecutive subsequence of $\mathrm{ord}(S, d'')$; and (ii) replacing each such subsequence with a single point, we obtain α. Finally, we show that T is $\mathcal{D}'$-monotone, where $\mathcal{D}' = \{d'', d'\}$. Let P be a path between two points u and v in T; if P is d-monotone then the points of P are in d-general position and thus the orthogonal projections of all points of P correspond to distinct (non-multiple) points of α. Hence the points in P are also in d''-general position, and P is also d''-monotone. □

Arguing as in the proof of Theorem 6, we get the following result for any k.

Theorem 7. *Given a set S of n point and any positive integer k, there exists a function $f \colon \mathbb{N} \to \mathbb{N}$ such that we can compute a minimum k-directional monotone spanning tree of S in $O(f(k) \cdot n^{2k(2k-1)} \log n)$ time.*

Proof. Let $\sigma = \langle d_1, d_2, \ldots, d_h \rangle$ be the circular sequence of directions with $h \leq \binom{n}{2}$ as defined in the proof of Theorem 5, and which can be computed in $O(n^2 \log n)$ time. By applying Theorem 2 to every set of k distinct directions in σ, we consider all candidate $\mathcal{D}$-monotone trees over all sets $\mathcal{D}$ of k directions. Since there are $\binom{h}{k} \in O(n^{2k})$ sets, this takes $O(f(k) \cdot n^{2k(2k-1)} \log n)$ time. By using an exchange argument as in the proof of Theorem 6, it can be proven that it is sufficient to restrict ourselves to sets $\mathcal{D}$ of k directions for which S is $\mathcal{D}$-monotone. The statement follows. □

D Additional Material for Section 6

Lemma 15. *Let T be a solution to the* MMST$(S_k, \mathcal{D})$ *problem and let $x \in S_k \setminus \{o\}$ be a polygon vertex having* $\deg(x) \geq 2$*. Then, for any two edges (x, u_1)*

and (x, u_2) that are consecutive in counter clockwise order around x in T and form an angle smaller than π, it holds that $\angle u_1 x u_2 = \frac{\pi}{2k}$. Equivalently, since the angle formed at x by the edges from x to any two consecutive polygon vertices is equal to $\frac{\pi}{2k}$, u_1 and u_2 are either consecutive polygon vertices, or one of them, say u_1, coincides with o and u_2 is the vertex following the anti-diametric of x in ccw. order.

Proof. Consider the wedges in $\mathcal{W}_D(x)$ as defined by the lines $\overline{d_1}(x), \overline{d_2}(x), \ldots, \overline{d_k}(x)$; see Fig. 18a. Observe that the wedges partition the circle on which the polygon vertices lie into k distinct circular arcs of equal length. To see this, consider an arbitrary wedge and let y and z be the points it intersects the unit circle centered at o. Then, angle $\angle yxz = \frac{\angle yoz}{2}$ since x is on the circle with center o and, thus, the circular arcs formed by the wedges are also formed by equal angles at the center of the unit polygon. As a result, the wedges at x also partition the polygon vertices into k distinct sets, each consisting of two vertices, since vertices are all equally distanced on the circle; see Fig. 18a.

Assume, for the sake of contradiction, that $\angle u_1 x u_2$ is greater than $\frac{\pi}{2k}$. That is, either u_1 and u_2 are non-consecutive vertices of the polygon or one of them, say u_1, coincides with o and u_2 is not the vertex ccw. to the anti-symmetric of x with respect to o. We consider these two cases separately.

