On Minimum- and Maximum-Weight Minimum Spanning Trees with Neighborhoods

University of Waterloo Technical Report CS-2012-14

Reza Dorrigiv[*,1], Robert Fraser[† ,2], Meng He[‡,1], Shahin Kamali[§ ,2], Akitoshi Kawamura[¶,3], Alejandro López-Ortiz[‖,2], and Diego Seco[**,4]

[1] *Dalhousie University, Halifax, Canada*
[2] *University of Waterloo, Waterloo, Ontario, Canada*
[3] *University of Tokyo, Tokyo, Japan*
[4] *University of A Coruña, A Coruña, Spain*

August 27, 2012

Abstract

We study optimization problems for the Euclidean minimum spanning tree (MST) on imprecise data. To model imprecision, we accept a set of disjoint disks in the plane as input. From each member of the set, one point must be selected, and the MST is computed over the set of selected points. We consider both minimizing and maximizing the weight of the MST over the input. The minimum weight version of the problem is known as the minimum spanning tree with neighborhoods (MSTN) problem, and the maximum weight version (MAX-MSTN) has not been studied previously to our knowledge. We provide deterministic and parameterized approximation algorithms for the MAX-MSTN problem, and a parameterized algorithm for the MSTN problem. Additionally, we present hardness of approximation proofs for both settings.

1 Introduction

We consider geometric problems dealing with imprecise data. In this setting, each point of the input is provided as a *region of uncertainty*, i.e., a geometric object such as a line, disk, set of points, etc., and the exact position of the point may be anywhere in the object. Each object is understood to represent the set of possible positions for the corresponding point. In our work, we

[*] rdorrigiv@cs.dal.ca
[†] r3fraser@cs.uwaterloo.ca
[‡] mhe@cs.dal.ca
[§] s3kamali@cs.uwaterloo.ca
[¶] kawamura@is.s.u-tokyo.ac.jp
[‖] alopez-o@cs.uwaterloo.ca
[**] dseco@udc.es

study the Euclidean minimum spanning tree (MST) problem. Given a tree T, we define its weight $w(T)$ to be the sum of the weights of the edges in T. For a set of fixed points P in Euclidean space, the weight of an edge is the distance between the endpoints, and we write $mst(P)$ for the weight of the MST on P. Thus, $mst(P) = \min w(T)$, where the minimum is taken over all spanning trees T on P.

Given a set of disjoint disks as input, we wish to determine the minimum and maximum weight MSTs possible when a point is fixed in each disk. The minimum weight MST version of the problem has been studied previously, and is known as the minimum spanning tree with neighborhoods problem (MSTN). This paper introduces the maximum weight MST version of the problem, which we call the MAX-MSTN problem. Assume we are given a set $D = \{D_1, \ldots, D_n\}$ of disjoint disks in the plane, i.e., $D_i \cap D_j = \emptyset$ if $i \neq j$. The MSTN problem on D asks for the selection of a point $p_i \in D_i$ for each $D_i \in D$ such that the weight of the MST of the selected points is minimized. Similarly, MAX-MSTN asks for a selection of p_i such that the weight of the MST of the selected points is maximized.

1.1 Related Work

The first known MST algorithm was published over 80 years ago [10], and a number of successful variants have followed (see [7] for the history of the problem). A review of models of uncertainty and data imprecision for computational geometry problems is provided in [9]. Here, we discuss a few results that are directly related to the MST problem and our model of imprecision.

The MSTN problem on unit disks has been shown to admit a PTAS [13]. A hardness proof for a generalization of MSTN where the neighborhoods are either disks or rectangles appeared in [13], but the proof was faulty. One of the authors later conjectured that a reduction from planar 3-SAT might be used to show the hardness of the MSTN problem [12, p.106]. In Section 3.2, we prove this conjecture.

Löffler and van Kreveld [9] demonstrated that it is algebraically difficult to compute the MST when the regions of uncertainty are continuous regions of the plane, even for very simple inputs such as disks or squares, as the solution may involve the roots of high degree polynomials. It is of independent interest to see if the problem is combinatorially difficult. In the same paper, authors proved that the MSTN problem is (combinatorially) NP-hard if the regions of uncertainty are not pairwise disjoint, through a reduction from the minimum Steiner tree problem. In this paper we prove the hardness of the special case in which the regions are pairwise disjoint.

Erlebach et al. [5] used a model of uncertainty where information regarding the weight of an edge between a pair of points or the position of a point may be obtained by pinging the edge or vertex, and they sought to minimize the number of pings required while obtaining the optimal solution. The distinction is that in their work, they were interested in reducing the amount of communication that is required to locate points within a region of uncertainty, while in our model, the objective is to optimize the MST given regions of uncertainty.

Researchers have considered other related problems that deal with imprecise data. The travelling salesman with neighborhoods (TSPN) problem has been studied extensively. The problem was introduced by Arkin and Hassin [1], in a paper that has been applied, improved, built-upon or otherwise referenced over 150 times. There exists a PTAS for TSPN when the neighborhoods are disjoint unit disks [4]. The most general version of the problem, where regions may overlap and may have varying sizes, is known to be APX-hard [2]. The problem of maximizing the smallest pairwise distance in a set of n points with neighborhoods has also been studied and proved to be NP-hard [6].

1.2 Our Results

We present a variety of results related to the MSTN and MAX-MSTN problems. For both problems we assume the regions of uncertainty (disks) are disjoint.

- MAX-MSTN: deterministic 1/2-approximation;
- MAX-MSTN: parameterized $1 - \frac{2}{k+4}$-approximation (where k represents the separability of the instance, which is to be defined later);
- MAX-MSTN: proof of hardness of approximation;
- MSTN: parameterized $1 + 2/k$-approximation (k is the separability of the instance);
- MSTN: proof of hardness of approximation.

The deterministic approximation algorithm for MAX-MSTN (Section 2.1) is based on choosing the center points of the disks; the interesting aspect in this section lies in the analysis. The parameterized algorithms (Sections 2.2 and 3.1) for both settings were inspired by the observation that the approximation factor improves rapidly as the distance between disks increases. To address this, we introduce a measure of how much separation exists between the disks, which we call *separability*, and we analyze the approximation factor of the MST on disk centers with respect to separability.

For both hardness of approximation results, we establish that there is no FPTAS for the problems unless P=NP. Although the hardness proofs both consist of reductions from planar 3-SAT, the gadgets used are quite distinct and either reduction is interesting even given the existence of the other. In both cases, we construct an instance of our problem from the planar 3-SAT instance, and show that it is possible to compute the weight of the optimal solution to our problem assuming that the 3-SAT instance is satisfiable. If the instance is not satisfiable, we prove that the weight is changed by at least a constant amount (reduced by at least 0.33 units for MAX-MSTN, and increased by at least 0.84 units for MSTN).

2 MAX-MSTN

In this section we study a couple of approximation algorithms for the MAX-MSTN problem, and then we present the proof of hardness of approximation. We begin with a deterministic algorithm below, followed by a parameterized algorithm in Section 2.2.

2.1 Deterministic 1/2-Approximation Algorithm

To approximate the solution to MAX-MSTN, we first consider the algorithm that builds an MST on the centers of the disks. We show this algorithm approximates the optimal solution within a

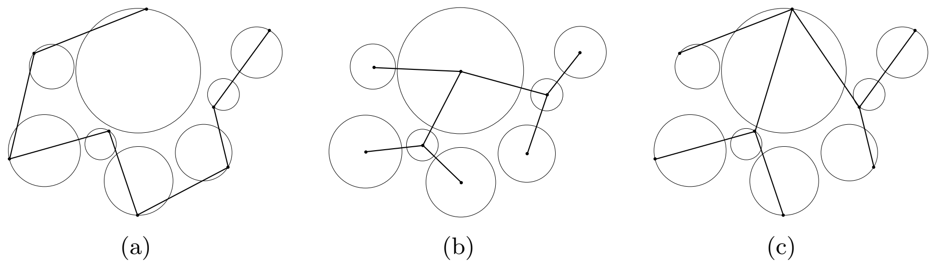


Figure 1: To compare $w(T_c)$ with $w(T_{\text{opt}})$, we use an intermediate tree T_c'. (a) The optimal result for MAX-MSTN (T_{opt}). (b) The MST T_c on centers. (c) The spanning tree T_c' with the same topology as T_c, using the points of T_{opt}.

factor of 1/2, i.e., the weight of the MST built on the centers is not smaller than half of that of the optimal tree.

Theorem 2.1. *Consider the* MAX-MSTN *problem for a set D of disjoint disks. Let T_c denote the MST on the centers of the disks, and let T_{opt} be the maximum MST (i.e., the optimal solution to the problem). Then $w(T_c) \geq 1/2 \cdot w(T_{opt})$.*

Proof. Let T_c' be the spanning tree (not necessarily an MST) with the same topology (i.e., combinatorial structure of the tree) as T_c but on the points of T_{opt} (see Figure 1). Since T_c' and T_{opt} span the same set of points, and T_{opt} is an MST, we have $w(T_{\text{opt}}) \leq w(T_c')$. On the other hand, since T_c' and T_c have the same topology, we have $w(T_c') \leq 2w(T_c)$; this is because when we move the points from the center to somewhere else in the disks, the weight of each edge increases by at most the sum of the radii of the two involved disks and, since the disks are disjoint, the increase is at most equal to the original weight. To summarize, we have $w(T_{\text{opt}}) \leq w(T_c')$ and $w(T_c') \leq 2w(T_c)$, which completes the proof. □

2.2 Parameterized $1 - \frac{2}{k+4}$-Approximation Algorithm

Observe that in order to get the approximation algorithm for MAX-MSTN in Section 2.1, we require disks to be disjoint. Intuitively, if we know that disks are further apart, we can get better approximation ratios. We formalize this intuition by providing a parameterized analysis, i.e., we express the performance of the algorithm in terms of a *separability parameter*[1]. Let $r_{\max}$ be the maximum radius of our disks. We say that a given input for our problem satisfies *k-separability* if the minimum distance between any two disks is at least $k \cdot r_{\max}$. The separability of an input instance I is defined as the maximum k such that I satisfies k-separability. With this definition, we have the following result:

Theorem 2.2. *For* MAX-MSTN *when the regions of uncertainty are disjoint disks with separability parameter $k > 0$, the algorithm that builds an MST on the centers of the disks achieves a constant approximation ratio of $\frac{k+2}{k+4} = 1 - \frac{2}{k+4}$.*

[1] Separability is similar in spirit to the notion of a well-separated pair; see [3].

