Noname manuscript No.
(will be inserted by the editor)

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

Reza Dorrigiv · Robert Fraser · Meng He · Shahin Kamali · Akitoshi Kawamura · Alejandro López-Ortiz · Diego Seco

Received: date / Accepted: date

Abstract We study optimization problems for the Euclidean Minimum Spanning Tree (MST) problem 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

A preliminary version of this work appeared at the 2012 Workshop on Approximation and Online Algorithms (WAOA) [11].

Reza Dorrigiv
Dalhousie University, Halifax, Canada
E-mail: rdorrigiv@cs.dal.ca

Robert Fraser
University of Manitoba, Winnipeg, Canada
E-mail: fraser@cs.umanitoba.ca

Meng He
Dalhousie University, Halifax, Canada
E-mail: mhe@cs.dal.ca

Shahin Kamali
University of Waterloo, Waterloo, Canada
E-mail: s3kamali@uwaterloo.ca

Akitoshi Kawamura
University of Tokyo, Tokyo, Japan
E-mail: kawamura@is.s.u-tokyo.ac.jp

Alejandro López-Ortiz
University of Waterloo, Waterloo, Canada
E-mail: alopez-o@uwaterloo.ca

Diego Seco
University of Concepción, Concepción, Chile
E-mail: dseco@udec.cl

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.

Keywords Computational Geometry, Imprecise Data, Minimum Spanning Trees

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 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 (MSTN) problem. 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 points $p_i \in D_i$ for each $D_i \in D$ so that the weight of the MST of the selected points is maximized.

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

- MAX-MSTN: 1/2-approximation algorithm;
- MAX-MSTN: parameterized $\left(1 - \frac{2}{k+4}\right)$-approximation algorithm (where k represents the *separability* of the instance);
- MAX-MSTN: proof of NP-hardness;
- MSTN: parameterized $(1 + 2/k)$-approximation algorithm (k is the *separability* of the instance);
- MSTN: proof of NP-hardness.

The approximation algorithm for MAX-MSTN (Section 4.1) is based on choosing the centre points of the disks; the interesting aspect in this section lies in the analysis. The parameterized algorithms (Sections 4.2 and 5.1) 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 centres with respect to separability.

For both hardness results, we establish that not only are the problems NP-hard, but also 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 then show that the weight of an optimal solution to the given problem on our construction may be determined *a priori*, using the assumption 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 0.66 units for MSTN).

2 Related Work

The first known MST algorithm was published over 80 years ago [20], and a number of successful variants have followed (see [19] for the history of the problem). A review of models of uncertainty and data imprecision for computational geometry problems is provided in [22]. Here, we discuss a few results that are directly related to the MST problem and our model of imprecision.

The MSTN problem on disjoint unit disks has been shown to admit a PTAS [25]. An NP-hardness proof for a generalization of MSTN where the neighborhoods are either disks or rectangles appeared in [25]. This proof was faulty however, and one of the authors later conjectured that a reduction from planar 3-SAT might be used to show the NP-hardness of the MSTN problem [24, p.106]. In Section 5.2, we prove this conjecture. The hardness of MSTN is of interest, and the previous result has been cited a number of times, e.g. [9, 10].

When regions of uncertainty are modelled as disks or squares, even the problem of maximizing the smallest pairwise distance in a set of n imprecise points is NP-hard [17].

Löffler and van Kreveld [22] discussed two results on computing Minimum Spanning Trees under the model of uncertainty that we are considering. First, they demonstrated that it is algebraically difficult (see Section 3) 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. Secondly, they demonstrated that the problem is also 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. They also presented a number of new results for the convex hull problem on imprecise points.

Erlebach et al. [15] 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 weight of the MST given regions of uncertainty.

The minimum k-spanning tree (k-MST) problem asks for the minimum weight MST over a set of $k \leq n$ points, where n is the number of vertices in the input. This was shown to be NP-hard by reduction from the Steiner tree problem [18]. The Euclidean version of k-MST was shown to admit a PTAS by Arora [2], as a corollary of his Traveling Salesman Problem result.

The Travelling Salesman Problem with Neighborhoods (TSPN) 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 [12]. The most general version of the problem, where regions may overlap and may have varying sizes, is known to be APX-hard [5].

3 Algebraic Complexity

There are two main types of complexity that one encounters when working with optimization problems in computational geometry: algebraic complexity and combinatorial complexity. In this section, we provide a brief discussion of the former.

Many geometric optimization problems that are simple to formulate and conceptualize can nonetheless have solutions that are difficult to compute. Borrowing from an example in [4] and [22], consider the problem of computing a minimum weight tree over a set of points, where the tree consists of one root vertex of degree k and k leaf vertices of degree one. If we know the positions of all of the leaves and simply wish to compute an optimal position for the root vertex, then this is an algebraically difficult problem. Formally, suppose we have points $P = \{p_1, \ldots, p_k\}$, where $p_i = (x_i, y_i)$, and we wish to place the point $c = (x, y)$ in the plane. Then the optimal weight w of the tree is

$$w = \min_{x,y} \sum_{i=1}^{k} \sqrt{(x_i - x)^2 + (y_i - y)^2} \tag{1}$$

The problem is well known as the Fermat-Weber problem, and Bajaj [4, Table 1] shows that even for $k = 5$, that the solution to this problem requires the determination of the roots of a polynomial of degree 8, for which there is no known solution. The problem was first proposed by Fermat for problems where $k = 3$ [16], while exact solutions to the settings for $k = 3$ and $k = 4$ were found by Cavalieri in 1647 (with a tightening by Heinen in 1834) and Fagnano in 1775, respectively [3].

Durocher and Kirkpatrick [13] studied the Fermat-Weber problem, and they suggest several approximation schemes which maintain a robust approximation while removing the algebraic complexity from the problem. The most basic approach is to compute the centre of mass of the points, which provides a 2-approximation for the position of the Fermat-Weber point (where the value of the optimal solution is the value of w in Equation 1).

3.1 Euclidean MST Problems

We note that Euclidean MST problems are not known to be in NP, since comparing two candidate solutions is an instance of the sum of square roots problem[1], which is not known to be in NP. The sum of square roots problem accepts two sequences of integers as input, $A = a_1, \ldots, a_m$ and $B = b_1, \ldots, b_n$, and is asked to determine whether $\sum_{i=1}^{m} \sqrt{a_i} > \sum_{i=1}^{n} \sqrt{b_i}$. While the status of the general version of the sum of square roots problem is unknown, there is a randomized polynomial time algorithm to determine whether $\sum_{i=1}^{m} \sqrt{a_i} = \sum_{i=1}^{n} \sqrt{b_i}$ [7]. Every MST problem in this paper is studied under the Euclidean distance metric, and so are generalizations of the standard Euclidean MST problem and are not known to be in NP.

4 MAX-MSTN

In this section we study a few approximation algorithms for the MAX-MSTN problem, and then we present the proof of hardness of approximation. We begin with a 1/2-approximation algorithm below, followed by a parameterized algorithm in Section 4.2.

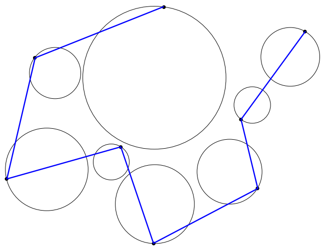
4.1 1/2-Approximation Algorithm on Disjoint Disks

To approximate the solution to MAX-MSTN, we first consider the algorithm that builds an MST on the centres of the disks. We show this algorithm approximates the optimal solution within a factor of 1/2, i.e., the weight of the MST built on the centres is not smaller than half of that of the optimal tree.

