Geometric Spanning Trees Minimizing the Wiener Index

A. Karim Abu-Affash ✉
Software Engineering Department, Shamoon College of Engineering, Beer-Sheva, Israel

Paz Carmi ✉
Computer Science Department, Ben-Gurion University, Beer-Sheva, Israel

Ori Luwisch ✉
Computer Science Department, Ben-Gurion University, Beer-Sheva, Israel

Joseph S. B. Mitchell ✉
Department of Applied Mathematics and Statistics, Stony Brook University, Stony Brook, NY, USA

Abstract

The Wiener index of a network, introduced by the chemist Harry Wiener [31], is the sum of distances between all pairs of nodes in the network. This index, originally used in chemical graph representations of the non-hydrogen atoms of a molecule, is considered to be a fundamental and useful network descriptor. We study the problem of constructing geometric networks on point sets in Euclidean space that minimize the Wiener index: given a set P of n points in $\mathbb{R}^d$, the goal is to construct a network, spanning P and satisfying certain constraints, that minimizes the Wiener index among the allowable class of spanning networks.

In this work, we focus mainly on spanning networks that are trees and we focus on problems in the plane ($d = 2$). We show that any spanning tree that minimizes the Wiener index has non-crossing edges in the plane. Then, we use this fact to devise an $O(n^4)$-time algorithm that constructs a spanning tree of minimum Wiener index for points in convex position. We also prove that the problem of computing a spanning tree on P whose Wiener index is at most W, while having total (Euclidean) weight at most B, is NP-hard.

Computing a tree that minimizes the Wiener index has been studied in the area of communication networks, where it is known as the *minimum routing cost spanning tree problem*.

Keywords and phrases Wiener Index, Hardness proof, Minimum routing cost spanning tree, Minimum routing cost, Hamiltonian path

Digital Object Identifier 10.57717/cgt.v3i1.52

Funding This work was partially supported by Grant 2016116 from the United States – Israel Binational Science Foundation. Work by P. Carmi and O. Luwisch was partially supported by the Lynn and William Frankel Center for Computer Sciences. Work by J. Mitchell was partially supported by NSF (CCF-2007275) and by DARPA (Lagrange).

1 Introduction

The *Wiener index* of a weighted graph $G = (V, E)$ is the sum, $\sum_{u,v \in V} \delta_G(u, v)$, of the shortest path lengths in the graph between every pair of vertices, where $\delta_G(u, v)$ is the weight of the shortest (minimum-weight) path between u and v in G. The Wiener index was introduced by the chemist Harry Wiener in 1947 [31]. The Wiener index and its several variations have found applications in chemistry, e.g., in predicting the antibacterial activity of drugs and modeling crystalline phenomena. It has also been used to give insight into various chemical and physical properties of molecules [29] and to correlate the structure of molecules with their biological activity [20]. The Wiener index has become part of the general scientific culture, and it is still the subject of intensive research [2, 10, 12, 34]. In its applications in chemistry, the Wiener index is most often studied in the context of unweighted graphs.

The study of minimizing the sum of interpoint distances also arises naturally in the network design field, where the problem of computing a spanning tree of minimum Wiener index is known as the *Minimum Routing Cost Spanning Tree* (MRCST) problem [18, 15].

Given an undirected graph $G = (V, E)$ and a (nonnegative) weight function on the edges of G, representing the delay on each edge, the routing cost $c(T)$ of a spanning tree T of G is the sum of the weights (delays) of the paths in T between every pair of vertices: $c(T) = \sum_{u,v\in V} \delta_T(u, v)$, where $\delta_T(u, v)$ is the weight of the (unique) path between u and v in T. The MRCST problem aims to find a minimum routing cost spanning tree of a given weighted undirected graph G, thereby seeking to minimize the expected cost of a path within the tree between two randomly chosen vertices. The MRCST was originally introduced by Hu [18] and is known to be NP-complete in graphs, even if all edge weights are 1 [19]. Wu et al. [33] presented a polynomial time approximation scheme (PTAS) for the MRCST problem. Specifically, they showed that the best k-star (a tree with at most k internal vertices) yields a $(\frac{k+3}{k+1})$-approximation for the problem, resulting in a $(1 + \varepsilon)$-approximation algorithm of running time $O\big(n^{2\lceil \frac{2}{\varepsilon}\rceil-2}\big)$.

While there is an abundance of research related to the Wiener index, e.g., computing and bounding the Wiener indexes of specific graphs or classes of graphs [16, 17, 25] and explicit formulas for the Wiener index for special classes of graphs [3, 23, 27, 31, 30], to the best of our knowledge, the Wiener index has not received much attention in geometric settings. In this work, we study the Wiener index and the minimum routing cost spanning tree problem in selected geometric settings, hoping to bring this important and highly applicable index to the attention of computational geometry researchers.

1.1 Our contributions and overview

Let P be a set of n points in the plane. We study the problem of computing a spanning tree of P that minimizes the Wiener index when the underlying graph is the complete graph over P, with edge weights given by their Euclidean lengths. In Section 2, we prove that the optimal tree (that minimizes the Wiener index) is non-crossing (i.e., has no crossing edges). As our main algorithmic result, in Section 3, we give a polynomial-time algorithm to solve the problem when the points of P are in convex position; this result strongly utilizes the structural result that the edges of an optimal tree do not cross, which enables us to devise a dynamic programming algorithm. Then, in Section 4, we prove that the "Euclidean Wiener Index Tree Problem", in which we seek a spanning tree on P whose Wiener index is at most W, while having total (Euclidean) weight at most B, is (weakly) NP-hard. Finally, in Section 5, we discuss the problem of finding a minimum Wiener index *path* spanning P.