Case 1: u_1 and u_2 are two non-consecutive polygon vertices (Fig. 18b). In this case, we define a circular sector C (gray in Fig. 18b) by rotating the line l_{x,u_1} counter-clockwise around x untill it coincides with the line l_{x,u_2}. The sector C partitions S_k into two subsets, namely, the set S_C of points lying in C and the set $S_k \setminus S_C$. All points in S_C are connected to T through some path either to u_1 or to u_2, since edges (x, u_1) and (x, u_2) are two consecutive around x edges (in ccw. order). Now, let $w_1, w_2 \in S_C$ and $P_1 = \langle x, u_1, \ldots, w_1 \rangle$ and $P_2 = \langle x, u_2, \ldots, w_2 \rangle$ be two paths in T, such that w_1 is the last polygon vertex connected to u_1 in ccw. order and w_2 is the last polygon vertex connected to u_2 in clockwise order. Note that one of w_1 or w_2 may coincide with u_1 or u_2, respectively. Observe that w_1 and w_2 are two consecutive vertices of the polygon. If w_1 and w_2 were not consecutive, then there would be a point y between w_1 and w_2 which would not be connected to T. Also, note that w_1 and w_2 should lie in different wedges of $\mathcal{W}_\mathcal{D}(x)$, since otherwise, due to Lemma 2, path $\langle w_1, \ldots, x, \ldots w_2 \rangle$ would not be monotone. Let $\overline{d_i}(x)$, for some direction $d_i \in \mathcal{D}, 1 \leq i \leq k$, be the line passing between w_1 and w_2. Then, path $P = \langle w_1, \ldots, x, \ldots, w_2 \rangle$ must be d_i-monotone. To see that, observe that since path P is d-monotone with respect to at least one direction $d \in \mathcal{D}$, its endpoints must lie on different sides of $\overline{d}(x)$. But, given that only a single line through x that is perpendicular to a direction in $\mathcal{D}$ can pass between w_1 and w_2, we conclude that path P is d_i-monotone. Thus, all vertices in P lie between the parallel lines $\overline{d_i}(w_1)$ and $\overline{d_i}(w_2)$.
This is a contradiction since the strip bounded by lines $\overline{d_i}(w_1)$ and $\overline{d_i}(w_2)$ contains only x and, possibly, o, but definitely neither u_1 nor u_2. Note that the case where w_1 coincides with u_1 is similar, since then all vertices of path

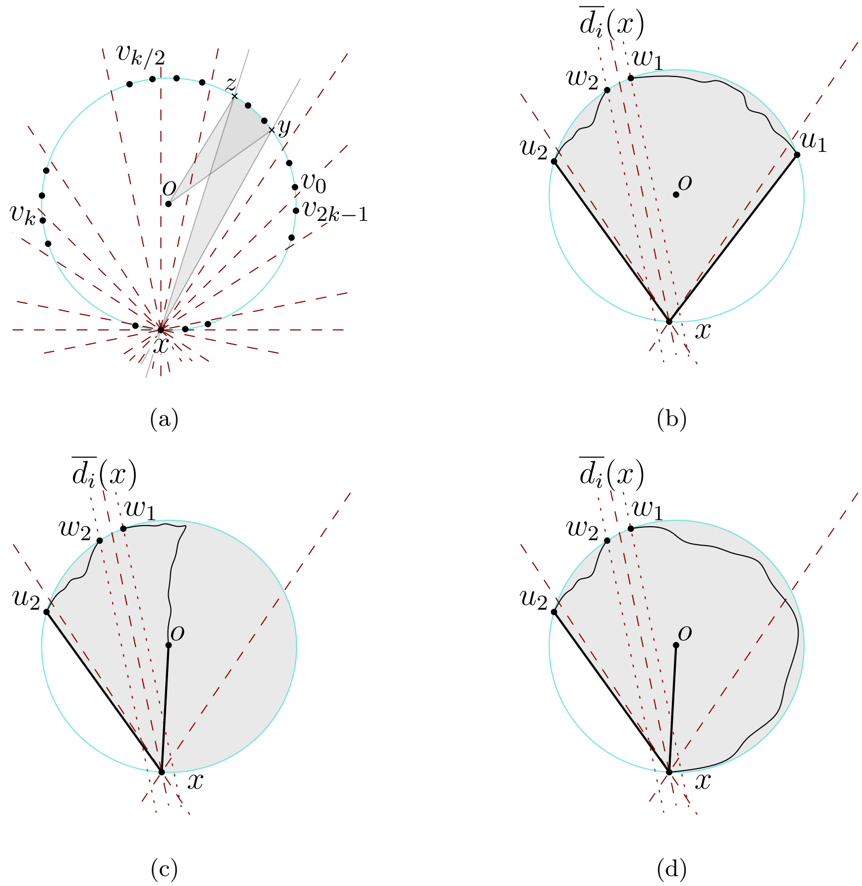


Fig. 18: (a) All but one wedges in $\mathcal{W}_{\mathcal{D}}(x)$ contains two polygon vertices. (b-d) The path setting exploited in the proof of Lemma 15.

P should lie between lines $\overline{d_i}(u_1)$ and $\overline{d_i}(w_2)$. This is again, a contradiction, because the strip bounded by these two lines does not contain vertex u_2.