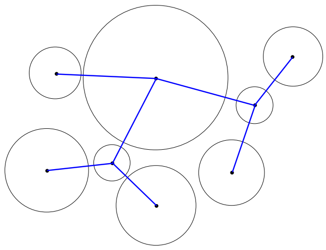
Theorem 1 *Consider the* MAX-MSTN *problem for a set D of disjoint disks. Let T_c denote the MST on the centres of the disks, and let T_{OPT} be the maximum MST (i.e., the optimal solution to the problem). Then $w(T_c) \geq 1/2 \cdot w(T_{\text{OPT}})$.*

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

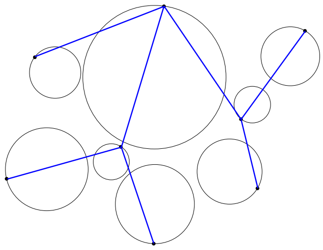
[1] For an informal discussion of sum-of-square-roots-hard problems, see [14].



(a) The optimal result for MAX-MSTN (T_{OPT}).



(b) The MST T_c on centres.



(c) The spanning tree T_c' with the same topology as T_c, using the points of T_{OPT}.

Fig. 1 To compare $w(T_c)$ with $w(T_{\text{OPT}})$, we use an intermediate tree T_c'.

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 $w(T_{\text{OPT}}) \le w(T_c')$. On the other hand, since T_c' and T_c have the same topology, we have $w(T_c') \le 2w(T_c)$; this is because when we move the points from the centre 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 $w(T_{\text{OPT}}) \le w(T_c')$ and $w(T_c') \le 2w(T_c)$, which completes the proof. □

4.2 Parameterized $\left(1 - \frac{2}{k+4}\right)$-Approximation Algorithm on Disjoint Disks

Observe that in order to get the approximation algorithm for MAX-MSTN in Section 4.1, we require the disks to be disjoint. Intuitively, if we know that the 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[2]. Let $r_{\max}$ be the maximum radius of any disk in the input set. 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

[2] Separability is similar in spirit to the notion of a well-separated pair; see [8].

I is defined as the maximum k such that I satisfies k-separability. With this definition, we have the following result:

Theorem 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 centres of the disks achieves a constant approximation ratio of $\frac{k+2}{k+4} = 1 - \frac{2}{k+4}$.*

Proof Let T_c be the MST on the centres 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 $\frac{k+2}{k+4} = 1 - \frac{2}{k+4}$. □

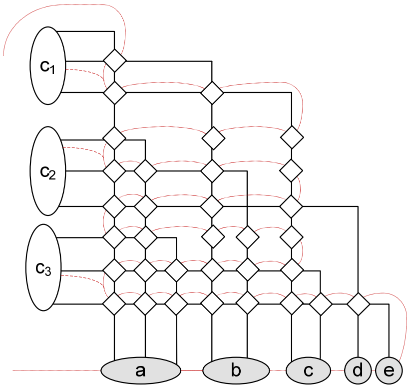
The approximation ratio gets arbitrarily close to 1 as k increases. This confirms our intuition that if the disks are further apart (more separate), we get a better approximation factor.

4.3 NP-Hardness of MAX-MSTN

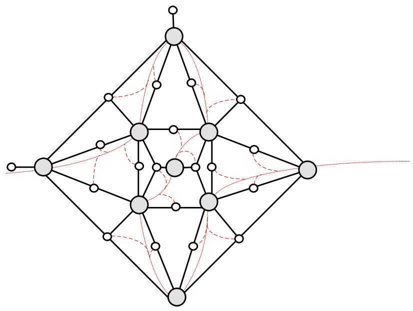
We present a hardness proof for the MAX-MSTN problem with a reduction from the planar 3-SAT problem. Planar 3-SAT is a variant of 3-SAT in which the graph $G = (V, E)$ associated with the formula is planar.

Theorem 3 MAX-MSTN *is* NP*-hard, and it 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 [21] 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 addition of a path through all the vertices V_v while maintaining planarity. We call this path the *spinal path* and denote it



(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.

Fig. 2 The reduction from 3-SAT to planar 3-SAT as presented in [21]. (a) The variable and clause vertices of 3-SAT are located on the x and y axes respectively, and the edges are drawn as orthogonal paths. (b) A planar gadget is placed on each intersection point. Each gadget introduces some new variable and clause vertices. In [21], it is observed that there is a path (we call it the spinal path) that covers all variable vertices of the 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).

by $P = (V_v, E_P)$. We further observe that additional edges can be added to P to get a *spinal tree* $T = (V, E_T)$ 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 ensure that a polynomial number of disks are sufficient for the reduction, we use a special embedding of the graph in the plane in which the edges are orthogonally drawn. We use the following theorem, where m and n are the number of edges and vertices of the graph respectively:

Theorem 4 *[6, Theorem 4] Let H be a simple graph without nodes of degree ≤ 1. Then H has an orthogonal drawing in an $\frac{m+n}{2} \times \frac{m+n}{2}$-grid with one bend per edge. The box size of each node v is at most $\frac{\deg(v)}{2} \times \frac{\deg(v)}{2}$. It can be found in $O(m)$ time.*

We use an orthogonal drawing of the planar 3-SAT instance in our reduction, although we will expand the drawing by a factor of 2 so that all edges of the drawing are separated by at least 2 units. All edges of the drawing will be replaced by *wires* in the reduction, where a wire consists of a set of points (disks of radius 0) placed along an edge of the drawing every unit distance. This ensures that each wire has a unique MST over the points, and

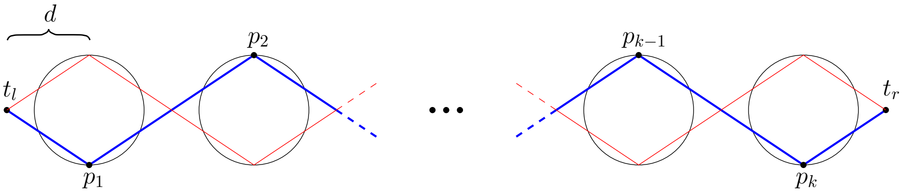


Fig. 3 The two possibilities for the MAX-MSTN solution for chain of unit disks. Here we have $d = 1.5$.

that the number of points required for all wires is polynomial in the size of the input. The gadgets used in the reduction may not fit within boxes of size $\frac{\deg(v)}{2} \times \frac{\deg(v)}{2}$, but they are within a constant factor of this bound. The variable gadgets require boxes of size at most $15.75\,(\deg(v)+1) \times 11$ (we discuss these gadgets shortly), while clause gadgets require boxes of (constant) size 108×52 units. Note that this growth also requires extending some wires, and this growth may be bounded by m times the length of the relevant side(s) of the box for each gadget. Therefore, the size of the drawing remains polynomial in the size of the input provided the gadgets also require a polynomial number of disks (and we will show that this is the case).

Next, we study an important theorem for our reduction which states that an optimal solution for MAX-MSTN on a *chain* of unit disks has a characteristic zigzag pattern. A chain is a set of k unit disks $\mathcal{D}^c = \{D_1^c, \ldots, D_k^c\}$ whose centres are collinear, incident upon a horizontal line ℓ_{hz}, and adjacent disk centres 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 centre of the leftmost disk, and t_r is located d units to the right of the centre of the rightmost disk of the chain.