1.2 Related work

A problem related to ours is the minimum latency problem, also known as the traveling repairman problem TRP: Compute a path, starting at point s, that visits all points, while minimizing the sum of the distances (the "latencies") along the path from s to every other point (versus between *all* pairs of points, as in the Wiener index). There is a PTAS for TRP (and the k-TRP, with k repairmen) in the Euclidean plane and in weighted planar graphs [28].

Wiener index optimization also arises in the context of computing a noncontracting embedding of one metric space into another (e.g., a line metric or a tree metric) in order to minimize the average distortion of the embedding (defined to be the sum of all pairs distances in the new space, divided by the sum of all pairs distances in the original space). It is NP-hard to minimize average distortion when embedding a tree metric into a line metric; there is a constant-factor approximation (based on the k-TRP) for minimizing the average distortion in embedding a metric onto a line (i.e., finding a Hamiltonian path of minimum Wiener index) [11], which, using [28], gives a $(2+\varepsilon)$-approximation in the Euclidean plane.

A related problem that has recently been examined in a geometric setting is the computation of the Beer index of a polygon P, defined to be the probability that two randomly (uniformly) distributed points in P are visible to each other [1]; the same paper also studies the problem of computing the expected distance between two random points in a polygon, which is, like the Wiener index, based on computing the sum of distances (evaluated as an integral in the continuum) between all pairs of points.

Another area of research that is related to the Wiener index is that of *spanners*: Given a weighted graph G and a real number $t > 1$, a t*-spanner* of G is a spanning sub-graph G^* of G, such that $\delta_{G^*}(u,v) \leq t \cdot \delta_G(u,v)$, for every two vertices u and v in G. Thus, the shortest path distances in G^* approximate the shortest path distances in the underlying graph G, and the parameter t represents the approximation ratio. The smallest t for which G^* is a t-spanner of G is known as the *stretch factor*. There is a vast literature on spanners, especially in geometry (see, e.g., [4, 5, 6, 7, 8, 13, 22, 24, 26]). In a geometric graph, G, the *stretch factor* between two vertices, u and v, is the ratio between the length of the shortest path from u to v in G and the Euclidean distance between u and v. The *average stretch factor* of G is the average stretch factor taken over all pairs of vertices in G. For a given weighted connected graph $G = (V,E)$ with positive edge weights and a positive value W, the *average stretch factor spanning tree* problem seeks a spanning tree T of G such that the average stretch factor (over $\binom{n}{2}$ pairs of vertices) is bounded by W. For points in the Euclidean plane, one can construct in polynomial time a spanning tree with constant average stretch factor [9].

2 Preliminaries

Let P be a set of n points in the plane and let $G = (P,E)$ be the complete graph over P. For each edge $(p,q) \in E$, let $w(p,q) = |pq|$ denote the weight of (p,q), given by the Euclidean distance, $|pq|$, between p and q. Let T be a spanning tree of P. For points $p,q \in P$, let $\delta_T(p,q)$ denote the weight of the (unique) path between p and q in T. Let $W(T) = \sum_{p,q \in P} \delta_T(p,q)$ denote the Wiener index of T, given by the sum of the weights of the paths in T between every pair of points. For a point $p \in P$, let $\delta_p(T) = \sum_{q \in P} \delta_T(p,q)$ denote the total weight of the paths in T from p to every point of P. A spanning tree of P is called *non-crossing* if no two of its edges cross each other.

▶ **Theorem 1.** *Let T be a spanning tree of P that minimizes the Wiener index $W(T)$. Then, T is non-crossing.*

Proof. Assume towards a contradiction that there are two edges (a,c) and (b,d) in T that cross each other. Let F be the forest obtained by removing the edges (a,c) and (b,d) from T. Thus, F contains three sub-trees. Assume, without loss of generality, that a and b are in the same sub-tree T_{ab}, and c and d are in separated sub-trees T_c and T_d, respectively; see

Figure 1. Let n_{ab}, n_c, and n_d be the number of points in T_{ab}, T_c, and T_d, respectively. Thus,

$$\begin{aligned} W(T) = {} & W(T_{ab}) + n_c \cdot \delta_a(T_{ab}) + n_d \cdot \delta_b(T_{ab}) \\ & + W(T_c) + (n_{ab} + n_d) \cdot \delta_c(T_c) + n_c(n_{ab} + n_d) \cdot |ac| \\ & + W(T_d) + (n_{ab} + n_c) \cdot \delta_d(T_d) + n_d(n_{ab} + n_c) \cdot |bd| \\ & + n_c \cdot n_d \cdot \delta_T(a, b)\,. \end{aligned}$$

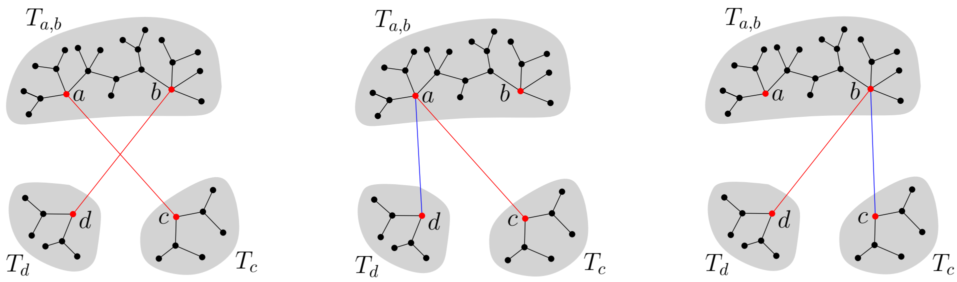

Figure 1 The trees T, T', and T'' (from left to right).