Proof. Let T_c be the MST on the centers of the disks. We can extend the analysis in the proof of Theorem 1 to show that the approximation factor is $\frac{k+2}{k+4} = 1 - \frac{2}{k+4}$ for any input that satisfies k-separability. Define T_{opt} and T'_c as before. Consider an arbitrary edge e in T'_c and let D_i and D_j be the two disks connected by e. Let r_i and r_j be the radii of D_i and D_j, respectively, and let d be the distance between D_i and D_j. In T_c the disks D_i and D_j are connected by an edge e' whose weight is $d + r_i + r_j$. The weight of e, on the other hand, can be at most $d + 2r_i + 2r_j$. Therefore, the ratio between the weight of an edge in T_c and its corresponding edge in T'_c is at least

$$\frac{d + r_i + r_j}{d + 2r_i + 2r_j} \geq \frac{kr_{\max} + r_i + r_j}{kr_{\max} + 2r_i + 2r_j} \geq \frac{kr_{\max} + r_{\max} + r_{\max}}{kr_{\max} + 2r_{\max} + 2r_{\max}} = \frac{k+2}{k+4}.$$

Since this holds for any edge of T'_c, we get $w(T_c) \geq \frac{k+2}{k+4} w(T'_c) \geq \frac{k+2}{k+4} w(T_{\text{opt}})$, and we get an approximation factor of $\frac{k+2}{k+4}$. □

The approximation ratio gets arbitrarily close to 1 as k increases. This confirms our intuition that if the disks are further apart (more separate), we get a better approximation factor.

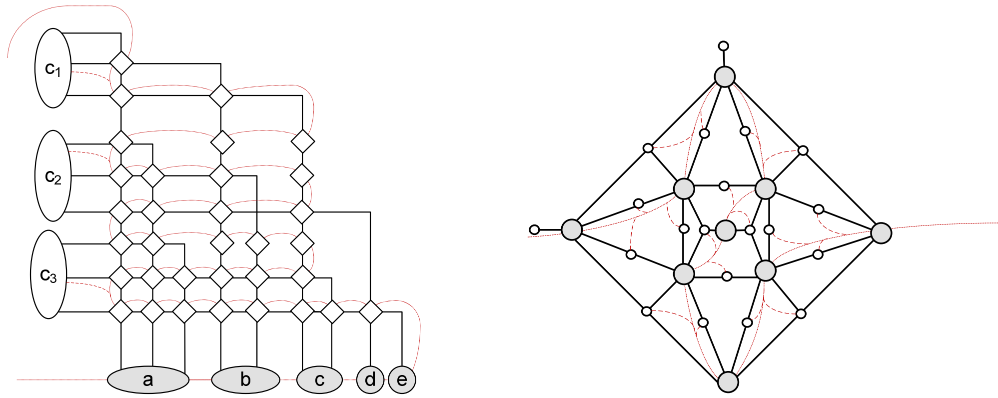
2.3 Hardness of Approximation

We present a hardness proof for the MAX-MSTN problem by a reduction from the planar 3-SAT problem [8]. Planar 3-SAT is a variant of 3-SAT in which the graph $G = (V, E)$ associated with the formula is planar.

Theorem 2.3. MAX-MSTN *does not admit an FPTAS unless* P=NP.

We show a reduction from any instance of the planar 3-SAT problem to the MAX-MSTN problem. In planar 3-SAT, we have a planar bipartite graph $G = (V, E)$, where $V = V_v \cup V_c$, so that there is a vertex in V_v for each variable and a vertex in V_c for each clause; there is an edge (v_i, v_j) in E between a variable vertex $v_i \in V_v$ and a clause vertex $v_j \in V_c$ if and only if the clause contains a literal of that variable in the 3-SAT instance. In [8] it was shown that the planar 3-SAT problem is NP-hard via a reduction from the standard 3-SAT problem. Further, it was observed that the resulting instance of planar 3-SAT permits the construction of a path $P = (V_v, E_P)$ using a set of edges E_P such that $E \cap E_P = \emptyset$, where P is connected and passes through all vertices in V_v without crossing any edge in E. We call this path P the *spinal path*. We further observe that additional edges can be added to P to get a *spinal tree* T which also covers clause vertices V_c. In this sense T will be a tree that covers all vertices without crossing an edge of G such that all vertices corresponding to clauses are leaves. These observations are illustrated in Figure 2. To prove the hardness of MAX-MSTN, we make use of the spinal tree.

To begin with, we study an important theorem for our reduction, which states that an optimal MAX-MSTN solution in a *chain* of unit disks has a characteristic zig-zag pattern. A chain is a set of k unit disks $D^c = \{D^c_1, \ldots, D^c_k\}$ whose centers are collinear, incident upon a horizontal line L_{hz}, and adjacent disk centers are $2d$ (with $d \geq 1$) units distant from each other. Furthermore, there are two terminal points T_L and T_R incident upon L_{hz}, where T_L is located d units to the left of the center of the leftmost disk, and T_R is located d units to the right of the center of the rightmost disk of the chain.



(a) The reduction from 3-SAT to planar 3-SAT.

(b) The planar gadget located in the intersection points. Variable and clause vertices are represented by large and small circles, respectively.

Figure 2: The reduction from 3-SAT to planar 3-SAT as presented in [8]. The variable and clause vertices of 3-SAT are located respectively in x and y axis, and the edges are drawn as orthogonal paths (a). A planar gadget is placed on each intersection point. Each gadget includes some new variable and clause vertices (b). In [8], it is observed that there is a path (we call it the spinal path) that covers all variable vertices of planar instance without crossing any edge (solid lines). We observe that additional edges can be added to the spinal path to obtain a tree (spinal tree) which spans clause variables as leaves (dashed lines).

Theorem 2.4. *Given a chain of disks, the solution to the* MAX-MSTN *problem on the chain* D^c *and the points* T_L *and* T_R *is the set of points* $\{T_L, p_1, \ldots, p_k, T_R\}$*, where* p_i *is the selected point for disk* D_i^c*, and these points form one of the two possible zig-zag paths that traverse the extreme top and bottom points of the disks (Figure 3).*

We define Z_D as this proposed zig-zag path that alternates between the extreme upper and lower points of a set of disks D, where the centers of all disks in D are collinear. Note that the MST for the chain of disks forms a path starting from T_L and ending at T_R.

To prove Theorem 2.4, we need a few preliminary results.

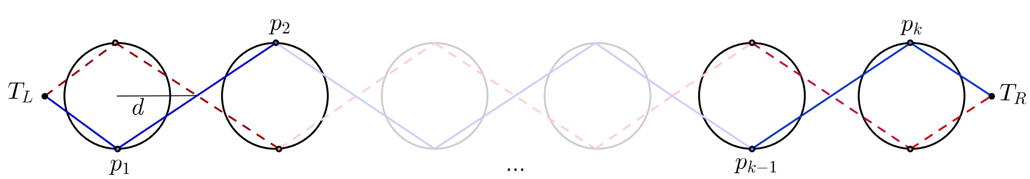


Figure 3: The two possibilities for the MAX-MSTN solution for chain of disks. Here we have $d = 1.5$.

Lemma 2.5. *Given a disk, if the set of edges containing a point in the disk is fixed for any position of the point, any* MAX-MSTN *solution does not contain a point inside the disk (i.e., the selected point is on the circumference of the disk).*

Proof. Suppose we are given a point p in a disk D, and a set of points $Q, |Q| \geq 1$, where there exists an edge between p and each point in Q. Given any line ℓ where $p \in \ell$, let p' be a point on $\ell \cap D$. Let w be the sum of the weights of edges between p' and each point in Q. The weight of each edge (as p' runs along ℓ) is a convex function, and therefore so is the sum w [11]. Hence, the sum of the weights of the edges is maximized at one of the two intersection points of ℓ with D. □

The following lemma states that a MAX-MSTN solution should follow the path of a ray of light which is started at point T_L and is reflected on the intersection with each disk.

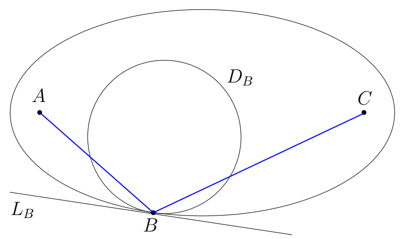
Lemma 2.6. *Let p_1 be the selected point of a* MAX-MSTN *solution on the leftmost disk in a chain. Consider the segment Lp_1 as a ray of light that is reflected on the interior face of each disk. Any valid* MAX-MSTN *solution follows the path traversed by the ray, i.e., the two neighboring segments of the MST are the reflection of each other on the tangent line of the intersection point.*

Proof. Let B be the selected point of any disk in a MAX-MSTN solution and A and C be the points of the MST connected to B on the left and right in the tree, respectively (for the leftmost disk we have $B = p_1$ and $A = T_L$). Let D_B be the disk that B belongs to, and by Lemma 2.5, B is on the circumference of D_B. Let L_B be the line tangent to D_B at point B. For this proof we refer to the diagram in Figure 4(a). By definition, the ellipse with foci A and C consists of the points P that are equidistant, on the aggregate, to A and C, i.e., $|AP| + |PC|$ is a constant. Points inside the ellipse are closer, on the aggregate, to A and C while points outside the ellipse are farther.