Theorem 5 *Given a chain of disks, the solution to the* MAX-MSTN *problem on the chain $\mathcal{D}^c$ 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 zigzag paths that traverse the extreme top and bottom points of the disks (see Figure 3).*

We define Z_D as this proposed zigzag path that alternates between the extreme upper and lower points of a set of disks $\mathcal{D}^c$, where the centres of all disks in $\mathcal{D}^c$ are collinear. Note that the MST for the chain of disks with points in such positions forms a polygonal chain starting from t_l and ending at t_r.

To prove Theorem 5, we require a few additional lemmas.

Lemma 1 *Given a disk, if the set of edges containing an endpoint in the disk is fixed for any position of the point, any* MAX-MSTN *solution does not contain a point on the interior of 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 [23]. 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 an optimal MAX-MSTN solution follows the path of a ray initiated at point t_l and reflected at a point of intersection with each disk.

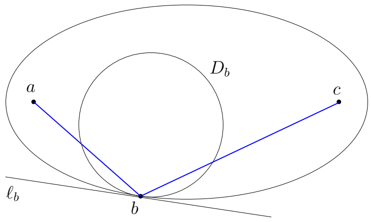
Lemma 2 *Let p_1 be the selected point of a* MAX-MSTN *solution on the leftmost disk in a chain. Consider the ray $\overrightarrow{t_l p_1}$ which is then reflected on the interior face of each disk. Any valid* MAX-MSTN *solution follows the path traversed by the ray, i.e., the two neighboring segments 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 $a = t_l$ and $b = p_1$). Let D_b be the disk such that $b \in D_b$, and by Lemma 1, 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 set of points $\mathcal{P}$ that are equidistant, on the aggregate, to a and c, i.e., $\text{dist}(a, p) + \text{dist}(p, c)$ is a constant for all $p \in \mathcal{P}$. Points inside the ellipse are closer, on the aggregate, to a and c while points outside the ellipse are farther.

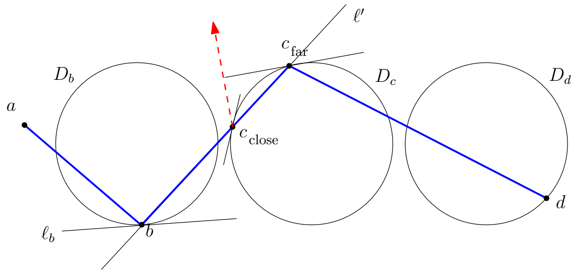
If the disk D_b and an ellipse incident upon b are not tangent at b, it follows that there are points in D_b outside the ellipse and hence the length of the MST grows if such a point is chosen instead of b, which contradicts the assumption that the MST through b is maximal. Since a and c are the foci, the projective property of the ellipse implies that $\overrightarrow{bc}$ is the reflection of $\overrightarrow{ab}$ on the tangent ℓ_b to the ellipse.

Let c_{close} and 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 $\overleftrightarrow{bc_{\text{far}}}$, 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 at 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 then inductively apply the same reasoning by replacing a with b and b with c. □

Lemma 2 implies that selecting the first point p_1 on the leftmost disk D_1 defines all other points that are selected by the MAX-MSTN algorithm. In particular, if p_1 is the extreme top or bottom point of D_1, then the points selected for all other disks will be on the extreme top or bottom points of the disks as well, which produces the Z_D configuration described in Theorem 5. Note that if the reflected line on any disk does not intersect the next disk to the right, or if at the rightmost disk the reflected line is not incident upon



(a) The ellipse with foci a and c and incident upon b should be tangent to D_b if $\overline{ab}$ and $\overline{bc}$ are edges in a MAX-MSTN solution.



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

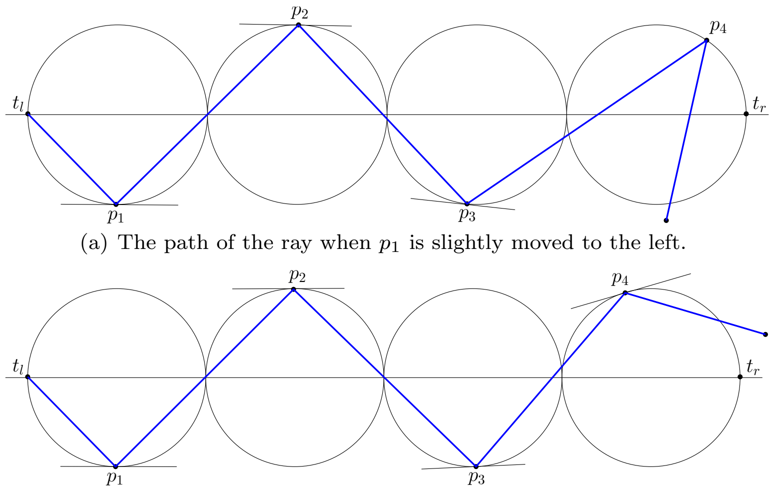
Fig. 4 The reflection effect.

the terminal point t_r, then in these cases the initial selection for p_1 does not produce a MAX-MSTN solution (Figure 5).

Lemma 3 *Let $\mathcal{P} = \{p_1, p_2, \ldots, p_n\}$ be the selected points of a* MAX-MSTN *solution on the consecutive disks in a horizontal chain, from left to right. Then for any $i \in \{1, \ldots, n\}$ we have the following two properties:*

1. *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.*
2. *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 To establish the first property, let p_i and p_{i+1} both be on the bottom halves 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 centres). Such a transformation 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 halves of consecutive disks). Using mathematical induction, the second property is direct from Lemma 2. More precisely, if p_{i-1} and p_i are on the right and left halves of their disks respectively, then Lemma 2 dictates that p_{i+1} is on the right half of D_{i+1}. Similarly, if p_{i-1} and p_i are on the left and right halves of their disks respectively, then p_{i+1} is on the left half of D_{i+1} (this effect may be seen in Figure 5). $\square$



(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.

Fig. 5 If the selected point p_1 on the leftmost disk is not the extreme top or bottom point, then the path of the ray does not pass through t_r. Note that $d = 1$ in this illustration.

To prove Theorem 5, we show that if we select p_1 as any point except the extreme bottom or top points of the leftmost disk D_1, then the path described in Lemma 2 does not pass through the terminal point t_r. For each disk $D_i \in D$, define the *canonical line* as the line passing through the centre of D_i and p_i, and the *canonical angle* α_i as the acute angle between ℓ_{hz} and the canonical line of the disk if the lines are not perpendicular. Note that Lemma 3 implies that if the canonical lines have defined slope, then either all canonical lines have positive slope or all have negative slope. To make the explanation simpler, consider another disk D_0 centred at a distance $2d$ to the left of the centre of the leftmost disk, and let p_0 be the intersection of D_0 with the line passing through t_l and p_1 (Figure 6). Since t_l is located at the midpoint of the centres of p_0 and p_1, observe that the canonical angles of D_0 and D_1 are equal regardless of the choice of p_1, i.e., $\alpha_0 = \alpha_1$.