Let T' be the spanning tree of P obtained from T by replacing the edge (b, d) by the edge (a, d). Similarly, let T'' be the spanning tree of P obtained from T by replacing the edge (a, c) by the edge (b, c). Thus,

$$\begin{aligned} W(T') = {} & W(T_{ab}) + (n_c + n_d) \cdot \delta_a(T_{ab}) \\ & + W(T_c) + (n_{ab} + n_d) \cdot \delta_c(T_c) + n_c(n_{ab} + n_d) \cdot |ac| \\ & + W(T_d) + (n_{ab} + n_c) \cdot \delta_d(T_d) + n_d(n_{ab} + n_c) \cdot |ad|\,, \end{aligned}$$

and

$$\begin{aligned} W(T'') = {} & W(T_{ab}) + (n_c + n_d) \cdot \delta_b(T_{ab}) \\ & + W(T_c) + (n_{ab} + n_d) \cdot \delta_c(T_c) + n_c(n_{ab} + n_d) \cdot |bc| \\ & + W(T_d) + (n_{ab} + n_c) \cdot \delta_d(T_d) + n_d(n_{ab} + n_c) \cdot |bd|\,. \end{aligned}$$

Therefore,

$$W(T) - W(T') = n_d\big(\delta_b(T_{ab}) - \delta_a(T_{ab})\big) + n_d(n_{ab} + n_c)\big(|bd| - |ad|\big) + n_c \cdot n_d \cdot \delta_T(a, b)\,,$$

and

$$W(T) - W(T'') = n_c\big(\delta_a(T_{ab}) - \delta_b(T_{ab})\big) + n_c(n_{ab} + n_d)\big(|ac| - |bc|\big) + n_c \cdot n_d \cdot \delta_T(a, b)\,.$$

If $W(T) - W(T') > 0$ or $W(T) - W(T'') > 0$, then this contradicts the minimality of T, and we are done.

Assume that $W(T) - W(T') \leq 0$ and $W(T) - W(T'') \leq 0$. Since $n_c > 0$ and $n_d > 0$, we have

$$\delta_b(T_{ab}) - \delta_a(T_{ab}) + (n_{ab} + n_c)\big(|bd| - |ad|\big) + n_c \cdot \delta_T(a, b) \leq 0\,,$$

and

$$\delta_a(T_{ab}) - \delta_b(T_{ab}) + (n_{ab} + n_d)\big(|ac| - |bc|\big) + n_d \cdot \delta_T(a, b) \leq 0\,.$$

Thus, by summing these inequalities, we have

$$(n_{ab}+n_c)\big(|bd|-|ad|\big)+(n_{ab}+n_d)\big(|ac|-|bc|\big)+(n_c+n_d)\cdot\delta_T(a,b)\le 0\,.$$

That is,

$$n_{ab}\big(|bd|+|ac|-|ad|-|bc|\big)+n_c\big(|bd|+\delta_T(a,b)-|ad|\big)+n_d\big(|ac|+\delta_T(a,b)-|bc|\big)\le 0\,.$$

Since $n_{ab}, n_c, n_d > 0$, and, by the triangle inequality, $|bd| + |ac| - |ad| - |bc| > 0$, $|bd| + \delta_T(a,b) - |ad| > 0$, and $|ac| + \delta_T(a,b) - |bc| > 0$, this is a contradiction. ◀

3 An exact algorithm for points in convex position

Let $P = \{p_1, p_2, \ldots, p_n\}$ be a set of n points in convex position in the plane, ordered in clockwise-order with an arbitrary first point p_1; see Figure 2. For simplicity of presentation, we assume that all indices are taken modulo n. For each $1 \le i \le j \le n$, let $P[i,j] \subseteq P$ be the set $\{p_i, p_{i+1}, \ldots, p_j\}$. Let $T_{i,j}$ be a spanning tree of $P[i,j]$, and let $W(T_{i,j})$ denote its Wiener index. For a point $x \in \{i,j\}$, let $\delta_x(T_{i,j})$ be the total weight of the paths from p_x to every point of $P[i,j]$ in $T_{i,j}$. That is $\delta_x(T_{i,j}) = \sum_{p\in P[i,j]} \delta_{T_{i,j}}(p_x, p)$.

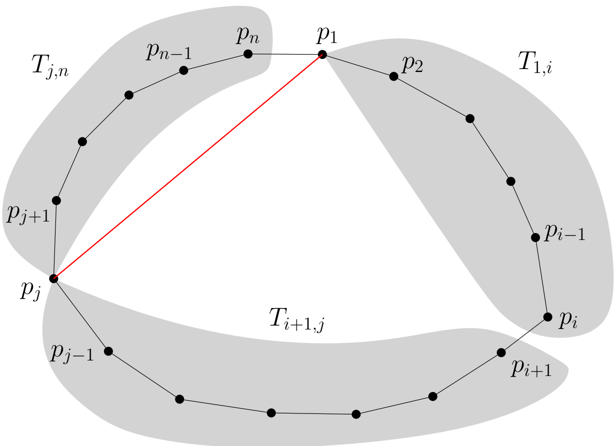

Figure 2 The convex polygon that is obtained from P. p_1 is connected to p_j in T.