Case 2: One of u_1, u_2 coincides with o, say u_1, and u_2 is not the vertex following the anti-diagonal of x in ccw. order. Line l_{x,u_2} partitions the point set into two subsets, namely, set S_C consisting of all points located on the same side of l_{x,u_2} as point o is, and set $S_k \setminus S_C$. Observe that all vertices in S_C connect to T through some path either to x or o or u_2. We distinguish the following cases:

Case 2a: There is at least one vertex in S_C that is connected to T through o (Fig. 18c). Again, let $w_1, w_2 \in S_C$ and $P_1 = \langle x, o, \dots, w_1 \rangle$ and $P_2 = \langle x, u_2, \dots, w_2 \rangle$ be two paths in T such that w_1 is the last polygon vertex connected to o in ccw. order and w_2 is the last polygon vertex connected to u_2 in clockwise order. As before, w_1 and w_2 are

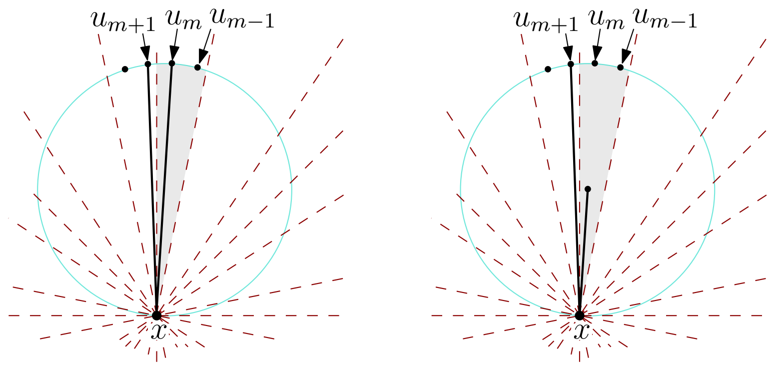


Fig. 19: Connections of degree-two vertices in the proof of Lemma 12

two consecutive polygon vertices lying in different wedges of $\mathcal{W}_{\mathcal{D}}(x)$ and separated by $\overline{d_i}(x)$, for some direction $d_i \in \mathcal{D}, 1 \leq i \leq k$. Then, path $P = \langle w_1, \ldots, o, x, u_2, \ldots, w_2 \rangle$ must be d_i-monotone, which only happens if points o, x, u_2 lie in the strip bounded by lines $\overline{d_i}(w_1)$ and $\overline{d_i}(w_2)$. This is a contradiction since the only vertices in the strip are x and, possibly, o.

Case 2b: Vertex o is a leaf in T (Fig. 18d). Define S_C as in Case 2a. Then, all points in S_C are connected to T only through x and u_2. Let $w_1, w_2 \in S_C$ and $P_1 = \langle x, \ldots, w_1 \rangle$ and $P_2 = \langle x, u_2, \ldots, w_2 \rangle$ be two paths such that w_1 is the last polygon vertex connected to x in ccw. order and w_2 is the last polygon vertex connected to u_2 in clockwise order. Again, w_1 and w_2 are two consecutive polygon vertices lying in different wedges of $\mathcal{W}_{\mathcal{D}}(x)$ and separated by $\overline{d_i}(x)$, for some direction $d_i \in \mathcal{D}, 1 \leq i \leq k$. Then, path $P = \langle w_2, u_2 \ldots, x, \ldots, w_1 \rangle$ is d_i-monotone only if u_2 and x both lie in the strip bounded by lines $\overline{d_i}(w_1)$ and $\overline{d_i}(w_2)$; a contradiction. □

Lemma 12 ($\star$). *Let T be a solution to the* MMST$(S_k, \mathcal{D})$ *problem, and let x be a polygon vertex. Then,* $\deg_T(x) \leq 2$.

Proof. Refer to Fig. 19a. Consider an arbitrary wedge of $\mathcal{W}_{\mathcal{D}}(x)$. The wedge contains exactly two (consecutive) polygon vertices, say u_{m-1}, u_m. Due to Property 2, x is not connected in T with both u_{m-1} and u_m; otherwise, the path $\langle u_{m-1}, x, u_m \rangle$ in T would not be $\mathcal{D}$-monotone. Given that for every three consecutive polygon vertices exactly two of them lie in the same wedge, they can't be all three incident to x in T. Then, by Lemma 15, x can have at most two neighbors.