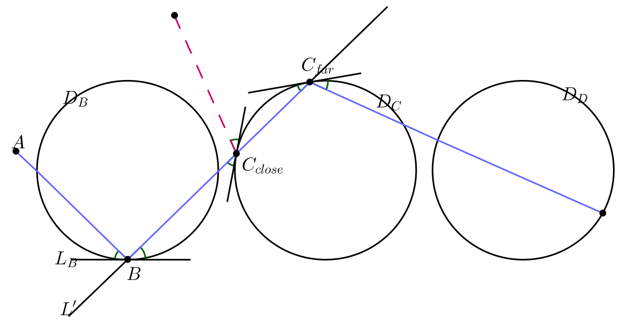
Now if the circle D_B and an ellipse incident upon B are not tangent at B, it follows that there are points in D_B outside the ellipse and hence the length of the MST grows if such a point is chosen instead of B, which contradicts the assumption that the MST through B is maximal. Since A and C are the foci, the projective property of the ellipse implies that BC is the reflection of AB on the tangent L_B to the ellipse.

Let C_{close},C_{far} be the intersection points of a line L' incident upon B and disk D_C, so that C_{close} is closer to B than C_{far}. If we apply the same argument by replacing A with B and B with C_{close}, we see that the reflected path is located to the left of the line BC_{far}, which is not a feasible path (e.g., the red line in Figure 4(b)). This implies that the MAX-MSTN solution cannot contain C_{close}, hence, C can only be located on C_{far}. The proof is complete if we apply this argument by setting $A = T_L$ and $B = p_1$ to set $C = C_{\text{far}}$ and inductively apply the same reasoning by replacing A with B and B with C. □

The above lemma implies that selecting the first point p_1 on the leftmost disk D_1 defines all other points that should be selected by the MAX-MSTN solution algorithm. In particular if p_1 is the extreme point of D_1 at top or bottom, the points selected for other disks will be on the extreme points of other disks as well to form the Z_D configuration described in Theorem 2.4. Note that if the reflected line on any disk does not intersect the next disk (the one to the right), or at the rightmost disk the reflected line is not incident upon the terminal point T_R, then in these cases the initial selection for p_1 does not produce a MAX-MSTN solution (Figure 5).



(a) The ellipse with foci A and C and incident upon B should be tangent to D_B if AB and BC are edges in a MAX-MSTN solution.



(b) If the MAX-MSTN solution selects point B, it has to select point C_{far}.

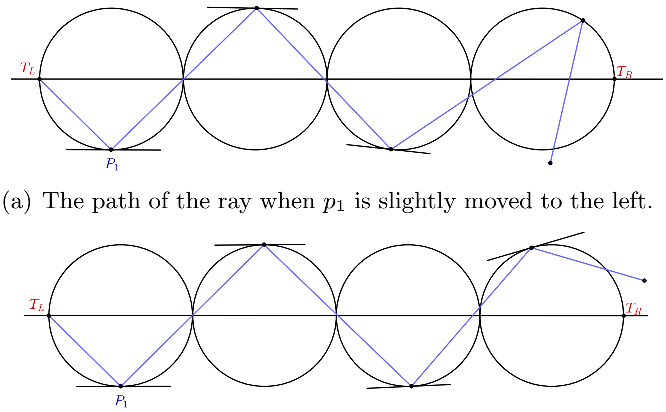
Figure 4: The reflection effect.

Lemma 2.7. *Let $p_1, p_2, \ldots, p_i$ be the selected points of a* MAX-MSTN *solution on the consecutive disks (from left to right). Then for any i we have the following:*
-If p_i is on the bottom-half of disk D_i, then p_{i+1} is on the top-half of disk D_{i+1} and vice versa.
-If p_i is on the right-half of disk D_i, then p_{i+1} is on the left-half of disk D_{i+1} and vice versa.

Proof. Let p_i and p_{i+1} be both on the bottom-half of two consecutive disks; a larger MST can be found by replacing all points p_j ($j \geq i+1$) with their reflection over L_{hz}, the horizontal line passing through the disk centers. Such transform increases the weight of the edge (p_i, p_{i+1}) and preserves the weight of other edges of the MST (the same holds if two selected points are on the top half of consecutive disks). Using mathematical induction, the second property is direct from Lemma 2.6. More precisely, if p_{i-1} and p_i are respectively on the right and left halves of their disks, then Lemma 2.6 ensures that p_{i+1} is on the right half of D_{i+1} (Figure 6(a)). Similarly, if p_{i-1} and p_i are respectively on the left and right halves of their disks, then p_{i+1} is on the left half of D_{i+1} (Figure 6(b)). □

To prove Theorem 2.4, we show that if we select p_1 as any point except the extreme points on bottom or top of the leftmost disk D_1, the reflection path described in Lemma 2.6 does not pass the terminal point T_R. For each disk $D_i \in D$, define the *canonical line* as the line passing through the center of D_i and p_i, and the *canonical angle* α_i as the smaller angle between L_{hz} and the canonical line of the disk. Note that Lemma 2.7 implies that either all canonical lines have positive slope or all have negative slope. To make the explanation simpler, consider another disk D_0 with center at distance $2d$ from that of the leftmost disk, and let p_0 be the intersection of D_0 with the line passing T_L and p_1 (Figure 7). Since T_L is located at the mid point of the centers of p_0 and p_1, observe that the canonical angle of D_0 and D_1 are equal regardless of the choice of p_1, i.e., $\alpha_0 = \alpha_1$.

Lemma 2.8. *If the selected point is any other point than the extreme top or bottom points of disk D_1, the sizes of the canonical angles form a strictly decreasing sequence, i.e., $\pi/2 > \alpha_0 = \alpha_1 > \alpha_2 > \ldots > \alpha_n$.*



(a) The path of the ray when p_1 is slightly moved to the left.

(b) The path of the ray when p_1 is slightly moved to the right.

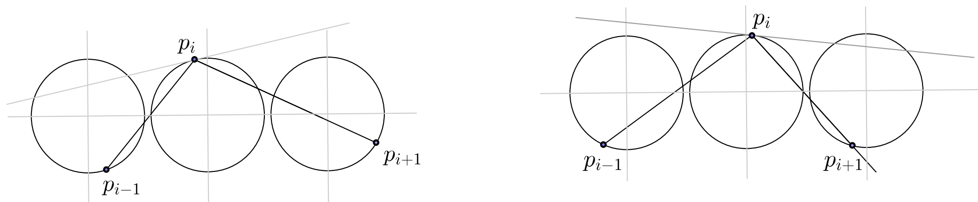
Figure 5: If the selected point p_1 on the leftmost disk is not the extreme top or bottom point, the path of the ray does not pass T_R.

Proof. We use mathematical induction. As observed before, we have $\alpha_0 = \alpha_1$. In the base case, we show $\alpha_1 > \alpha_2$. Consider otherwise, i.e., $\alpha_1 \le \alpha_2$. Note that the slopes of all canonical lines are either positive or negative. W.l.o.g. assume all slopes are negative (see Figure 7). Let X be the intersection point of the line passing through p_1 and p_2 with the horizontal line L_{hz}. Also let x be the distance between X and the center of D_1. Since $\alpha_1 \le \alpha_2$, we have $x \le d$. Let O be center of D_1 and θ be the angle between lines $\overline{p_0p_1}$ and $\overline{Op_1}$, which is the same as the angle between $\overline{p_1p_2}$ and $\overline{Op_1}$ (by the reflection law). Applying the Law of Sines for triangle $\triangle Op_1X$ we get $x = \sin\theta/\sin(\alpha_1 - \theta)$. Let Y be the intersection point of $\overline{p_0p_1}$ and horizontal line L_{hz} (which is T_L in the base case); and let y be the distance between Y and O. Since $\alpha_0 \ge \alpha_1$, we have $y \ge d$ (equivalence occurs in the base case). Applying the Law of Sines on triangle $\triangle Op_1Y$, we get $y = \sin\theta/\sin(\pi - \alpha_1 - \theta)$.

Since $x \le d$ and $y \ge d$, we get $\sin\theta/\sin(\alpha_1 - \theta) \le \sin\theta/\sin(\pi - \alpha_1 - \theta)$, which implies that $\alpha_1 \ge \pi/2$; this cannot happen as we assumed α_0 is strictly smaller than $\pi/2$. So we conclude $\alpha_1 > \alpha_2$. To complete the proof, we can apply the same argument by replacing p_0, p_1, and p_2 with p_{i-1}, p_i and p_{i+1} to show $\alpha_i > \alpha_{i+1}$ for $i > 2$. □

Now we are ready to prove Theorem 2.4.

Proof. Lemma 2.6 implies that the MAX-MSTN solution should follow the path of a ray of light that is started at T_L and is reflected at a point p_i of each disk, and finally passes through T_R. Note that the two zig-zag paths stated in the theorem have such a property. We show no other ray can do so. Assume the selected point of the leftmost disk is not on the extreme top or bottom (if it is, the reflected path would be one of the paths described in the theorem). By Lemma 2.8, the canonical angles of the disks are strictly decreasing. This implies that for any disk D_i, the reflected path does not pass the point in the middle of the segment between the centers of D_i and D_{i+1} (in other case, the canonical angles of the two disks would be equal). In particular, if one



(a) If p_{i-1} and p_i are respectively on the right and left half-disks, by the reflection property p_{i+1} will be on the right half-disk.

(b) If p_{i-1} and p_i are respectively on the left and right half-disks, by the reflection property p_{i+1} will be on the left half-disk.

Figure 6: Illustration of Lemma 2.7.