Let T be a minimum Wiener index spanning tree of P and let W^* be its Wiener index, i.e., $W^* = W(T)$. Notice that, for any $1 \le i < j \le n$, the points in $P[i,j]$ are in convex position, since the points in P are in convex position. Since T is a spanning tree, each point, particularly p_1, is adjacent to at least one edge in T. Let p_j be the point with maximum index j that is connected to p_1 in T. Moreover, by Theorem 1, T is non-crossing. Thus, there exists an index i, with $1 \le i < j$, such that all of the points in $P[1,i]$ are closer to p_1 than to p_j in T, and all of the points in $P[i+1,j]$ are closer to p_j than to p_1 in T. Let $T_{1,i}$, $T_{i+1,j}$, and $T_{j,n}$ be the sub-trees of T containing the points in $P[1,i]$, $P[i+1,j]$, and $P[j,n]$, respectively; see Figure 2. Hence,

$$\begin{aligned}W^* &= W(T_{1,i}) + (n-i)\cdot\delta_1(T_{1,i})\\ &\quad + W(T_{i+1,j}) + (n-j+i)\cdot\delta_j(T_{i+1,j})\\ &\quad + W(T_{j,n}) + (j-1)\cdot\delta_j(T_{j,n})\\ &\quad + i(n-i)\cdot|p_1p_j|.\end{aligned}$$

For $1 \le i < j \le n$, let $W_j[i,j] = W(T_{i,j}) + (n-j+i-1) \cdot \delta_j(T_{i,j})$ be the minimum value obtained by a spanning tree $T_{i,j}$ of $P[i,j]$ rooted at p_j. Similarly, let $W_i[i,j] = W(T_{i,j}) + (n-j+i-1) \cdot \delta_i(T_{i,j})$ be the minimum value obtained by a spanning tree $T_{i,j}$ of $P[i,j]$ rooted at p_i. Thus, we can write W^* as

$$W^* = W_1[1,n] = W_1[1,i] + W_j[i+1,j] + W_j[j,n] + i(n-i) \cdot |p_1 p_j| \,.$$

Therefore, in order to compute W^*, we compute $W_1[1,i]$, $W_j[i+1,j]$, $W_j[j,n]$, and $i(n-i) \cdot |p_1 p_j|$ for each j between 2 and n and for each i between 1 and j, and take the minimum over the sum of these values. In general, for every $1 \le i < j \le n$, we compute $W_j[i,j]$ and $W_i[i,j]$ recursively using the following formulas; see also Figure 3.

$$W_j[i,j] = \min_{\substack{i \le k < j \\ k \le l < j}} \left\{ W_k[i,k] + W_k[k,l] + W_j[l+1,j] + (l-i+1)(n-l+i-1) \cdot |p_k p_j| \right\},$$

and

$$W_i[i,j] = \min_{\substack{i < k \le j \\ i \le l < k}} \left\{ W_i[i,l] + W_k[l+1,k] + W_k[k,j] + (j-l)(n-j+l) \cdot |p_i p_k| \right\}.$$

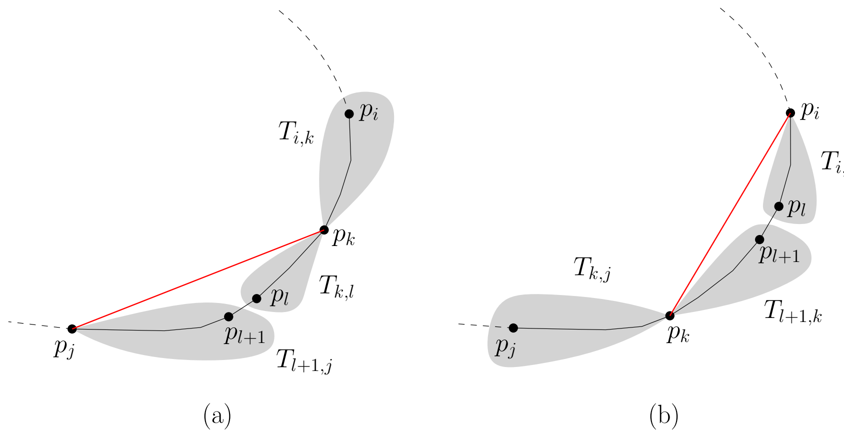

Figure 3 A sub-problem defined by $P[i,j]$. (a) Computing $W_j[i,j]$. (b) Computing $W_i[i,j]$.

We compute $W_j[i,j]$ and $W_i[i,j]$, for each $1 \le i < j \le n$, using dynamic programming as follows. We maintain two tables $\overrightarrow{M}$ and $\overleftarrow{M}$ each of size $n \times n$, such that $\overrightarrow{M}[i,j] = W_j[i,j]$ and $\overleftarrow{M}[i,j] = W_i[i,j]$, for each $1 \le i < j \le n$. We fill in the tables using Algorithm 1.

Notice that when we fill in the cell $\overrightarrow{M}[i,j]$, all of the cells $\overrightarrow{M}[i,k]$, $\overleftarrow{M}[k,l]$, and $\overrightarrow{M}[l+1,j]$, for each $i \le k < j$ and for each $k \le l < j$, are already computed, and when we fill in the cell $\overleftarrow{M}[i,j]$, all of the cells $\overleftarrow{M}[i,l]$, $\overrightarrow{M}[l+1,k]$, and $\overleftarrow{M}[k,j]$, for each $i < k \le j$ and for each $i \le l < k$, are already computed. Thus, each cell in the table is computed in $O(n^2)$ time, and the whole table is computed in $O(n^4)$ time. Therefore, $W^* = W_1[1,n] = \overleftarrow{M}[1,n]$ can be computed in $O(n^4)$ time.

The following theorem summarizes the result of this section.