Lemma 4 *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., $\pi/2 > \alpha_0 = \alpha_1 > \alpha_2 > \ldots > \alpha_n$.*

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., $\alpha_1 \leq \alpha_2$. Note that the slopes of all canonical lines are either positive or negative. Without loss of generality assume all slopes are negative (see Figure 6). Let p_x be the intersection point of the line passing through p_1 and p_2 with the horizontal line ℓ_{hz}. dist(p_x, q_1) is the distance between p_x and the centre point q_1 of D_1. Since $\alpha_1 \leq \alpha_2$, we have dist$(p_x, q_1) \leq d$. Let θ be the angle between lines $\overline{p_0p_1}$ and $\overline{q_1p_1}$, which is the same as the angle between $\overline{p_1p_2}$ and $\overline{q_1p_1}$

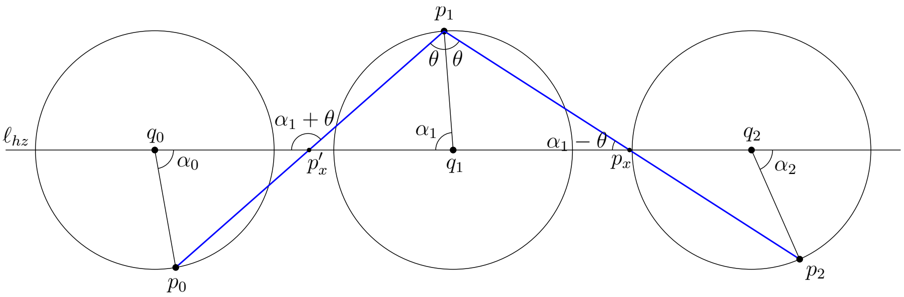


Fig. 6 If $\alpha_0 < \pi/2$, the canonical angles are strictly decreasing.

(by the reflection law). Applying the Law of Sines for triangle $\triangle q_1p_1p_x$ we get dist $(p_x, q_1) = \sin\theta/\sin(\alpha_1 - \theta)$. Let p'_x be the intersection point of $\overline{p_0p_1}$ and horizontal line ℓ_{hz} (which is t_l in the base case). Since $\alpha_0 \geq \alpha_1$, we have dist $(p'_x, q_1) \geq d$ (equivalence occurs in the base case). Applying the Law of Sines on triangle $\triangle q_1p_1p'_x$, we get dist $(p'_x, q_1) = \sin\theta/\sin(\pi - \alpha_1 - \theta)$.

Since dist $(p_x, q_1) \leq d$ and dist $(p'_x, q_1) \geq d$, we get $\sin\theta/\sin(\alpha_1 - \theta) \leq \sin\theta/\sin(\pi - \alpha_1 - \theta)$, which implies that $\alpha_1 \geq \pi/2$; 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 5.

Proof Lemma 2 implies that the MAX-MSTN solution should follow the path of a ray that is initiated at t_l, reflected at each point p_i of disk D_i, and finally passes through t_r. Note that the two Z_D (zigzag) paths stated in the theorem have this property. We show that no other ray can do so. Assume the selected point of the leftmost disk is not on the extreme top or bottom (otherwise the reflected path would be one of the paths described in the theorem). By Lemma 4, the canonical angles of the disks are strictly decreasing. This implies that for any disk D_i, the reflected path does not pass through the mid-point of the segment between q_i and q_{i+1} (the centres of D_i and D_{i+1}, respectively). Otherwise, the canonical angles of the two disks would be equal. In particular, if one assumes the existence of an extra disk on the right, the reflected path does not pass through 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 5, 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 (i.e. points) placed in close succession so that we may interpret them as a fixed line in the MAX-MSTN 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 correspond to 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.

4.3.1 Variable Gadgets

For each variable x_i, we build a set $\mathcal{D}^i$ consisting of $3c + 2$ unit disks in the configuration described in Theorem 5, spaced by $d = 21/8 = 2.625$, where c is the number of clauses containing instances of the variable (note that adjacent disk centres are $2d = 5.25$ units apart). 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 centres of $\mathcal{D}^i$ 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 6.5 units from a disk centre, along a line incident upon the disk centre and perpendicular to ℓ_{hz}, i.e., the terminal point of the wire and the disk centre share the same x-coordinate (see Figure 7). We provide more details regarding e-wires shortly. Finally, suppose without loss of generality that disk $D_j^i \in \mathcal{D}^i$ 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 5 *Suppose we are given a variable gadget where points are placed in the disks as described in Theorem 5. Then a point in a disk is either distance* 7.5 *or* 5.5 *from the nearest point in an e-wire, and we may arrange it so that these distances correspond to agreement or disagreement respectively between*

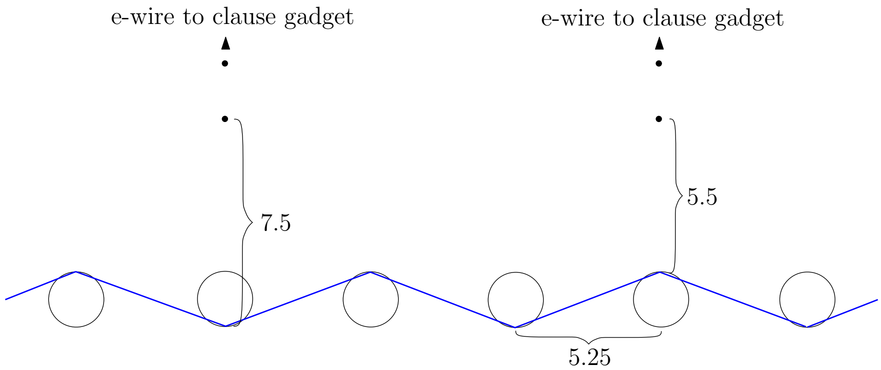


Fig. 7 The configuration used for the variable gadgets. Here, two clauses include the variable in the opposite truth values. Filled small circles represent disks of radius 0, while larger empty circles represent unit disks. E-wires are placed so that the nearest point on any disk is at distance 5.5, and the terminal point of the e-wire is on a line that is vertical and incident upon the disk centre. 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 7.5 units distant, while a mismatch in truth values means the nearest point in the gadget would only be distance 5.5 away. The disk centres 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.

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 5, we arbitrarily select one of the 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 7.5. However, a mismatched truth value would mean that a point lies only distance 5.5 from the e-wire when the points are in the Z_D configuration. □

4.3.2 Clause Gadgets

For each clause in the 3-SAT instance, we build a clause gadget by assembling unit disks and wires (composed of disks of zero radius spaced by two units) as shown in Figure 8. The unit disks are placed in chain, either horizontally or vertically, using $d = 4$ for spacing (i.e., disks are centred 4 units away from the two terminal points, and disks are placed with centres 8 units apart from each other). From the spinal tree, a wire joins to one of the terminal disks of a chain.