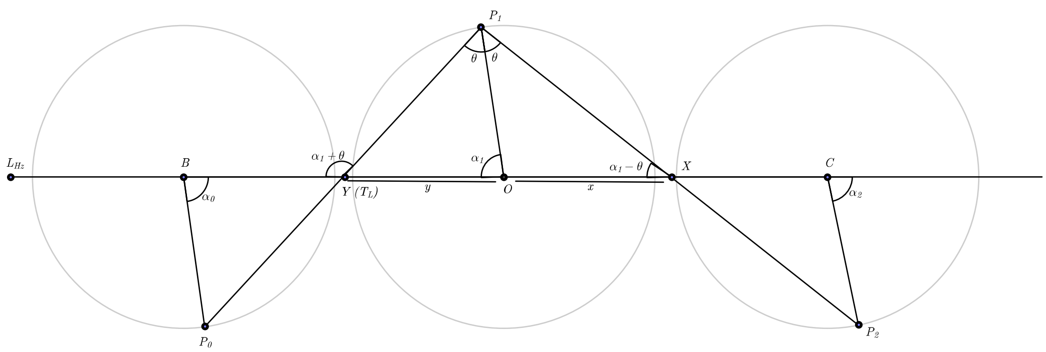


Figure 7: If $\alpha_0 < \pi/2$, the canonical angles are strictly decreasing.

assumes an extra disk exists on the right, the reflection path does not pass T_R. This implies that selecting any point other than an extreme point on the top or bottom of the leftmost disk results in a non-optimal MAX-MSTN solution for the chain of disks. □

Now that we have established Theorem 2.4, we revisit the reduction. Recall that we are reducing to the planar 3-SAT problem, in which an instance of 3-SAT is represented as the planar graph $G = (V, E)$, and that we have a spinal tree $T = (V, E_T)$, where T is connected and $G \cup T = (V, E \cup E_T)$ remains planar while $E \cap E_T = \emptyset$ (Figure 2). Furthermore, we make use of *wires*, where a wire is a set of disks of radius 0 (points) placed in close succession, so that we may interpret them as fixed lines in the MST solution.

For some intuition on the structure of the reduction, consider the graph $G \cup T$. In this graph, we replace all variable and clause vertices with variable and clause gadgets, respectively. Also, we replace the edges in $G \cup T$ with fixed wires so that they will be part of any MST, and these fixed wires are disjoint from the gadgets. The fixed wires include the edges of G (which we call *e-wires*) and also the edges of the spinal tree. The clause and variable gadgets include some disks, and the goal of the MAX-MSTN algorithm is to select points in each disk so that the resulting MST has maximum weight. Note that any MST is composed of the disconnected fixed wires that are each connected to some gadgets. We design the gadgets so that the e-wires (i.e., the edges of G) attach exclusively to clause gadgets. The combinatorial structure of the resulting MST includes the spinal tree as a sub-tree (that is why we call it 'spinal'); hence, an optimal MAX-MSTN algorithm selects the points in gadgets in a way to impose the maximum weight for the edges that connect the e-wires to the spinal tree.

2.3.1 Variable Gadgets

For each variable x_i, we build a set of $3c+2$ disks D^i in the configuration described in Theorem 2.4, spaced by $d = 21/8 = 2.625$, where c is the number of clauses containing instances of the variable. The reasons for the particular choices of distances are explained in the Reduction section below. The terminal points T_L^i and T_R^i of the gadget are joined to the spinal tree of the construction. Specifically, a wire joins T_R^i to $T_L^{i+1}, i \in \{1, \ldots, n-1\}$ for each of the n variable gadgets. Assume for ease of discussion that the centers of D^i are incident upon a horizontal line L_{hz}, so that the terms above, below, left, and right are well defined.

Wires from the clause gadgets may approach the variable gadget from above, below, or both. As mentioned earlier, we call these the *e-wires*, because each such wire corresponds to an edge of E in the input planar 3-SAT graph. Each e-wire terminates at a point that is distance $1+\sqrt{29}$ from a disk center, along a line incident upon the disk center and perpendicular to L_{hz}, i.e., the terminal point of the wire and the disk center share the same x-coordinate (see Figure 8). We provide more details regarding e-wires shortly. Finally, suppose w.l.o.g. that $D_j^i \in D^i$ has the terminal points of an e-wire above it. All other e-wires are restricted so that no other e-wires may approach D_j^i, and furthermore, no e-wires may approach disks D_{j-1}^i or D_{j+1}^i from above, so that there is at least distance $4d = 10.5$ between adjacent e-wires. In other words, e-wires that approach the variable gadget from the same side have at least one disk between them.

Lemma 2.9. *Suppose we are given a variable gadget where points are placed in the disks as described in Theorem 2.4. Then a point in a disk is either distance $2+\sqrt{29}$ or $\sqrt{29}$ from the nearest point in an e-wire, and we may arrange it so that these distances correspond to agreement or disagreement respectively between the truth value of the variable gadget and that of the instance of the variable represented by the e-wire.*

Proof. Given the two possible Z_D configurations shown in Theorem 2.4, we arbitrarily select one of such configurations as the `true` setting, and the opposite configuration as `false`. This way, we can place the terminal points of the e-wires near the disks of the variable gadget so that if the truth value used for the variable gadget matches that of the e-wire, then the minimum distance between the e-wire and the nearest point in the variable gadget is $2+\sqrt{29}$. However, a mismatched truth value would mean that a point lies only distance $\sqrt{29}$ from the e-wire when the points are in the Z_D configuration. □

2.3.2 Clause Gadgets

For each clause in the 3-SAT instance, we build a clause gadget by assembling wires and disks as shown in Figure 9. Given some point c which is the center of the gadget, consider three rays

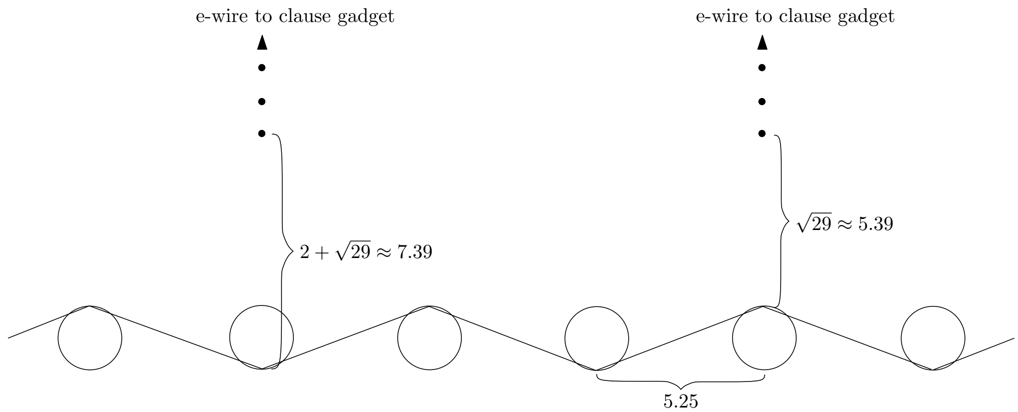


Figure 8: The configuration used for the variable gadgets. Here, two clauses include the variable in the opposite truth values. Filled small circles represent disks of radius 0, while larger empty circles represent unit disks. E-wires are placed so that the nearest point on any disk is at distance $\sqrt{29}$, and the terminal point of the e-wire is on a line that is incident upon and perpendicular to the disk center. This way, if the Z_D configuration for the variable gadget matches the truth value of the e-wire, then the nearest point to the e-wire in the gadget is $2 + \sqrt{29}$ units distant, while a mismatch in truth values means the nearest point in the gadget would only be distance $\sqrt{29}$ away. The disk centers in the gadget are placed at a distance 5.25 apart so that the nearest point to an e-wire can only be from the nearest disk, regardless of the path through the disks.

$\overrightarrow{r_1}, \overrightarrow{r_2}, \overrightarrow{r_3}$ emanating from c that are equally spaced by 120°. For the first 50 units of each ray, points are placed every unit distance to form wires extending from c. Next, unit disks are placed with centers distanced $50 + \sqrt{15}, 50 + 3\sqrt{15}$ and $50 + 5\sqrt{15}$ from c. A final point is placed along each ray at a distance $50 + 6\sqrt{15}$ from c. From the spinal tree, a wire joins to one of the terminal disks along a ray.

Each e-wire terminates with four branches near the clause gadget and a single terminal near the variable gadget which corresponds to the literal. As mentioned in the variable gadgets section, the terminal point p_t of the e-wire near the variable gadget lies at distance $\sqrt{29} \approx 7.39$ above a disk D in the gadget. The other terminals of the e-wire are placed so that a pair of points is near the middle disk on two of the rays, as shown in Figure 9. To be specific, suppose that the construction is positioned so that the center point of the middle disk of one ray is placed at the origin of the plane, and the ray is aligned with the x-axis. Call this the *canonical position* of the ray. Terminal points from one e-wire are placed at the coordinates $(-2, -6)$ and $(2, -6)$, while terminal points from another e-wire for the clause are placed at positions reflected through the x-axis, i.e., $(-2, 6)$ and $(2, 6)$. This way, the terminal points of each pair of e-wires for a clause are positioned symmetrically about a disk in the clause gadget.