Consider now the case where x is connected with o (Fig. 19b). The wedge of $\mathcal{W}_{\mathcal{D}}(x)$ which contains o also contains two polygon vertices u_{m-1} and u_m that cannot be incident to x in T. One of them, say u_m is anti-diametric to x. Then, x can only be connected to the polygon vertex u_{m+1} lying in the neighboring wedge of the one containing o and, thus, x can have degree at most two. □

Theorem 3 (⋆). *The only solution to the* $\mathrm{MMST}(S_k, \mathcal{D})$ *problem is the star* $T^\star$ *with center* o *and* $\deg_{T^\star}(o) = 2k \in \Omega(|S_k|)$.

Proof. Let tree T be a solution to the $\mathrm{MMST}(S_k, \mathcal{D})$ problem and assume that T in not the $2k$-star with o at its center. We first show that a vertex of degree one cannot be the endpoint of a chord. Let u be an arbitrary polygon vertex that is a leaf of T and let (u, w) be the edge of T that is incident to u. If edge (u, w) was a chord, then vertices on both sides of (u, w) would connect in T through w and, thus, $\deg(w) \geq 3$. However, this is not possible since, by Lemma 12, we have that $\deg(w) \leq 2$. Therefore, u is the endpoint of either a ray or an external edge. We further observe that, as T is not the $2k$ star, T contains at least one external edge which has as an endpoint a polygon vertex of degree one. If this was not the case, then the polygon vertices of degree two together with o would form one or more cycles, contradicting the acyclicity of T.

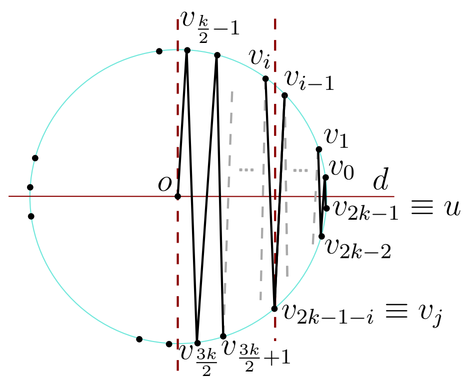


Fig. 20: The path setting exploited in the proof of Theorem 3.

Consider now a polygon vertex u of degree one that is connected to T with an external polygon edge. For easiness of presentation, we rotate the point set (and we renumber the vertices accordingly) so that u coincides with vertex v_{2k-1} of Fig. 8a. Note that since we have assumed that k is even, it holds that $\mathcal{W}_{\mathcal{D}} = \mathcal{W}_{\overline{\mathcal{D}}}$ and thus, one of the directions of $\mathcal{D}$, say d, is horizontal.

We now examine how u, which after the rotation and the renumbering is referred to as v_{2k-1}, is connected to o in T. Refer to Fig. 20. Vertex v_{2k-1} cannot be connected in T to v_{2k-2} through the external edge (v_{2k-1}, v_{2k-2}). If it was, then, due to Lemma 15, v_{2k-2} must be in turn adjacent to v_0. But then, both v_0 and v_{2k-1} fall in the same wedge of $\mathcal{W}_{\mathcal{D}}(v_{2k-2})$ and, thus, the path $\langle v_{2k-1}, v_{2k-2}, v_0 \rangle$ cannot be monotone. Therefore, v_{2k-1} is connected with v_0. Then, due to Lemma 15, v_0 is adjacent to v_{2k-2} which, in turn, is adjacent to v_1, and so on. This path, continues until we reach the center o. To see this, note that if the path ends before reaching o then it ends at a polygon vertex of degree one. This contradicts the fact that T is a connected spanning tree of S_k. In addition, o is reached through edge $(v_{\frac{k}{2}-1}, o)$. If this was not the case and o was adjacent in this path to a vertex w which was before $v_{\frac{k}{2}-1}$ in ccw. order, the angle formed at w by (o, w) and its preceding edge in the path would be greater than $\frac{\pi}{2k}$, which is impossible due to Lemma 15. Thus, T contains the path P that starts at v_{2k-1}, ends at o, and contains all points to the right of the vertical line through o. We will show that T cannot be of minimum length.