▶ **Theorem 2.** *Let P be a set of n points in convex position. Then, a spanning tree of P of minimum Wiener index can be computed in $O(n^4)$ time.*

Algorithm 1 *ComputeOptimal*(P)

$n \leftarrow |P|$
for each $i \leftarrow 1$ to n **do**
  $\overrightarrow{M}[i,i] \leftarrow 0$
  $\overleftarrow{M}[i,i] \leftarrow 0$
for each $i \leftarrow n$ to 1 **do**
  for each $j \leftarrow i$ to n **do**
    $\overrightarrow{M}[i,j] \leftarrow \min\limits_{\substack{i \le k < j \\ k \le l < j}} \{ \overrightarrow{M}[i,k] + \overleftarrow{M}[k,l] + \overrightarrow{M}[l+1,j] + (l-i+1)(n-l+i-1) \cdot |p_k p_j| \}$
    $\overleftarrow{M}[i,j] \leftarrow \min\limits_{\substack{i < k \le j \\ i \le l < k}} \{ \overleftarrow{M}[i,l] + \overrightarrow{M}[l+1,k] + \overleftarrow{M}[k,j] + (j-l)(n-j+l) \cdot |p_i p_k| \}$
return $\overleftarrow{M}[1,n]$

4 Hardness proof

Let P be a set of points in the plane and let T be a spanning tree of P. Recall that $W(T) = \sum_{p,q \in P} \delta_T(p,q)$ denotes the Wiener index of T, where $\delta_T(p,q)$ is the length of the path between p and q in T. We define the weight of T to be $wt(T) = \sum_{(p,q) \in T} |pq|$, where $|pq|$ is the Euclidean distance between p and q. For an edge (p,q) in T, let $N_T(p,q) = |\{x \in P : \delta_T(p,x) < \delta_T(q,x)\}|$ be the number of points in T that are closer to p than to q. Then, $N_T(q,p)$ is the number of points in P that are closer to q than to p. The Wiener index $W(T)$ can be written as follows [32]:

$$W(T) = \sum_{(p,q) \in T} N_T(p,q) \cdot N_T(q,p) \cdot |pq|.$$

In this section, we prove that the following problem is NP-hard.

Euclidean Wiener Index Tree Problem: Given a set P of points in the Euclidean plane, a cost W, and a budget B, decide whether there exists a spanning tree T of P, such that $W(T) \le W$ and $wt(T) \le B$.

▶ **Theorem 3.** *The Euclidean Wiener Index Tree Problem is weakly NP-hard.*

Proof. We reduce from the Partition problem, which is known to be weakly NP-hard [14]. In the Partition problem, we are given a set $X = \{x_1, x_2, \dots, x_n\}$ of n positive integers with an even sum $R = \sum_{i=1}^{n} x_i$, and the goal is to decide whether there is a subset $S \subseteq X$, such that $\sum_{x_i \in S} x_i = \frac{1}{2}R$.

Given an instance $X = \{x_1, x_2, \dots, x_n\}$ of the Partition problem, with integer x_i's, we construct a set P of $m = n^3 + 3n + 1$ points in the plane as follows. The set P consists of a point s, n points $p_1, p_2, \dots, p_n$ located evenly spaced on a circle of radius nR centered at s, and a cluster C of n^3 points located evenly spaced on a circle of radius $\frac{1}{n^{10}R}$ centered at s; see Figure 4. Moreover, for each $1 \le i \le n$, we locate two points l_i and r_i, each at distance x_i from p_i, with the distance $|l_i r_i| = \frac{1}{2}x_i$. Finally, we set

$$\begin{aligned} B &= \left(n^2 + \frac{7}{4}\right)R + \frac{1}{n^7 R}, \text{ and} \\ W &= 3n^2(m-3)R + \left(\frac{9}{4}m - \frac{13}{4}\right)R + \frac{1}{n^4 R} + \frac{3}{n^6 R} \\ &= 3n^5 R + \frac{45}{4}n^3 R - 6n^2 R + \frac{27}{4}nR - R + \frac{1}{n^4 R} + \frac{3}{n^6 R}. \end{aligned}$$

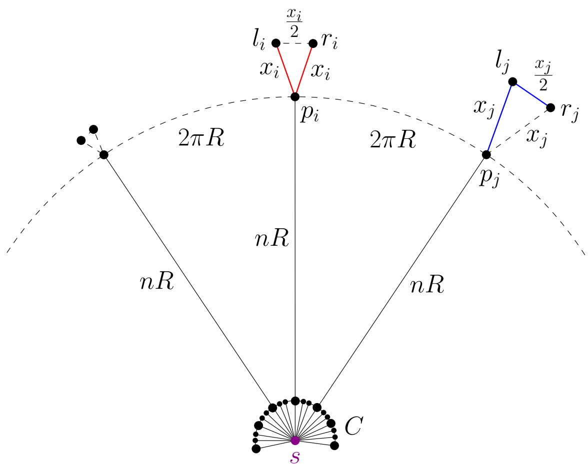

Figure 4 The set P constructed for the reduction. Connecting the points l_i, r_i, and p_i for $x_i \in S$ (red) and connecting the points l_j, r_j, and p_j for $x_j \in X \setminus S$ (blue).

We now prove the correctness of the reduction. Assume that there exists a set $S \subseteq X$, such that $\sum_{x_i \in S} x_i = \frac{1}{2}R$. We construct a spanning tree T for the points of P as follows. T consists of four sets of edges E_1, E_2, E_3, and E_4, such that

- $E_1 = \{(s, p_i) : 1 \leq i \leq n\}$,
- $E_2 = \{(p_i, l_i), (p_i, r_i) : x_i \in S\}$,
- $E_3 = \{(p_j, l_j), (l_j, r_j) : x_j \in X \setminus S\}$, and
- $E_4 = \{(s, p) : p \in C\}$; see Figure 4.