Supposing points were selected in these three disks in a zig-zag Z_D configuration (shown to be an optimal configuration in Theorem 2.4), then the edges between points in adjacent disks have weight 8, and those from the first and last disk to the nearest points along the ray have weight 4. The distance from the point in the middle disk to the nearest point on one e-wire is $\sqrt{29} \approx 5.39$,

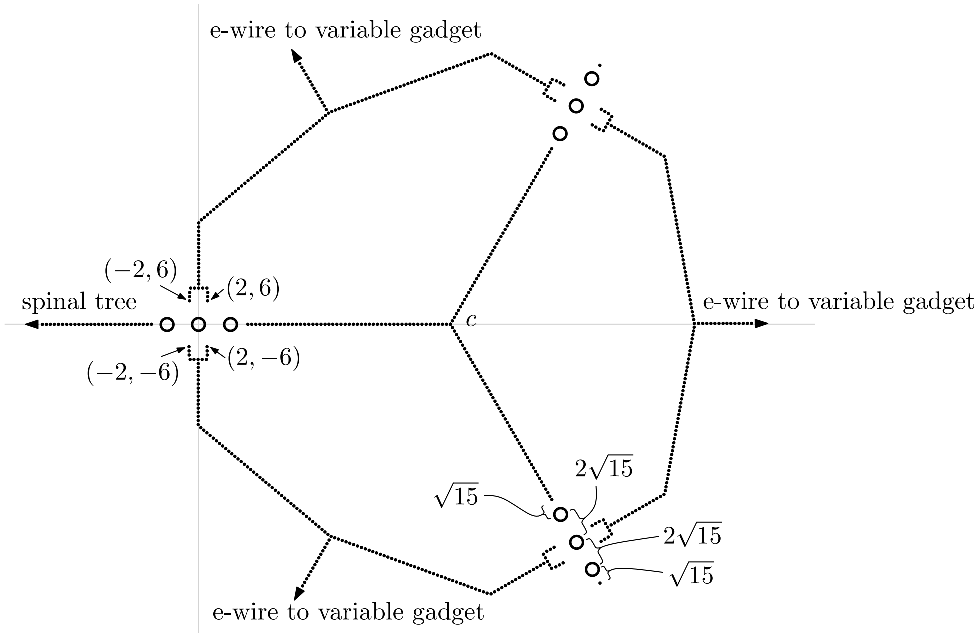


Figure 9: The configuration used for the clause gadgets, with the gadget placed in the canonical position. The filled small circles (black dots) are disks of radius 0, while the larger empty circles are unit disks (the disks are drawn with larger than unit radius for legibility). A wire extends from the gadget as the spinal tree, and three e-wires go to three variable gadgets (one for each literal in the corresponding clause of the 3-SAT instance).

and to the nearest point on the opposite e-wire is $\sqrt{53} \approx 7.28$.

Lemma 2.10. *Suppose that a ray of the clause gadget is placed in the canonical position. Then the weight of the* MAX-MSTN *solution is optimized when the point for the middle disk is placed at* $(0, 1)$ *or* $(0, -1)$.

Proof. Observe that the minimum distance from a point on an e-wire to the first or the third disk along a ray of the clause gadget is $\sqrt{\left(2\sqrt{15}-2\right)^2+36}-1 \approx 7.31$, while the maximum distance from such a point to a point in the middle disk is $\sqrt{53} \approx 7.28$. Therefore, if edges of the MST exist between an e-wire and points in the disks of the clause gadget, such edges join to the point in the middle disk.

Suppose that no edges exist between the e-wires and the point in the disk; the optimal path through the disks is the Z_D configuration, and the lemma holds. Next, suppose only one of the e-wires is joined with an edge; the weight of this edge is maximized at the farther of the two candidate positions for the point in the middle disk, and so the lemma holds again. Finally suppose that both e-wires have edges joining to the point in the middle disk. We know that the point that maximizes the sum of the weights w_s of all edges incident upon the point is found on the edge of the disk by Lemma 2.5. Suppose w.l.o.g. that the point p is chosen in the middle disk so that the x-coordinate is ≥ 0, and so we may assume that the edges join to the right terminals of each e-wire (positioned at $(2, 6)$ and $(2, -6)$). The distance from these two points to a point on the right half of a unit circle centered at the origin is maximized at either $(0, 1)$ or $(0, -1)$, and so the lemma follows. □

Lemma 2.11. *If all three e-wires associated with a clause gadget are joined to the clause gadgets with edges, then the weight of these edges is maximized when two e-wires are joined with edges of weight $\sqrt{29}$, and the other is joined with an edge of weight $\sqrt{53}$.*

Proof. By Lemma 2.10, we know that all clause gadgets will use the Z_D configuration through their disks. Therefore, the weight of an edge between the clause gadget and an e-wire is either $\sqrt{29}$ or $\sqrt{53}$. Further, because two e-wires approach each set of disks, if one e-wire is distance $\sqrt{29}$ from a point in the clause, then another e-wire is $\sqrt{53}$ from the same point. The e-wires are arranged so that each pair of e-wires associated with a clause gadget has this relationship. Therefore, there are two possible settings:

1. Each e-wire is distance $\sqrt{29}$ from the nearest point in the clause gadget.
2. One e-wire is distance $\sqrt{53}$ from the nearest point in the clause gadget, while the other two e-wires are distance $\sqrt{29}$ from the nearest point in the clause gadget. The configuration of the path through the triple of disks between the latter two e-wires is inconsequential.

Since we are computing a minimum spanning tree with maximum weight, the second configuration is preferable. □

2.3.3 Reduction

The key to the reduction is that if and only if there exists a satisfying assignment to a given planar 3-SAT instance, then an optimal MAX-MSTN solution to the construction outlined above will join all of the e-wires to the clause gadgets, leaving the variable gadgets unaffected. Therefore, we begin the reduction by determining the weight of an optimal solution. If there is no satisfying assignment, the optimal MAX-MSTN algorithm selects the points in a way that the Z_D path through at least one variable gadget is affected, and the total weight of the optimal MAX-MSTN solution is reduced. We determine a lower bound on this effect, and in so doing, establish the hardness of the problem.

Let us consider the structure of an optimal solution to MAX-MSTN if there exists a satisfying assignment, and in particular we examine the weights of the edges required to join each of the three e-wires associated with a clause gadget (the same reasoning applies to all clause gadgets). Recall that an e-wire may be distance $\sqrt{29} \approx 5.39$ or $\sqrt{53} \approx 7.28$ from the nearest point in the clause gadget, by Lemma 2.10. Assuming that the points in the variable gadget are in the Z_D configuration, the nearest point to an e-wire in the variable gadget may be $\sqrt{29} \approx 5.39$ or $\sqrt{29} + 2 \approx 7.39$, by Lemma 2.9. Since at least one of the literals may be satisfied in the clause, the corresponding e-wire, call it e_{sat}, is distance 7.39 from its variable gadget when the variable gadget has the Z_D configuration associated with the satisfying truth assignment. To maximize the weight of the MAX-MSTN solution, the paths through the disks in the clause gadget should be set so that e_{sat} is $\sqrt{53}$ from the nearest point in each of the two disks that it approaches. Thus, the weight of the edge to connect to e_{sat} is $\sqrt{53}$. The other two e-wires, whether they correspond to literals that may be satisfied or not, are each connected with an edge of weight $\sqrt{29}$, by Lemma 2.11. Since no point in any variable gadget is closer to an e-wire than the points in the clause gadget, we have shown that the e-wires are all joined to the clause gadget in an optimal MAX-MSTN solution. Therefore, all variable gadgets are connected to the rest of the MST only at their endpoints and, by Theorem 2.4, selecting a setting other than the Z_D configuration in any variable gadget is sub-optimal.

We now compute the optimal weight of a MAX-MSTN solution under the assumption that there exists a satisfying assignment to the planar 3-SAT instance. Let w_{st} be the total weight of the MST over the wires in the spinal tree segments of the construction, and let w_{ew} be that for the e-wires, which are both fixed for any MAX-MSTN solution. The MAX-MSTN solution over the clause gadgets has weight $3(50+24)+2\sqrt{29}+\sqrt{53} \approx 240.05$, i.e., 50 for each set of points along each of the three rays before the disks, 24 is the weight of the optimal Z_D path through the disks, and two e-wires are connected with edges of weight $\sqrt{29}$ while the remaining one is connected with one of weight $\sqrt{53}$. Therefore, given that there are m clauses, the total weight for all clause gadgets is $w_{\text{cg}} = m(3(50+24)+2\sqrt{29}+\sqrt{53})$. Finally, assume there is a total of h disks in all of the variable gadgets of the construction. The total weight of the optimal Z_D configuration in these disks is $w_{\text{vg}} = h\sqrt{5.25^2+2^2} \approx h \cdot 5.62$. Therefore, the total weight of the optimal MAX-MSTN solution is $w_{\text{tot}} = w_{\text{st}} + w_{\text{ew}} + w_{\text{cg}} + w_{\text{vg}}$, all of which we can compute *a priori* once the MAX-MSTN instance is constructed.

Now consider the case that there is no satisfying assignment for the 3-SAT instance. Consider a truth assignment (defined by the alignment of zig-zag paths in variable gadgets) and a clause that is not satisfied by that assignment (such a clause exists since the 3-SAT instance is not satisfiable). Note that a MAX-MSTN algorithm might deviate from selecting zig-zag paths; this will be addressed shortly. For the clause that is not satisfied, we may dismiss the setting where each e-wire is $\sqrt{29}$ from a point in the clause gadget as sub-optimal, by Lemma 2.11. Therefore, one of the e-wires, call it e_{nsat}, is $\sqrt{53}$ from the nearest point in the clause gadget, and this affects the positions of the points in the corresponding variable gadget at the other end of the e-wire[2]. Let p_{vg} be the nearest point in the variable gadget to e_{nsat}. Notice that the Fermat point of the two points neighboring p_{vg} in the variable gadget and the nearest point in e_{nsat} lies above the disk containing p_{vg}, so moving p_{vg} down within the disk increases the weight of the total MST if there is an edge between p_{vg} and the nearest point in e_{nsat}. Moving the point at least $\sqrt{53}$ away from e_{nsat} increases the weight of the MAX-MSTN solution as much as possible. Now the configuration of the clause gadget is the same as in the satisfying assignment above, so we need only measure the reduction in weight resulting from the changes to the points in the variable gadget to see the total reduction in weight relative to an optimal MAX-MSTN solution for a satisfying assignment.