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

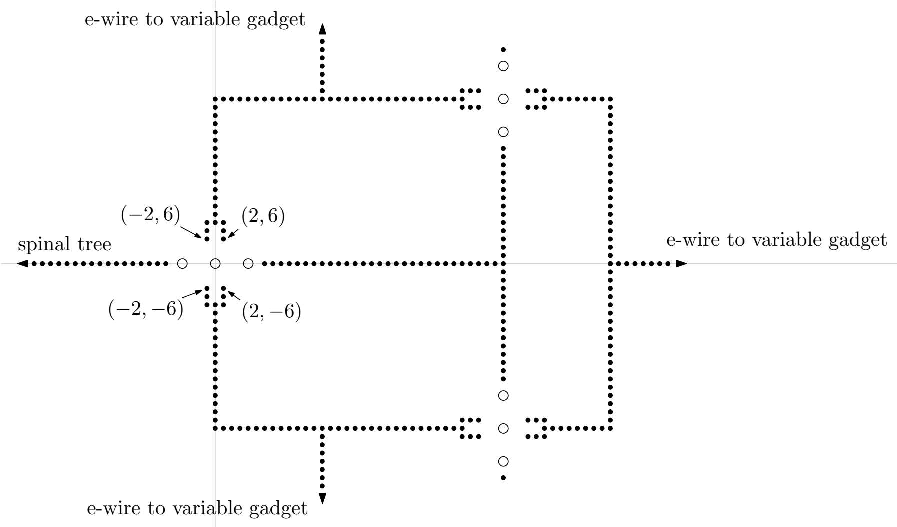


Fig. 8 The configuration used for the clause gadgets, with the gadget placed in the canonical position. The filled small circles (black dots) are disks of radius 0, while the larger empty circles are unit disks. A wire extends from the gadget as part of the spinal tree, and three e-wires go to three variable gadgets (one for each literal in the corresponding clause of the 3-SAT instance).

near the variable gadget lies at distance 5.5 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 8. To be specific, suppose that the construction is positioned so that the centre point of the middle disk of the horizontal chain is placed at the origin of the plane. Call this the *canonical position* of the gadget. 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. The configurations of the two vertical chains are analogous.

Supposing points were selected in these three disks in a zigzag Z_D configuration (shown to be an optimal configuration in Theorem 5), then the edges between points in adjacent disks have weight $2\sqrt{17} \approx 8.25$, and those from the first and last disk to the nearest points along the ray have weight $\sqrt{17} \approx 4.12$. The distance from the point in the middle disk to the nearest point on one e-wire is $\sqrt{29} \approx 5.39$, and to the nearest point on the opposite e-wire is $\sqrt{53} \approx 7.28$.

Lemma 6 *Suppose that a 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 of the chain is $\sqrt{(8-2)^2+(0-6)^2}-1 \approx 7.49$, 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 of the chain.

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 1. Suppose without loss of generality 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 disk centred at the origin is maximized at either $(0,1)$ or $(0,-1)$, and so the lemma follows.

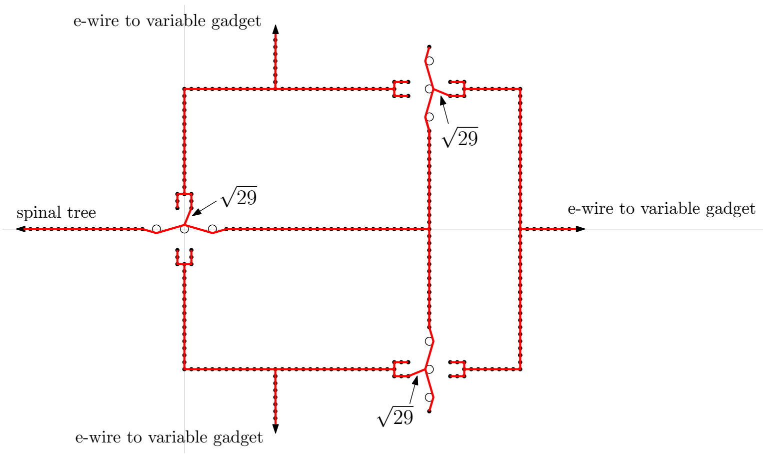
The argument is analogous for the two vertical chains of disks in the gadget. □

Lemma 7 *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 $\sqrt{53}$.*

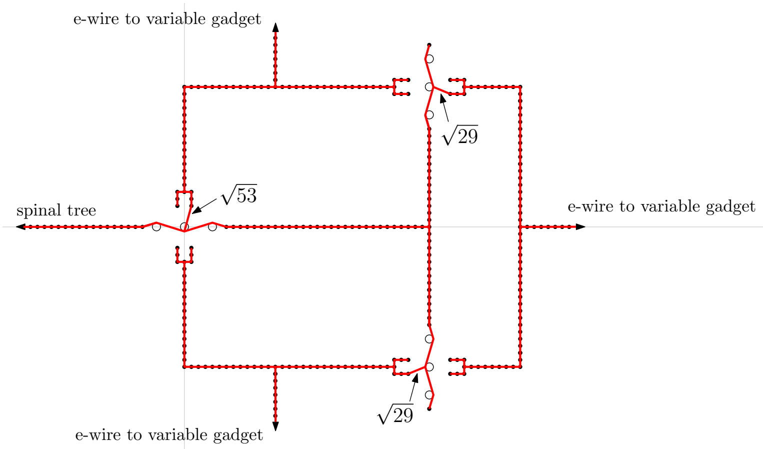
Proof By the proof of Lemma 6, we know that all clause gadgets will use the Z_D configuration through their disks. Therefore, the weight of an edge between the clause gadget and an e-wire is either $\sqrt{29}$ or $\sqrt{53}$. Further, because two e-wires approach each set of disks, if one e-wire is distance $\sqrt{29}$ 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 (Figure 9(a)).
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 (Figure 9(b)).

Since we are computing a Minimum Spanning Tree with maximum weight, the second configuration is preferable. □



(a) A suboptimal configuration.



(b) An optimal configuration. Note that the top left portion of the gadget requires an edge of weight $\sqrt{53}$ in order to be connected to the remainder of the gadget (one of the four possible such edges is shown).

Fig. 9 Paths through the MAX-MSTN clause gadget.

4.3.3 Reduction

The key to the reduction is that 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, if and only if there exists a satisfying assignment to the given planar 3-SAT instance. 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 6. 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 5.5 or 7.5, by Lemma 5. Since at least one of the literals may be satisfied in the clause (by our assumptions), the corresponding e-wire, call it e_{sat}, is distance 7.39 from the nearest point in the 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 7. 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 5, 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 a fixed weight, consisting of the weight of all wires, plus $2\sqrt{29} + \sqrt{53} \approx 240.05$ for each gadget to join the e-wires to the disks. 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[3].

Now consider the case that there is no satisfying assignment for the 3-SAT instance. In particular, consider a truth assignment (defined by the alignment of zigzag 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 zigzag paths;

[3] Note that the number of edges in the construction is a polynomial in the size of the input. Since all we need to determine is whether the weight of the solution is at least 0.34 units less than w_{tot}, we can round the measure of the total weight to a number of bits that is logarithmic in the size of the input.

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 7. 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[4]. 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 zigzag pattern. That is, assume without loss of generality that at some point 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 one disk between them at minimum. To find an appropriate bound, first we assume that the closest points to each e-wire, call these p_ℓ and p_r, remain in the positions that would be optimal for a Z_D configuration, i.e., at the farthest 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.

Case 1: Two separating disks (Figure 10). Without loss of generality, 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 centred at $(5.25, 0)$, and p_2 is in one centred 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:

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

[4] If the Z_D configurations were maintained, then the e-wire could be joined sub-optimally to the variable gadget with weight 5.5, since the assignment is not satisfying. However, a larger MST can be achieved by deviating from the Z_D configuration in the variable gadget.

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 (the weight remains greater than 16.48).