Consider tree T' formed by substituting the edges of path P by rays from o to the path vertices. Obviously T' is also monotone. To show that T is not optimal, it suffices to show that the length of path P is greater than the total length of the rays that substituted the edges of P in T'. In other words it suffices to show that $\|P\| > k$, where $\|P\|$ denotes the length of path P. For the length of path P, we have that $\|P\| = 1 + \sum_{i=1}^{k-1} 2\sin\left(\frac{\pi}{2k} i\right) = \cot\left(\frac{\pi}{4k}\right) > k$ (Lemma 16 and Lemma 17 in Appendix D). Thus, tree T is not of minimum length; a contradiction. We conclude that T is the $2k$-star centered at o. □

Lemma 16. $\sum_{i=1}^{k-1} 2\sin(\frac{\pi}{2k} i) = \cot(\frac{\pi}{4k}) - 1.$

Proof. For a sum of sine series we know from [34] that

$$\sum_{i=1}^{n} \sin(a + (i-1)b) = \sin\left(a + \frac{n-1}{2} b\right) \frac{\sin(\frac{nb}{2})}{\sin(\frac{b}{2})} \tag{1}$$

Additionally, from trigonometry we know that

$$\sin(a-b) = \sin(a)\cos(b) - \cos(a)\sin(b) \tag{2}$$

Be utilizing the above equations, we get:

$$\begin{aligned}
2\sum_{i=1}^{k-1} \sin\left(\frac{\pi}{2k} i\right) &= 2\sum_{i=1}^{k-1} \sin\left(\frac{\pi}{2k} + (i-1)\frac{\pi}{2k})\right) \\
&\overset{(1)}{=} 2\left[\sin\left(\frac{\pi}{2k} + \frac{(k-2)\pi}{4k}\right) \frac{\sin\left(\frac{(k-1)\pi}{4k}\right)}{\sin\left(\frac{\pi}{4k}\right)}\right] \\
&\overset{(2)}{=} 2\left[\sin\left(\frac{\pi}{4}\right) \frac{\sin(\frac{\pi}{4})\cos(\frac{\pi}{4k}) - \cos(\frac{\pi}{4})\sin(\frac{\pi}{4k})}{\sin(\frac{\pi}{4k})}\right] \\
&= \frac{\cos(\frac{\pi}{4k}) - \sin(\frac{\pi}{4k})}{\sin(\frac{\pi}{4k})} \\
&= \cot\left(\frac{\pi}{4k}\right) - 1
\end{aligned}$$

□

Lemma 17. $\cot(\frac{\pi}{4k}) > k$*, for* $k > 1$*.*

Proof. From the Cusa–Huygens inequality (see [41]), we know that

$$\frac{\sin(x)}{x} < \frac{2+\cos(x)}{3}, \quad 0 < x < \frac{\pi}{2} \tag{3}$$

Observe that

$$\begin{aligned} &\cot\left(\frac{\pi}{4k}\right) > k \\ \Rightarrow &\frac{\cos\left(\frac{\pi}{4k}\right)}{\sin\left(\frac{\pi}{4k}\right)} > k \\ \Rightarrow &\cos\left(\frac{\pi}{4k}\right) > k\sin\left(\frac{\pi}{4k}\right), \text{ since } \sin\left(\frac{\pi}{4k}\right) > 0 \\ &\Rightarrow \cos(u) > \frac{\pi}{4}\cdot\frac{\sin(u)}{u}, \text{ if we substitute } u = \frac{\pi}{4k} \end{aligned}$$

Note that if $u = \frac{\pi}{4k} \Rightarrow k = \frac{\pi}{4u}$ then $k \geq 2 \Rightarrow \frac{\pi}{4k} \leq \frac{\pi}{8} \Rightarrow u \leq \frac{\pi}{8}$. Additionally, from Eq. (3) we get

$$\frac{\pi}{4}\frac{\sin(u)}{u} < \frac{\pi}{4}\cdot\frac{2+\cos(u)}{3}$$

It is now sufficient to show that

$$\frac{\pi}{4}\cdot\frac{2+\cos(u)}{3} < \cos(u), \text{ for all } u \leq \frac{\pi}{8}.$$

This implies that

$$\cos(u) > \frac{2\pi}{12-\pi} \quad \Rightarrow \quad \cos(u) > \cos(0.78) \quad \Rightarrow \quad u < 0.78 \approx 44.8^\circ$$

Therefore, the inequality holds for every $u \leq \pi/8$ and $k \geq 2$. □