To determine a lower bound on this effect, assume that there is only one transition in the zig-zag pattern. That is, assume w.l.o.g. that at some point that the Z_D configuration is in the `true` setting, and then the pattern switches to the `false` setting for the remainder of the gadget. All the wires in the construction must maintain a minimum separation of at least $\sqrt{53}$ to preserve the desired structure of the MST, and so e-wires that approach the variable gadget from the same side must have at least two disks between them (since they correspond to mismatched truth values), and those approaching from opposite sides may have only one disk between them. To find an appropriate bound, first we assume that the closest points to each e-wire, call these p_ℓ and p_r,

[2] If the Z_D configuration were maintained, then the e-wire could be joined sub-optimally to the variable gadget with weight $\sqrt{29}$, since the assignment is not satisfying. However, a larger MST can be achieved by deviating from the Z_D configuration in the variable gadget.

remain in the positions that would be optimal for a Z_D configuration, i.e., at the furthest point possible from the axis of the variable gadget, and then we compute the maximum weight path between p_ℓ and p_r in the gadget. Next, we bound the possible error introduced by placing p_ℓ and p_r in the extreme positions.

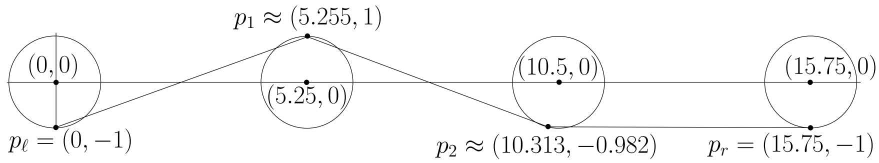


Figure 10: The optimal setting for Case 1. The optimal point in the first separating disk is slightly to the right of the extreme upper point in the disk, while that of the second disk is nearly 11° clockwise of the extreme lower point. Despite this difference, the difference in path weight between this optimal path and a path using the extreme points is less than 0.01 units.

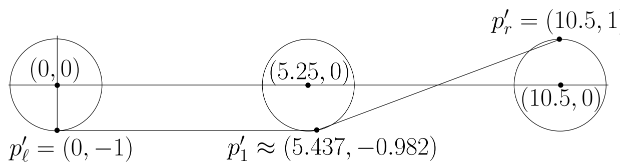


Figure 11: The optimal setting for Case 2. The abberation of the point in the separating disk from the extreme lower point is similar to that of the second separating disk in Case 1, and the difference in path weight between the optimal path and that using only the extreme points is less than 0.01 units.

Case 1: Two separating disks (Figure 10). W.l.o.g., we position points at coordinates $p_\ell = (0, -1)$ and $p_r = (15.75, -1)$, and seek the maximum weight path using points p_1 and p_2, where p_1 is in a unit disk centered at $(5.25, 0)$, and p_2 is in one centered at $(10.5, 0)$. If we define the points as $p_1 = (5.25 + \sin(\theta_1), \cos(\theta_1))$ and $p_2 = (10.5 + \sin(\theta_2), \cos(\theta_2))$, then the total weight w' is defined by:

$$\begin{aligned} f(\theta_1, \theta_2) = &\sqrt{(-5.25 - \sin(\theta_1))^2 + (-1 - \cos(\theta_1))^2} \\ &+ \sqrt{(5.25 + \sin(\theta_2) - \sin(\theta_1))^2 + (\cos(\theta_2) - \cos(\theta_1))^2} \\ &+ \sqrt{(5.25 - \sin(\theta_2))^2 + (-1 - \cos(\theta_2))^2}. \end{aligned}$$

Using Maple, we observe that the optimal path uses the values $\theta_1 \approx 0.005$ and $\theta_2 \approx 3.33$ radians, for a total weight of more than 16.49 over the three edges. By moving p_1 to $(5.25, 1)$ and p_2 to $(10.5, -1)$, we reduce the total weight of the edges by less than 0.01, so that the weight remains greater than 16.48.

Case 2: One separating disk (Figure 11). Now points are placed at $p'_\ell = (0, -1)$ and $p'_r = (10.5, 1)$, and we seek the path using a point p'_1 again found in a unit disk centered at $(5.25, 0)$.

Define $p'_1 = (5.25 + \sin(\theta), \cos(\theta))$, and the total weight w' is now defined by:

$$f(\theta) = \sqrt{(-5.25 - \sin(\theta))^2 + (-1 - \cos(\theta))^2} + \sqrt{(5.25 - \sin(\theta))^2 + (1 - \cos(\theta))^2}.$$

Using Maple, we observe that such a path has maximum weight greater than 10.87, realized when $\theta \approx 2.95$ radians. To create a simplified configuration, we shift this point so that $p'_1 = (5.25, -1)$, and we observe that this change reduces the total weight of the edges by less than 0.01, so that the weight is greater than 10.86.

The final remaining possible suboptimality is the assumed positions for p_r in Case 1 and p'_ℓ in Case 2 (p_ℓ and p'_r are in the midst of points in the Z_D configuration). Consider the adjusted path from $p'_1 = (5.25, -1)$ to p'_ℓ to a point in the disk left of D'_ℓ, i.e., the disk centered at $(-5.25, 0)$. This setting is again analogous to Case 2, and so we can conclude that this assumed position reduces the total weight by less than another 0.01 units. We know by Theorem 2.4 that any changes in position from the extreme points in the remainder of the gadget only decreases the total weight of the MST.

Therefore, the overall introduced errors of our assumptions are less than 0.02 for each case. For Case 1, we conclude that the total weight lost as a result of an unsatisfiable clause is greater than $3\sqrt{31.5625} - 16.49 - 0.02 \approx 0.34$. In Case 2, the effect is greater than $2\sqrt{31.5625} - 10.87 - 0.02 \approx 0.34$. Hence, given a planar 3-SAT instance, we know that the total weight of the optimal solution to the MAX-MSTN construction is less than $w_{\text{tot}} - 0.34$ if and only if there is no satisfying assignment for the instance. Therefore, if we choose a value of ε so that $\varepsilon < 0.34/w_{\text{tot}}$, then a $(1-\varepsilon)$-approximate solution to the MAX-MSTN problem may be used to determine whether there is a satisfying assignment for the planar 3-SAT instance. Since the latter problem is NP-hard, we conclude that MAX-MSTN does not admit an FPTAS unless P=NP.

3 MSTN

In this section we present a parameterized algorithm for the MSTN problem, followed by the proof of hardness of approximation.

3.1 Parameterized $1 + 2/k$-Approximation Algorithm

Recall that to have *k-separability* means that the minimum distance between any two disks is at least $kr_{\max}$, and the separability of an input instance I is defined as the maximum k such that I satisfies k-separability.

Theorem 3.1. *For* MSTN *when the regions of uncertainty are disjoint disks with separability parameter $k > 0$, the algorithm that builds an MST on the centers of the disks achieves a constant approximation ratio of $\frac{k+2}{k} = 1 + 2/k$.*

Proof. Assume that we have a set D of n disks that satisfies k-separability. Let T_c be the MST on the centers and T_{opt} be an optimal MST, i.e., an MST that contains one point from each disk and its weight is the minimum possible. Define *Temp* as the spanning tree (not necessarily an MST) with the same topology as T_{opt} but on the points of T_c, i.e., on the centers. Since T_c is an MST on centers, we have $w(T_c) \leq w(\mathit{Temp})$. Consider an arbitrary edge e in *Temp* and let D_i and D_j be the two disks that are connected by e. Let r_i and r_j be the radii of D_i and D_j, respectively, and let d be the distance between D_i and D_j. In T_{opt} the disks D_i and D_j are connected by an edge e' whose weight is at least d. The weight of e on the other hand is $d + r_i + r_j$. Therefore the ratio between the weight of an edge in T_{opt} and its corresponding edge in *Temp* is at least

$$\frac{d}{d + r_i + r_j} \geq \frac{kr_{\max}}{kr_{\max} + r_i + r_j} \geq \frac{kr_{\max}}{kr_{\max} + r_{\max} + r_{\max}} = \frac{k}{k+2}.$$

Since this holds for any edge of *Temp*, we get $w(T_c) \leq w(\mathit{Temp}) \leq \frac{k+2}{k} w(T_{\text{opt}})$. Therefore we get an approximation factor of $\frac{k+2}{k} = 1 + 2/k$ for the algorithm. □

As with the parameterized algorithm for MAX-MSTN, as the disks become further apart (as k grows), the approximation factor approaches 1.

3.2 Hardness of Approximation

To prove the hardness of the MSTN problem, we present a reduction from the planar 3-SAT problem. Recall that planar 3-SAT is a variant of 3-SAT in which the graph $G = (V, E)$ associated with the formula is planar.

Theorem 3.2. MSTN *does not admit an FPTAS unless* P=NP*.*

In the hardness proof of MAX-MSTN, we used a spinal tree in the reduction. In this section, we use the *spinal path* as a path $P = (V_v, E_P)$ with a set of edges E_P such that $E \cap E_P = \emptyset$, where P passes through all variable vertices in G without crossing any edge in E. As mentioned earlier, the restricted version of planar 3-SAT remains NP-hard [8]. To reduce planar 3-SAT to MSTN, we begin by finding a planar embedding of the graph associated with the SAT formula. We force the inclusion of the spinal path as a part of the MST using wires. We define a wire as a set of disks of radius 0 placed in close succession, so that we may interpret a wire as a fixed line in the MSTN solution. We replace each variable vertex of V by a *variable gadget* in our construction. These gadgets are composed of a set of disks and some wires, and are defined in such a way that we may choose the points so that the size of the MST is equal to a certain value, if and only if the SAT formula is satisfiable.