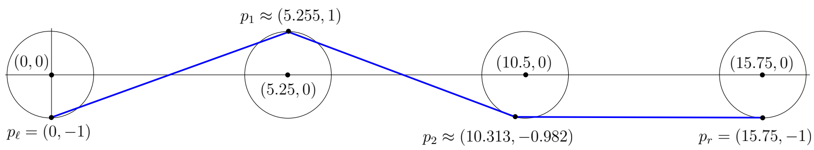


Fig. 10 The optimal setting for Case 1. The optimal point in the first separating disk is slightly to the right of the extreme upper point in the disk, while that of the second disk is nearly 11° clockwise of the extreme lower point. Despite these differences, the change in path weight between this optimal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

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 centred 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).

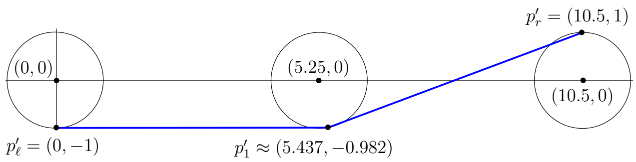


Fig. 11 The optimal setting for Case 2. The abberation of the point in the separating disk from the extreme lower point is similar to that of the second separating disk in Case 1, and the difference in path weight between the optimal path and that using only the extreme points is less than 0.01 units.

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 centred 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 5 that any changes in position from the extreme points in the remainder of the gadget only decrease 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. By Theorem 4, we have that the number of disks (including points) required for the construction is polynomial in the size of the input, and all points and disk centres are either on the integer grid or else have rational coordinates where the denominator is a constant (all square roots in the discussion are only a property of the analysis). This establishes the NP-hardness of the MAX-MSTN problem.

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. We cannot rule out the possibility of a PTAS, because the gap between solutions on satisfiable and non-satisfiable instances that we have demonstrated is not a factor that is proportional to the size of the optimal solution on the MAX-MSTN construction.

5 MSTN

In this section we present a parameterized approximation algorithm for the MSTN problem, followed by the proof of NP-hardness. Recall that for MSTN, we seek a minimum weight spanning tree on imprecise input represented as disks.

5.1 Parameterized $(1 + 2/k)$-Approximation Algorithm on Disjoint Disks

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

Theorem 6 *For* MSTN *when the regions of uncertainty are disjoint disks with separability parameter $k > 0$, the algorithm that builds an MST on the centres of the disks achieves an approximation factor 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 centres 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 T_m 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 centres. Since T_c is an MST on centres, we have $w(T_c) \leq w(T_m)$. Consider an arbitrary edge e in T_m 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 T_m 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 T_m, we get $w(T_c) \leq w(T_m) \leq \frac{k+2}{k} w(T_{\text{OPT}})$. 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. The principal difference between this analysis and that of the MAX-MSTN setting (Section 4.2) is that the intermediate spanning tree T_m is defined as the tree whose vertices are disk centres and whose topology is the same as T_{OPT}, whereas T'_c is the spanning tree with the same topology as T_c but on the vertices of T_{OPT}.

5.2 NP-Hardness of MSTN

To prove the hardness of the MSTN problem, we present a reduction from the planar 3-SAT problem. Planar 3-SAT is a variant of 3-SAT in which the graph $G = (V, E)$ associated with the formula is planar (see Figure 2 and the associated text for further details).

Theorem 7 MSTN *is* NP*-hard. Furthermore,* MSTN *does not admit an FPTAS, unless* P=NP.

In the hardness proof of MAX-MSTN (Section 4.3), we used a spinal tree in the reduction. In this section, we use the *spinal path*, defined 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, this restricted version of planar 3-SAT remains NP-hard [21]. 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. A wire is a set of disks of radius 0 placed in close succession (usually 2 units apart) so that we may interpret them 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 the optimal MSTN solution has a weight less than k (for some value k dependent on the input) if and only if the SAT formula is satisfiable.

As in Section 4.3, we use an orthogonal drawing of the planar 3-SAT instance plus the spinal path in our reduction, expanded by a factor of 2. All edges of the drawing are replaced by *wires*, where a wire consists of a set of points (disks of radius 0) placed along an edge of the drawing every unit distance so that each wire has a unique MST over the points and the number of points required for all wires is polynomial in the size of the input (see Theorem 4). Again, the gadgets used in the reduction may not fit within boxes of size $\frac{\deg(v)}{2} \times \frac{\deg(v)}{2}$, but they are within a constant factor of this bound. The variable gadgets require boxes of size at most $(8\deg(v)+5) \times 15$ units (we discuss these gadgets shortly), while clause gadgets consist of a single point. Therefore, the size of the drawing remains polynomial in the size of the input.

5.2.1 Variable Gadgets

A variable gadget is formed by k unit disks, where $k \leq 8m_i + 6$ and m_i is the number of clauses in the planar 3-SAT instance that include the variable. Each edge associated with a clause requires 4 disks, each edge of the spinal path requires 3 disks, and we may need to add extra disks as discussed in Figure 12. Disks are placed 2 units apart so that each disk is tangent to its two neighboring disks, and each pair of consecutive disks D_i, D_{i+1} around the gadget intersects at a single point $q_{i,i+1} = D_i \cap D_{i+1}$, which we call a *tangent point*[5]. Moreover, there is a set of points in the middle of the gadget, whose MST has fixed weight (although the topology is not unique). There are k terminal points to this MST, each unit distance from the nearest tangent point, i.e. there exists a point t_i for each $q_{i,i+1}$ so that $\text{dist}(t_i, q_{i,i+1}) = 1$. 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 2.5 units from the nearest tangent point, for reasons discussed in the Clause Gadgets section.

Lemma 8 *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 centre points of D_1 and D_2). Now, given a point $p \in \ell$ where p is unit distance from*

[5] Using this construction, pairs of disks of the gadget trivially intersect at a single point, which simplifies our analysis. To achieve strict disjointedness, the disks of the gadget 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 $n(8m+6)$ disks, where n and m are the number of variables and clauses respectively. Choosing an appropriate value of γ so that $2\gamma n(8m+6) \ll 0.66$ achieves the same result as our simplified analysis.

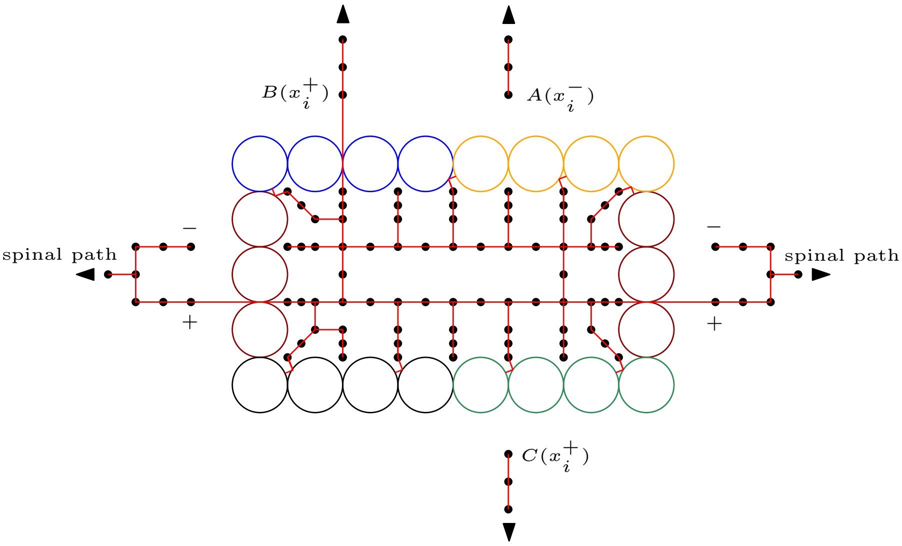


Fig. 12 A variable gadget contains up to $6 + 8m_i$ disks for a variable x_i, where m_i is the number of clauses containing literals of the variable. In this case, the gadget contains 22 disks. $B(x_i^+)$ and $C(x_i^+)$ are the endpoints of the wires that connect to clauses that include x_i in the positive form, while $A(x_i^-)$ terminates a wire connecting to a clause that includes x_i in negative form. The figure illustrates the case in which the algorithm has used the positive configuration for x_i, and clause B is satisfied by x. Clause C is satisfied via some other variable, as is clause A (assuming that it is satisfiable). Note that disks are connected to the wires inside the gadget in pairs. The gadget contains 4 disks for each wire from a clause gadget, and extra disks are used if there are differing numbers of wires approaching the top and bottom (this is why the upper bound has the factor $8m_i$, since it may happen that all m_i wires approach the top or bottom only).