It is easy to see that

$$\begin{aligned} wt(T) &= wt(E_1) + wt(E_2) + wt(E_3) + wt(E_4) \\ &= n^2 R + R + \frac{3}{4}R + n^3 \cdot \frac{1}{n^{10}R} \\ &= \left(n^2 + \frac{7}{4}\right)R + \frac{1}{n^7 R} = B\,. \end{aligned}$$

Moreover, the Wiener index of T is

$$\begin{aligned} W(T) &= \sum_{(p,q)\in T} N_T(p,q) \cdot N_T(q,p) \cdot |pq| \\ &= \sum_{(p,q)\in E_1} N_T(p,q) \cdot N_T(q,p) \cdot |pq| + \sum_{(p,q)\in E_2} N_T(p,q) \cdot N_T(q,p) \cdot |pq| \\ &\quad + \sum_{(p,q)\in E_3} N_T(p,q) \cdot N_T(q,p) \cdot |pq| + \sum_{(p,q)\in E_4} N_T(p,q) \cdot N_T(q,p) \cdot |pq| \\ &= 3\big(n^3 + 3n - 2\big)n^2 R + 2\sum_{x_i \in S}\big(n^3 + 3n\big)x_i \\ &\quad + \sum_{x_i \notin S}\Big(\big(n^3 + 3n\big)\frac{1}{2}x_i + 2\big(n^3 + 3n - 1\big)x_i\Big) + n^3\big(n^3 + 3n\big) \cdot \frac{1}{n^{10}R} \\ &= 3n^2\big(n^3 + 3n - 2\big)R + \big(n^3 + 3n\big)R + \frac{1}{4}\big(n^3 + 3n\big)R + \big(n^3 + 3n - 1\big)R + \frac{1}{n^4 R} + \frac{3}{n^6 R} \\ &= 3n^5 R + \frac{45}{4}n^3 R - 6n^2 R + \frac{27}{4}nR - R + \frac{1}{n^4 R} + \frac{3}{n^6 R} = W\,. \end{aligned}$$

Conversely, let T be a spanning tree of P with $wt(T) \leq B$ and $W(T) \leq W$.

▷ Claim 4. There are exactly n edges in T between the points in $C \cup \{s\}$ and the others.

Proof. Assume there are k such edges. Since the weight of each such edge is at least $nR - \frac{1}{n^{10}R}$, we have $wt(T) \geq k(nR - \frac{1}{n^{10}R})$. On the other hand, $wt(T) \leq B = (n^2 + \frac{7}{4})R + \frac{1}{n^7 R}$. If $k > n$, then we have $wt(T) \geq n^2R + nR - \frac{n+1}{n^{10}R} > B$, for sufficiently large n. Moreover, consider the connections between the triplets $\{p_i, l_i, r_i\}$, for each $1 \leq i \leq n$, and the points in $C \cup \{s\}$. There are only k triples that are connected to the points in $C \cup \{s\}$ directly via an edge of length at least $nR - \frac{1}{n^{10}R}$, and each one of the other $n - k$ triplets is connected via an additional edge of length at least $2nR \cdot \cos\left(\frac{\pi}{n}\right)$. Thus,

$$\begin{aligned}
W(T) &> k\Big(nR - \frac{1}{n^{10}R}\Big) \cdot 3n^3 + (n-k)\Big(nR - \frac{1}{n^{10}R} + 2nR \cdot \cos\left(\frac{\pi}{n}\right)\Big) \cdot 3n^3 \\
&= \Big(3kn + 3n^2 - 3kn + (n-k) \cdot 2n \cdot \cos\left(\frac{\pi}{n}\right)\Big)n^3R - \frac{3k}{n^7R} - \frac{3(n-k)}{n^7R} \\
&= \Big(3n^2 + (n-k) \cdot 2n \cdot \cos\left(\frac{\pi}{n}\right)\Big)n^3R - \frac{3}{n^6R} \\
&= 3n^5R + (n-k) \cdot 2n^4R \cdot \cos\left(\frac{\pi}{n}\right) - \frac{3}{n^6R}\,.
\end{aligned}$$

And, if $k < n$, then we have $W(T) > 3n^5R + 2n^4R \cdot \cos\left(\frac{\pi}{n}\right) - \frac{3}{n^6R} > W$, for sufficiently large n. Therefore, we conclude that $k = n$. ◀

Let P_i denote the triplet $\{p_i, l_i, r_i\}$, for $1 \leq i \leq n$. From the proof of Claim 4, it follows that for every $1 \leq i \leq n$, there is exactly one edge (p, q) in T for which $p \in C \cup \{s\}$ and $q \in P_i$. Moreover, it is easy to see that $q = p_i$. Thus, in every P_i, we have $(p_i, l_i) \in T$ or $(p_i, r_i) \in T$. Assume, without loss of generality, that $(p_i, l_i) \in T$. Therefore, either $(p_i, r_i) \in T$ or $(l_i, r_i) \in T$. Let $S \subseteq X$, such that $x_i \in S$ if and only if $(p_i, r_i) \in T$, and let $R' = \sum_{x_i \in S} x_i$.

To complete the proof, we show that if $R' \neq \frac{1}{2}R$, then either $wt(T) > B$ or $W(T) > W$, which leads to a contradiction.

Case 1: $R' > \frac{1}{2}R$. In this case, we have

$$\begin{aligned}
wt(T) &= n^2R + \sum_{x_i \in S}\big(|p_il_i| + |p_ir_i|\big) + \sum_{x_i \notin S}\big(|p_il_i| + |l_ir_i|\big) + \sum_{(p,q) \in T_C} |pq| \\
&= n^2R + \sum_{x_i \in S} 2x_i + \sum_{x_i \notin S} \frac{3}{2}x_i + \sum_{(p,q) \in T_C} |pq| \\
&= n^2R + 2R' + \frac{3}{2}(R - R') + \sum_{(p,q) \in T_C} |pq| \\
&= n^2R + \frac{1}{2}R' + \frac{3}{2}R + \sum_{(p,q) \in T_C} |pq|\,.
\end{aligned}$$

Since $\sum_{(p,q) \in T_C} |pq| > 0$, we have

$$wt(T) > \; n^2R + \frac{1}{2}R' + \frac{3}{2}R.$$

Moreover, since $R' > \frac{1}{2}R$ and R is an integer, we have $R' \geq \frac{1}{2}R + \frac{1}{2}$. Thus,

$$\begin{aligned}
wt(T) &> n^2R + \frac{1}{2}\Big(\frac{1}{2}R + \frac{1}{2}\Big) + \frac{3}{2}R \\
&= \Big(n^2 + \frac{7}{4}\Big)R + \frac{1}{4} > B\,.
\end{aligned}$$

Therefore, $wt(T) > B$.