3.2.1 Variable Gadgets

A variable gadget is formed by a k-*flower*, where $k = 4c + 6$ and c is the number of clauses in the planar 3-SAT instance that include the variable (each clause requires 4 disks, and each of the edges of the spinal path requires 3 disks). As illustrated in Figure 12, a k-flower is composed of k disks of unit radius, centered on the vertices of a regular k-gon. Also, each disk is tangent to its two neighboring disks, and each pair of consecutive disks D_i, D_{i+1} intersects at a single point $q_{i,i+1} = D_i \cap D_{i+1}$, which we call a *tangent point*[3]. Moreover, there is a k-*star* in the middle of the

[3]Using this construction, pairs of disks of the k-flower trivially intersect at a single point, which simplifies our analysis. To achieve strict disjointedness, the disks of the k-flower may be contracted to have radius $1 - \gamma$ so that the tangent point is now distance γ from the nearest point in the adjacent disks. Any path which uses the tangent point in our analysis will have less than 2γ units of additional weight on these shrunken disks, and there are fewer than $n(4m + 6)$ disks, where n and m are the number of variables and clauses respectively. Choosing an appropriate value of γ so that $2\gamma n(4m + 6) \ll 0.845$ achieves the same result as our simplified analysis.

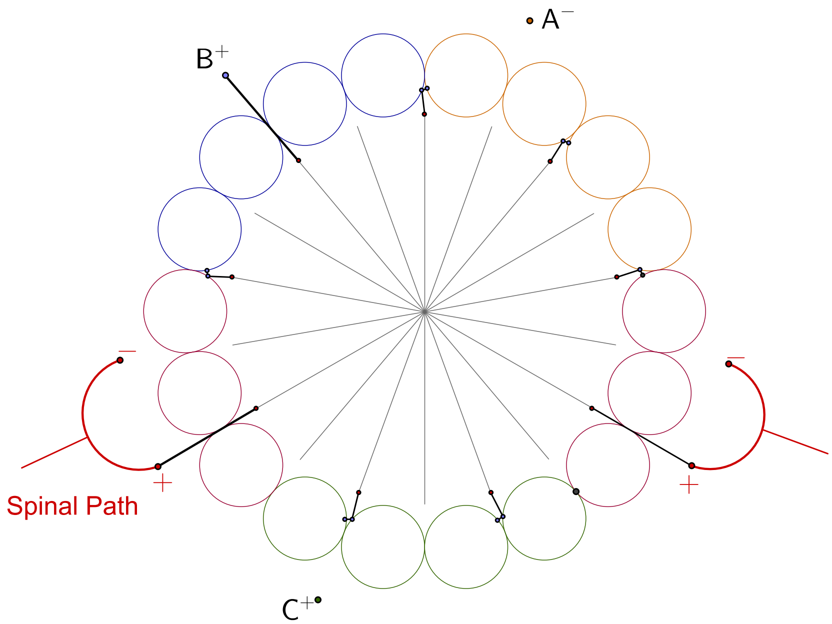


Figure 12: A variable gadget with eighteen disks (containing an 18-flower and an 18-star) for a variable x. Here B^+ and C^+ are the endpoints of the wires that connect to clauses that include x in the positive form, while A^- represents a clause that includes x in negative form. The picture illustrates the case in which the algorithm takes the positive choice for x, and clause B is satisfied with x. Clause C is satisfied via some other variable, as is clause A, assuming that it is satisfied. Note that every other path on the k-star connects to a pair of disks on the k-flower.

gadget composed of k fixed wires, where the i^{th} wire connects a point unit distance from the tangent point $q_{i,i+1}$ to the center point of the k-star. The spinal path is placed so that it approaches the variable gadget twice, and each of these approaches requires three disks. We split the wires of the spinal path once near the variable gadget as shown in Figure 12, and wires terminate at a distance ≈ 1.755 from the nearest tangent point, for reasons discussed in the Clause Gadgets section.

Lemma 3.3. *Suppose we are given two unit disks D_1 and D_2 that intersect exclusively at a single point $q = D_1 \cap D_2$, and a line ℓ such that $q \in \ell$ and ℓ is tangent to both D_1 and D_2 (i.e., ℓ is the perpendicular bisector of the center points of D_1 and D_2). Now, given a point $p \in \ell$ where p is unit distance from q, the shortest path consisting of points $p, q_1 \in D_1$, and $q_2 \in D_2$ has weight $d \approx 0.755$.*

Proof. If $q_1 = q_2 = q$, then the path has unit length, so a path of length d is shorter. A path with edges $e_1 = (q_1, p)$ and $e_2 = (p, q_2)$ has length at least 0.828, since the nearest point on D_1 or D_2 to p is $\sqrt{2} - 1 > 0.414$ units distant.

Therefore, we may assume without loss of generality that the path consists of the edges $e_1 = (p, q_1)$ and $e_2 = (q_1, q_2)$ and the path has length $d = w(e_1) + w(e_2)$, where $w(e)$ is the length of the edge e. Therefore, we must choose q_1 and q_2 so that d is minimized. Note that candidate positions for each of q_1 and q_2 may be restricted to the boundaries of their respective disks.

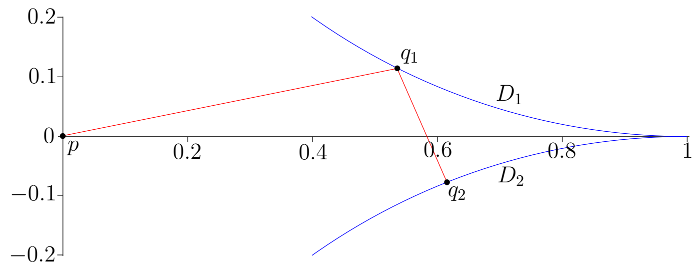


Figure 13: The shortest possible path is shown from a point at the origin to some point in each of two unit disks; one of the disks is centered at (1,1), the other is at (1,-1).

For the purposes of simplifying the proof, assume that p is at the origin of the Cartesian plane, and D_1 and D_2 are centered at $(1, 1)$ and $(1, -1)$, respectively. Then a point q_1 on the boundary of D_1 may be expressed as $(sin(\alpha) + 1, cos(\alpha) + 1)$, for some $\alpha \in [0 \dots 2\pi]$, and analogously $q_2 = (sin(\beta) + 1, cos(\beta) - 1)$, for some $\beta \in [0 \dots 2\pi]$. Therefore, we simply have to find the minimum of the function

$$f(\alpha, \beta) = \sqrt{(\sin\alpha + 1)^2 + (\cos\alpha + 1)^2} + \sqrt{(\sin\beta - \sin\alpha)^2 + (\cos\beta - \cos\alpha - 2)^2},$$

over the variables $\alpha \in [0 \dots 2\pi], \beta \in [0 \dots 2\pi]$. Using Maple, we see that this minimum has value $d \approx 0.755$, at $\alpha \approx 3.62, \beta \approx 5.89$. The optimal path in this setting is shown in Figure 13. Since this path is shorter than all other possible path configurations, we conclude that this is the shortest possible path including p and points $q_1 \in D_1$ and $q_2 \in D_2$. □

For the remainder of the discussion, we refer to the weight of this shortest path as the constant d. Before going to the details of the reduction, we consider optimal MSTN solutions when the problem instance is a variable gadget, as described above (without the wires approaching from clauses). We claim that such an instance has two possible MSTN solutions, and in each of these solutions consecutive pairs of disks are connected to a single wire of the k-star with a path of length d described in Lemma 3.3. We associate these two possible MSTN solutions with the two assignments for the variable. To prove the claim, we show that in an optimal MSTN solution for a k-star, there is no path containing points from more than two disks.

Lemma 3.4. *In an optimal MSTN solution for a k-star, a path containing a single wire of the k-star includes at most two disks from the k-flower, when $k \geq 8$.*

Proof. Recall that by Lemma 3.3, connecting a pair of disks to a k-wire may be done with weight d, while a wire may be connected to a single disk with weight $\sqrt{2}-1$. Therefore, three consecutive disks in a k-flower may be connected to two wires of the k-star using edges with weight $d+\sqrt{2}-1 \approx 1.169$, while four such disks may be connected with weight $2d \approx 1.51$.

Now consider three consecutive disks that we wish to connect to a single wire of the k-star. Given that $k \geq 8$, the minimum distance between the two non-adjacent disks is $d_{\min} \geq 2\sqrt{2+\sqrt{2}} - 2 \approx 1.696$. Therefore, a path simply connecting three disks (and yet still disjoint from the k-star) has greater weight than even the path joining four disks using two wires of the k-star, and thus an optimal path containing one wire of a k-star in the MST contains points from at most two disks of the k-flower. ☐

Corollary 3.5. *In the optimal MSTN solution for a k-flower (when k is even), each consecutive pair of disks is connected to a single wire of the k-star via a path of length d.*

This follows immediately from Lemmas 3.3 and 3.4. Hence, there are two possible solutions for MSTN on a k-flower where k is an even number (this is the case in our construction). We use this fact to assign a truth value for the variable gadget: one configuration is arbitrarily considered to be `true`, the other `false`. In Figure 12, we show an example where the `true` configuration is used, and every other wire of the k-star has an edge to some point in the k-flower. The `false` configuration would contain edges between the complementary set of wires of the k-star and the disks of the k-flower.

3.2.2 Clause Gadgets

The clause gadgets are composed of three wires that meet at a single point. Each wire of the clause gadget is placed so that it terminates at a distance $1 + d$ from a tangent point, where the terminal point is collinear with a line of the k-star on the relevant variable gadget. As a result, a line segment of length $2 + d$ units can connect the clause gadget to the k-star of a variable gadget, while also intersecting the shared point between two disks of the k-flower. If the truth value of the k-flower gadget matches that of the clause, this means that connecting the clause to the flower requires two units of extra weight, since otherwise the two disks are connected to the k-star with d weight, as outlined in Lemma 3.3. Therefore, given a clause gadget where at least one literal matches the truth value of the corresponding variable gadget, the clause gadget is connected to the MST with two units of additional weight.