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 $w(e_i)$ is the length of the edge e_i. 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.

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 centred 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 $\alpha \in [0 \ldots 2\pi]$, and analogously we define $q_2 = (sin(\beta) + 1, cos(\beta) - 1)$, for some $\beta \in [0 \ldots 2\pi]$. 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 $\alpha \in [0 \ldots 2\pi], \beta \in [0 \ldots 2\pi]$. 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 $q_2 \in D_2$. □

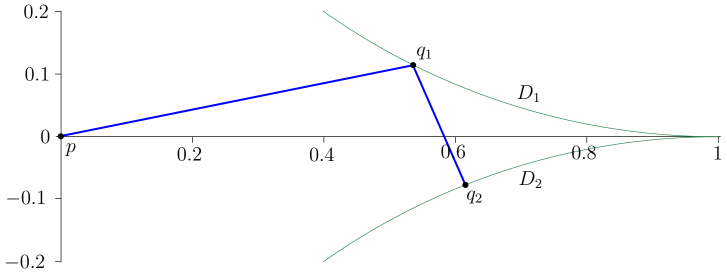


Fig. 13 The shortest possible path is shown from a point at the origin to some point in each of two unit disks; one of the disks is centred at $(1, 1)$, the other is at $(1, -1)$.

For the remainder of the discussion, we refer to the weight of this shortest path as the constant $d \approx 0.755$. 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 interior of the gadget with a path of length d described in Lemma 8. 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 the interior of the gadget, there is no path containing points from more than two disks.

Lemma 9 *In the optimal MSTN solution for a variable gadget, each consecutive pair of disks is connected to a distinct point on the interior of the gadget via a path of length d.*

Proof Recall that by Lemma 8, connecting a pair of disks in the configuration shown in Figure 13 may be done with a path of weight d, while the same point may be connected to a single disk with weight $\sqrt{2} - 1$. Therefore, three consecutive disks in a variable gadget may be connected to two wires of the interior of the gadget 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 interior of the gadget. The minimum distance between any two non-adjacent disks is $d_{\min} \geq 2\left(\sqrt{2} - 1\right)$, which occurs in the corners. Note that a path spanning this nearest distance passes through the point of the interior of the gadget nearest the three disks in the corner, and so the shortest path to connect to the third disk in the corner requires another edge of $\sqrt{2} - 1$. Two of these adjacent paths may now be replaced by a path of weight d, as shown in Lemma 8, and so the weight of the shortest path joining three consecutive disks to a point on the interior of the gadget is $d + \sqrt{2} - 1 \approx 1.169$. Since $d + \sqrt{2} - 1 > 3d/2$, and there are an even number of disks in the gadget, an optimal path containing one point of the interior of a variable gadget in the MST contains points from at most two disks of the variable gadget. □

Hence, there are two possible solutions for MSTN on a variable gadget. 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 interior of the gadget has an edge to some point in the disks. The `false` configuration would contain edges between the complementary set of wires of the interior of the gadget and the disks of the variable gadget.

5.2.2 Clause Gadgets

The clause gadgets are composed of three wires that meet at a single point. Each wire associated with the clause gadget is placed so that it terminates at a distance 2.5 from a tangent point of a variable gadget, as shown in Figure 12. As a result, a line segment of length 3.5 units can connect the clause gadget to the interior of a variable gadget, while also intersecting the shared point between two disks. If the truth value of the variable gadget matches that of the clause, this means that connecting the clause gadget to the variable gadget requires $3.5 - d$ units of extra weight, since otherwise the two disks are connected to the interior of the gadget with d weight, as outlined in Lemma 8. 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 $3.5 - d$ 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 10 *Joining a clause wire to a variable gadget that has a truth value differing from that of the clause requires at least* $4.16 - (3.5) \approx 0.66$ *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 interior of the gadget with weight 3.5 units, and an additional adjacent pair of disks may be joined to the interior of the gadget with a path of weight d. Therefore, the total weight of the edges incident upon points in four such disks is $3.5 + d$.

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 interior of the gadget requires edges of weight of at least 3.5 units, as discussed previously, and the shortest path intersects points from two disks; call them D_i and D_{i+1}. The neighboring two disks in the variable gadget, D_{i-1} and D_{i+2}, are not attached to the interior of the gadget by paths like those found in Lemma 8. Rather, each of these adjacent paths may be shortened to $\sqrt{2}$ to cover the two singleton disks. Note that there may be a non-empty sequence of pairs of disks connected as in Lemma 8 before the singleton is reached, creating a section of the gadget with an inverted truth value for the variable[6]. Therefore, the net extra weight of such a transition is $3.5 + 2\sqrt{2} - (3.5 + d) = 2\sqrt{2} - d \approx 2.0735$.

A configuration that may require less additional weight is to connect the clause wire to the interior of the gadget using a path with points in disks D_{i-1} and D_i (it is a slightly modified configuration from that of Lemma 8). For our analysis, we can place the centre of D_{i-1} at $(1, 1)$, the end of the interior wire between D_{i-1} and D_i at $(0, 0)$, and the end of the clause wire at $(3.5, -2)$ (Figure 14). The weight of the path from the interior of the gadget to D_{i-1} to the clause wire may be expressed by the function

$$f(\theta) = \sqrt{(1 + \sin\theta)^2 + (1 + \cos\theta)^2} + \sqrt{(2.5 - \sin\theta)^2 + (-3 - \cos\theta)^2},$$

which has a minimum weight greater than 4.16 units at $\theta \approx 3.43$ radians. Since this path intersects D_i, it is also the shortest path that includes a point $p_i \in D_i$. Therefore, without loss of generality a path connecting a clause wire to a wire in a variable gadget with a mismatched truth value has weight greater than 4.16. 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 interior of the gadget with a path of weight d. Therefore, the extra weight incurred for such a configuration is greater than $4.16 + d - (3.5 + d) \approx 0.66$. □

As described earlier, the terminal points of the clause wires (and the spinal path) are collinear with wires of the interior of the gadget. Note that the wires approaching the variable gadget 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.

[6] D_{i+2} may be more generally indexed as D_{i+2+4c}, where there is a block of $4c$ disks in the variable gadget joined to the interior of the gadget in a truth configuration opposite of that of the neighboring disks in the variable gadget. This does not affect the analysis, it simply relocates the singleton disk. Recall that by Lemma 9, such singletons would exist rather than having three disks connected by a path to a single edge of the interior of the gadget.