Case 2: $R' < \frac{1}{2}R$. In this case, we have

$$\begin{aligned}
W(T) = &\sum_{(p,q)\in T} N_T(p,q)\cdot N_T(q,p)\cdot |pq| \\
= &\,3\big(n^3+3n-2\big)n^2R + \sum_{x_i\in S}\big(n^3+3n\big)\big(|p_il_i|+|p_ir_i|\big) + \sum_{x_i\notin S}\big(n^3+3n\big)\cdot|l_ir_i| \\
&+ \sum_{x_i\notin S} 2\big(n^3+3n-1\big)\cdot|p_il_i| + \sum_{(p,q)\in T_C} N_T(p,q)\cdot N_T(q,p)\cdot|pq| \\
= &\,3\big(n^3+3n-2\big)n^2R + 2\big(n^3+3n\big)\sum_{x_i\in S} x_i + \frac{1}{2}\big(n^3+3n\big)\sum_{x_i\notin S} x_i \\
&+ 2\big(n^3+3n-1\big)\sum_{x_i\notin S} x_i + \sum_{(p,q)\in T_C} N_T(p,q)\cdot N_T(q,p)\cdot|pq| \\
= &\,3\big(n^3+3n-2\big)n^2R + 2\big(n^3+3n\big)R' + \frac{1}{2}\big(n^3+3n\big)\big(R-R'\big) \\
&+ 2\big(n^3+3n-1\big)\big(R-R'\big) + \sum_{(p,q)\in T_C} N_T(p,q)\cdot N_T(q,p)\cdot|pq| \\
= &\,3\big(n^3+3n-2\big)n^2R + 2\big(n^3+3n\big)R' + \frac{1}{2}\big(n^3+3n\big)R - \frac{1}{2}\big(n^3+3n\big)R' \\
&+ 2\big(n^3+3n-1\big)R - 2\big(n^3+3n-1\big)R' + \sum_{(p,q)\in T_C} N_T(p,q)\cdot N_T(q,p)\cdot|pq| \\
= &\,3n^5R + 9n^3R - 6n^2R - \left(\frac{1}{2}\Big(n^3+3n\Big) - 2\right)R' + \frac{5}{2}\big(n^3+3n\big)R - 2R \\
&+ \sum_{(p,q)\in T_C} N_T(p,q)\cdot N_T(q,p)\cdot|pq|\,.
\end{aligned}$$

Since $\sum_{(p,q)\in T_C} N_T(p,q)\cdot N_T(q,p)\cdot|pq| > 0$, we have

$$W(T) > \; 3n^5R + 9n^3R - 6n^2R - \left(\frac{1}{2}\Big(n^3+3n\Big) - 2\right)R' + \frac{5}{2}\big(n^3+3n\big)R - 2R.$$

Moreover, since $R' < \frac{1}{2}R$ and R is an integer, we have $R' \le \frac{1}{2}R - \frac{1}{2}$. Thus,

$$\begin{aligned}
W(T) &> 3n^5R + 9n^3R - 6n^2R - \left(\frac{1}{2}\Big(n^3+3n\Big) - 2\right)\left(\frac{1}{2}R - \frac{1}{2}\right) + \frac{5}{2}\big(n^3+3n\big)R - 2R \\
&= 3n^5R + \frac{45}{4}n^3R - 6n^2R + \frac{27}{4}nR - R + \frac{1}{4}n^3 + \frac{3}{4}n - 1 > W\,.
\end{aligned}$$

Therefore, $W(T) > W$. ◀

5 Paths that optimize Wiener index

Let P be a set of n points in the plane. In this section, we consider the problem of computing a *Hamiltonian path* on the point set P that minimizes the Wiener index. We first prove that, unlike spanning trees, an optimal Hamiltonian path may have crossing edges.

▶ **Theorem 5.** *A Hamiltonian path on a point set P that minimizes the Wiener index is not necessarily non-crossing.*

Proof. Consider the set P of $n = 2m + 2$ points in convex position as shown in Figure 5. The set P consists of four parts: two points p and q, and two clusters of points, P_l and P_r, where $|P_l| = |P_r| = m$. The points in cluster P_l (resp., P_r) are located on a convex arc of length $\frac{1}{m^3}$ that passes through the origin $(0, 0)$ (resp., through coordinate $(6, 0)$). The points p and q are located at coordinates $(5, 1)$ and $(5, -1)$, respectively.

Figure 5 A set P of $n = 2m + 2$ points in convex position.

Observe first that any Hamiltonian path of P that seeks to minimize the Wiener index will connect the points and clusters (p, q, P_l, and P_r) while using as few connections between the different clusters (P_l and P_r) as possible. To see this, recall that the contribution to the Wiener index of edge (u, v) in a Hamiltonian path Π of P is $N_\Pi(u, v) \cdot N_\Pi(v, u) \cdot |uv|$. Thus, the contribution to the Wiener index of an edge (u, v) in a Hamiltonian path Π of P connecting two points from different clusters is at least $(2m + 1) \cdot \sqrt{2} = \Omega(m)$, while the contribution of an edge (u, v) connecting two points inside a cluster is at most $(m + 1)(m + 1) \cdot \frac{1}{m^3} = O(\frac{1}{m})$. Therefore, any Hamiltonian path of P that seeks to minimize the Wiener index will connect the points in P_l by a path, and will connect the points in P_r by a path.