The spinal path wires terminate in positions exactly analogous to those of the clause gadgets so that the analysis is the same. This raises the possibility that the wires of a clause gadget may be connected to two variable gadgets, leaving a gap in the spinal path, but note that such a configuration does not affect the weight of the optimal tree. The spinal path is necessary however, since some variables may not be used by any clauses in an optimal solution.

Lemma 3.6. *Joining a clause wire to a k-flower that has a truth value differing from that of the clause requires at least ≈ 0.845 units of additional edge weight relative to a configuration with matching truth values.*

Proof. In an optimal MSTN solution on a construction corresponding to a satisfiable 3SAT instance, a pair of disks and a clause wire may be joined to the k-star with weight $2+d$ units, and an additional adjacent pair of disks may be joined to the k-star with a path of weight d. Therefore, the total weight of the edges incident upon points in four such disks is $2 + 2d$.

Now consider a configuration where the truth value of the literal for each variable in a clause does not match the truth value of the corresponding variable gadgets. Connecting one of the clause gadget wires to the k-star requires an additional weight of $2 + d$, as discussed previously, which intersects points from two disks; call them D_i and D_{i+1}. The neighboring two disks in the k-flower, D_{i-1} and D_{i+2}, are not attached to the k-star by paths like those found in Lemma 3.3. Rather, each of these adjacent paths may be shortened to $\sqrt{2}$ to cover the two singleton disks. Note that there

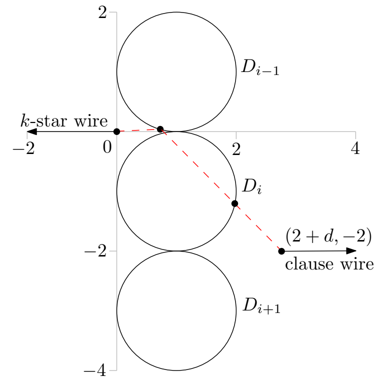


Figure 14: The shortest possible path is shown (the dashed line) from the end of the clause wire to points in D_i and D_{i-1}, and finally connecting to the k-star wire for D_i and D_{i-1}.

may be a non-empty sequence of pairs of disks connected as in Lemma 3.3 before the singleton is reached, creating a section of the flower with an inverted truth value for the variable[4]. Therefore, the net extra weight of such a transition is $2 + d + 2\sqrt{2} - (2 + 2d) = 2\sqrt{2} - d \approx 2.0735$.

A configuration that may require less additional weight is to connect the clause wire to the k-star using a path with points in disks D_{i-1} and D_i (it is a slightly modified configuration from that of Lemma 3.3). As k increases, the weight of such a path decreases. To minimize the length of the path, suppose that the centers of D_{i-1}, D_i and D_{i+1} are collinear (which occurs when $k = \infty$). Therefore, we can place the center of D_{i-1} at $(1, 1)$, the end of the k-star wire between D_{i-1} and D_i at $(0, 0)$, and the end of the clause wire at $(2 + d, -2)$ (Figure 14). The weight of the path from the k-star to D_{i-1} to the clause wire may be expressed by the function

$$f(\theta) = \sqrt{(1 + \sin\theta)^2 + (1 + \cos\theta)^2} + \sqrt{(1 + d - \sin\theta)^2 + (-3 - \cos\theta)^2},$$

which has a minimum length slightly greater than 3.60 units at $\theta \approx 3.49$ radians. Since this path intersects D_i, it is also the shortest path that includes a point $p_i \in D_i$. Therefore, w.l.o.g. a path connecting a clause wire to a wire in a variable gadget with a mismatched truth value has weight greater than 3.6. Note that such a path does not affect the truth value of the variable gadget, and so D_{i+1} and D_{i+2} may be joined to the k-star with a path of weight d. Therefore, the extra weight incurred for such a configuration is $> 3.6 + d - (2d + 2) \approx 0.845$. □

As described earlier, the terminal points of the clause wires (and the spinal path) are collinear with wires of the k-star. Since we never place these terminal points on adjacent wires of the k-star, the wires need not lie within 4 units of one another, and so there will not be edges directly between different clause wires or between a clause wire and the spinal path.

3.2.3 Reduction

We would like to reduce a given instance of planar 3-SAT to the MSTN problem. Note that the given 3-SAT instance is assumed to be embedded on the plane, and there exists a spinal path

[4] D_{i+2} may be more generally indexed as D_{i+2+4c}, where there is a block of $4c$ disks in the k-flower joined to the k-star in a truth configuration opposite of that of the neighboring disks in the k-flower. This does not affect the analysis, it simply relocates the singleton disk. Recall that by Lemma 3.4, such singletons would exist rather than having three disks connected by a path to a single edge of the k-star.

$P = (V_v, E_P)$ that passes all variable vertices without crossing any edge of G, such that all variable vertices but 2 have degree 2 in P (as mentioned at the beginning of Section 3.2, this restricted version is also NP-hard).

To create the instance of the MSTN problem, we fix the spinal path as a part of the MST, using wires consisting of disks of radius 0. We replace each variable node with a variable gadget as explained. Each clause gadget includes three wires, which we place so that they approach the associated variable gadgets as described.

The wires forming the spinal path, the m clause gadgets, and each of the n k-stars have a fixed weight, call the total weight of all these wires w_{wires}. The remaining weight of the MST is that of connecting to a point from each disk in the k-flowers, and that of connecting each clause gadget. Suppose there exists a satisfying assignment for the 3-SAT instance. Each pair of disks in the k-flowers can be connected with weight d; this will be the case for all but m pairs. The remaining m pairs will be connected with edges that also join to the clause gadgets in the manner described in Section 3.2.2 with weight $2 + d$. Therefore, assuming that there is a total of i pairs of disks in the k-flowers of the construction, the remaining weight of the MST is $w_{\text{disks}} = id + 2m$. Thus, if there exists a satisfying assignment to the 3-SAT instance, the total optimal weight of the MST is $w_{\text{tot}} = w_{\text{wires}} + w_{\text{disks}}$.

If there is no satisfying assignment, at least one of the clause gadgets must be connected to the MST in the manner described in Lemma 3.6, which requires an additional weight of at least 0.845. Now suppose there exists an FPTAS for MSTN. Given an instance of planar 3-SAT, we build the MSTN construction and determine w_{tot}. We choose a value of ε so that $\varepsilon < 0.845/w_{\text{tot}}$, and so a $(1 + \varepsilon)$-approximate solution to the MSTN problem may be used to determine whether there is a satisfying assignment for the planar 3-SAT instance. Since the latter problem is NP-hard, we conclude that MSTN does not admit an FPTAS unless P=NP.

4 Conclusions

We considered geometric MST with neighborhoods problems, and established that computing the MST of minimum or maximum weight is hard to approximate in this setting by proving that there is no FPTAS for either problem, assuming $\mathsf{P} \neq \mathsf{NP}$. We provided a parameterized algorithm for the MSTN problem based upon how well separated the disks are from one another. For MAX-MSTN, we showed that a deterministic algorithm that selects disk centers gives an approximation ratio of 1/2. Furthermore, we showed that when the instance of the problem satisfies k-separability, the same approach achieves a constant approximation ratio of $1 - \frac{2}{k+4}$.

For further research, it will be interesting to study this problem under different models of imprecision. Depending on the application, the regions of uncertainty may consist of other shapes, e.g., line segments, rectangles, etc., or they may be composed of discrete sets of points.

References

[1] Arkin, E., Hassin, R.: Approximation algorithms for the geometric covering salesman problem. Discrete Applied Mathematics 55(3), 197 – 218 (1994)

[2] de Berg, M., Gudmundsson, J., Katz, M., Levcopoulos, C., Overmars, M., van der Stappen, A.: TSP with neighborhoods of varying size. Journal of Algorithms 57(1), 22 – 36 (2005)

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

[4] Dumitrescu, A., Mitchell, J.S.: Approximation algorithms for TSP with neighborhoods in the plane. Journal of Algorithms 48(1), 135 – 159 (2003)

[5] Erlebach, T., Hoffmann, M., Krizanc, D., Mihalák, M., Raman, R.: Computing minimum spanning trees with uncertainty. In: Symposium on Theoretical Aspects of Computer Science. pp. 277–288 (2008)

[6] Fiala, J., Kratochvíl, J., Proskurowski, A.: Systems of distant representatives. Discrete Applied Mathematics 145(2), 306–316 (2005)

[7] Graham, R.L., Hell, P.: On the history of the minimum spanning tree problem. IEEE Annals of the History of Computing 7(1), 43–57 (1985)

[8] Lichtenstein, D.: Planar formulae and their uses. SIAM J. on Computing 11(2), 329–344 (1982)

[9] Löffler, M., van Kreveld, M.: Largest and smallest convex hulls for imprecise points. Algorithmica 56, 235–269 (2010)

[10] Nešetřil, J., Milková, E., Nešetřilová, H.: Otakar Borůvka on minimum spanning tree problem translation of both the 1926 papers, comments, history. Discrete Mathematics 233(13), 3 – 36 (2001)

[11] Sekino, J.: n-ellipses and the minimum distance sum problem. The American Mathematical Monthly 106(3), pp. 193–202 (1999)

[12] Yang, Y.: On several geometric network design problems. Ph.D. thesis, State University of New York at Buffalo (2008)

[13] Yang, Y., Lin, M., Xu, J., Xie, Y.: Minimum spanning tree with neighborhoods. In: Algorithmic Aspects in Information and Management, pp. 306–316 (2007)