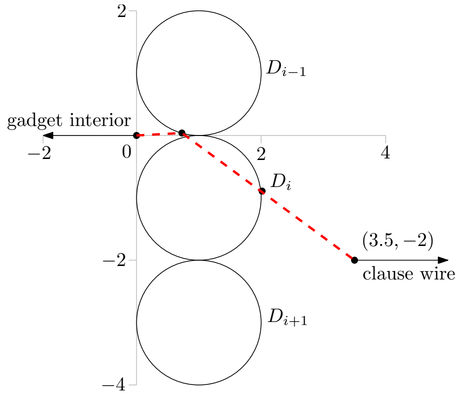


Fig. 14 The shortest possible path is shown (the dashed line) from the end of the clause wire to points in D_i and D_{i-1}, and finally connecting to the interior of the gadget for D_i and D_{i-1}.

5.2.3 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 in the plane, and there exists a spinal path $P = (V_v, E_P)$ that passes through all variable vertices without crossing any edge of G (as mentioned at the beginning of Section 5.2, 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 variable gadget interiors have a fixed weight, call the total weight of all these wires w_{wires}. The remaining weight of the MST is that of connecting a point from each disk in the variable gadgets, and that of connecting each clause gadget. Suppose there exists a satisfying assignment for the 3-SAT instance. Each pair of disks in the variable gadgets 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 with weight 3.5 in the manner described in the Variable Gadget section above. Therefore, assuming that there is a total of i pairs of disks in the variable gadgets of the construction, the remaining weight of the MST is $w_{\text{disks}} = id + (3.5 - d)m$. Thus, if there exists a satisfying assignment to the 3-SAT instance, the total optimal weight of the MST is $w_{\text{tot}} = w_{\text{wires}} + w_{\text{disks}}$[7]. If there is no satisfying assignment, at least one

[7] Note that the number of edges in the construction is a polynomial in the size of the input. Since all we need to determine is whether the weight of the solution is at least 0.66

of the clause gadgets must be connected to the MST in the manner described in Lemma 10, which requires an additional weight of at least 0.66 units.

All points in the construction were placed on the integer grid or else have fractional coordinates where the denominator is 2, and the size of the grid required for the construction is polynomially bounded in the size of the input. This establishes the NP-hardness of MSTN.

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 $\varepsilon < 0.66/w_{\text{tot}}$, 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.

6 Conclusions and Future Work

We considered geometric MST with Neighborhoods problems, and established that computing the MST of minimum or maximum weight is NP-hard in both cases, even when disks are disjoint. We extended the hardness results to show that the problems are hard to approximate by proving that there is no FPTAS for either problem, assuming $\mathsf{P} \neq \mathsf{NP}$. For MAX-MSTN, we showed that a deterministic algorithm which selects disk centres 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 $1 - \frac{2}{k+4}$. Finally, we provided a parameterized algorithm for MSTN based upon how well separated the disks are from one another. The algorithm finds a solution with a weight within a factor of $(1 + 2/k)$ of an optimal solution, where k is the separability of the instance.

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. Arora, S.: Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems. Journal of the ACM **45**(5), 753–782 (1998)
3. Bajaj, C.: Geometric optimization and computational complexity. Ph.D. thesis, Cornell University (1984)
4. Bajaj, C.: The algebraic degree of geometric optimization problems. Discrete and Computational Geometry **3**(1), 177 – 191 (1988).
5. 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)

units greater than w_{tot}, we can round the measure of the total weight to a number of bits logarithmic in the size of the input.

6. Biedl, T., Kaufmann, M.: Area-efficient static and incremental graph drawings. In: Proceedings of the European Symposium on Algorithms, *Lecture Notes in Computer Science*, vol. 1284, pp. 37–52 (1997)
7. Blömer, J.: Computing sums of radicals in polynomial time. In: Proceedings of the Symposium on Foundations of Computer Science (FOCS), pp. 670–677. IEEE Computer Society (1991)
8. Callahan, P.B., Kosaraju, S.R.: A decomposition of multidimensional point sets with applications to k-nearest-neighbors and n-body potential fields. Journal of the ACM **42**(1), 67–90 (1995)
9. Chambers, E., Erickson, A., Fekete, S., Lenchner, J., Sember, J., Venkatesh, S., Stege, U., Stolpner, S., Weibel, C., Whitesides, S.: Connectivity graphs of uncertainty regions. In: Proceedings of the International Symposium on Algorithms and Computation (ISAAC), *Lecture Notes in Computer Science*, vol. 6507, pp. 434–445 (2010)
10. Chou, C.C.: An efficient algorithm for relay placement in a ring sensor networks. Expert Systems with Applications **37**(7), 4830 – 4841 (2010)
11. Dorrigiv, R., Fraser, R., He, M., Kamali, S., Kawamura, A., López-Ortiz, A., Seco., D.: On minimum- and maximum-weight minimum spanning trees with neighborhoods. In: Proceedings of the 2012 Workshop on Approximation and Online Algorithms (WAOA), to appear.
12. Dumitrescu, A., Mitchell, J.S.: Approximation algorithms for TSP with neighborhoods in the plane. Journal of Algorithms **48**(1), 135 – 159 (2003).
13. Durocher, S., Kirkpatrick, D.: The projection median of a set of points. Computational Geometry **42**(5), 364 – 375 (2009). Special Issue on the Canadian Conference on Computational Geometry (CCCG 2005 and CCCG 2006)
14. Erickson, J. (user JɛffE): Sum-of-square-roots-hard problems? *Theoretical Computer Science on StackExchange*, URL: `http://cstheory.stackexchange.com/questions/4053/sum-of-square-roots-hard-problems` (2011).
15. Erlebach, T., Hoffmann, M., Krizanc, D., Mihalák, M., Raman, R.: Computing minimum spanning trees with uncertainty. In: Proceedings of the Symposium on Theoretical Aspects of Computer Science (STACS), pp. 277–288 (2008)
16. de Fermat, P.: Oeuvres de Fermat (Tome 1). Gauthier-Villars et Fils, Paris (1891)
17. Fiala, J., Kratochvíl, J., Proskurowski, A.: Systems of distant representatives. Discrete Applied Mathematics **145**(2), 306–316 (2005)
18. Fischetti, M., Hamacher, H.W., Jørnsten, K., Maffioli, F.: Weighted k-cardinality trees: Complexity and polyhedral structure. Networks **24**(1), 11–21 (1994)
19. 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)
20. Jarník, V.: O jistém problému minimálním (About a certain minimal problem). Práca Moravské Prírodovedecké Spolecnosti **6**, 57–63 (1930). (in Czech, German summary)
21. Lichtenstein, D.: Planar formulae and their uses. SIAM Journal on Computing **11**(2), 329–343 (1982)
22. Löffler, M., van Kreveld, M.: Largest and smallest convex hulls for imprecise points. Algorithmica **56**(2), 235–269 (2010).
23. Sekino, J.: n-ellipses and the minimum distance sum problem. The American Mathematical Monthly **106**(3), pp. 193–202 (1999)
24. Yang, Y.: On several geometric network design problems. Ph.D. thesis, State University of New York at Buffalo (2008)
25. Yang, Y., Lin, M., Xu, J., Xie, Y.: Minimum spanning tree with neighborhoods. In: Proceedings of Algorithmic Aspects in Information and Management (AAIM), *Lecture Notes in Computer Science*, vol. 4508, pp. 306–316 (2007)