Let Π_l (resp., Π_r) be a Hamiltonian path connecting the points in P_l (resp., P_r). Thus, the contribution of Π_l (or of Π_r) to the Wiener index of any Hamiltonian path of P is at least $(m-1)(2m+1) \cdot \frac{1}{m^3} = \Omega\big(\frac{1}{m}\big)$, and at most $(m-1)(m+1)(m+1) \cdot \frac{1}{m^3} = O(1)$. Therefore, it is sufficient to consider the 12 possible Hamiltonian paths defined on the points $(0, 0)$, $(6, 0)$, p, and q while treating each one of the points $(0, 0)$ and $(6, 0)$ as an approximation of a path (of Wiener index at most $O(1)$ and at least $\Omega\big(\frac{1}{m}\big)$) containing m points, starting and ending at this point. There are six types of path structures, and within each type there are two symmetric paths. We compute their Wiener indices as follows; see Figure 6.

Type A:

$$\begin{aligned} W(\Pi_A) &\le 2(2m+1) + \sqrt{2} \cdot 2 \cdot 2m + 6m(m+2) + O(1) \\ &= 4m + 2 + 4\sqrt{2}m + 6m^2 + 12m + O(1) \\ &= 6m^2 + 4\big(4 + \sqrt{2}\big)m + O(1)\,. \end{aligned}$$

Type B:

$$\begin{aligned} W(\Pi_B) &> 2(2m+1) + \sqrt{26} \cdot 2 \cdot 2m + 6m(m+2) \\ &= 4m + 2 + 4\sqrt{26}m + 6m^2 + 12m \\ &= 6m^2 + 4\big(4 + \sqrt{26}\big)m + \Theta(1)\,. \end{aligned}$$

Type C:

$$\begin{aligned} W(\Pi_C) &> \sqrt{2}(2m+1) + \sqrt{2}(m+1)(m+1) + \sqrt{26}m(m+2) \\ &= 2\sqrt{2}m + \sqrt{2} + \sqrt{2}m^2 + 2\sqrt{2}m + \sqrt{2} + \sqrt{26}m^2 + 2\sqrt{26}m \\ &= \left(\sqrt{26} + \sqrt{2}\right)m^2 + \Theta(m)\,. \end{aligned}$$

Type D:

$$\begin{aligned} W(\Pi_D) &> \sqrt{2}m(m+2) + \sqrt{26}(m+1)(m+1) + \sqrt{26}(2m+1) \\ &= \sqrt{2}m^2 + 2\sqrt{2}m + \sqrt{26}m^2 + 2\sqrt{26}m + \sqrt{26} + 2\sqrt{26}m + \sqrt{26} \\ &= \left(\sqrt{26} + \sqrt{2}\right)m^2 + \Theta(m)\,. \end{aligned}$$

Type E:

$$\begin{aligned} W(\Pi_E) &> \sqrt{2}(2m+1) + 6(m+1)(m+1) + \sqrt{26}(2m+1) \\ &= 2\sqrt{2}m + \sqrt{2} + 6m^2 + 12m + 6 + 2\sqrt{26}m + \sqrt{26} \\ &= 6m^2 + 2\left(6 + \sqrt{26} + \sqrt{2}\right)m + \Theta(1)\,. \end{aligned}$$

Type F:

$$\begin{aligned} W(\Pi_F) &> \sqrt{2}m(m+2) + 2(m+1)(m+1) + \sqrt{26}m(m+2) \\ &= \sqrt{2}m^2 + 2\sqrt{2}m + 2m^2 + 4m + 2 + \sqrt{26}m^2 + 2\sqrt{26}m \\ &= \left(2 + \sqrt{26} + \sqrt{2}\right)m^2 + \Theta(m)\,. \end{aligned}$$

This computation shows that the Type A Hamiltonian paths, which are self-crossing, have the smallest Wiener index (for sufficiently large n).

◀

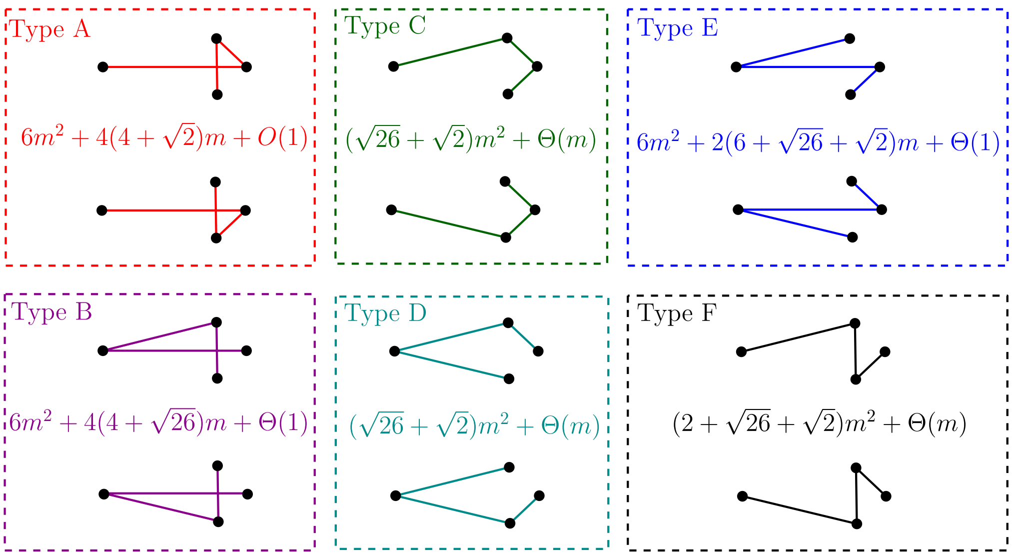

Figure 6 The 12 possible Hamiltonian paths that are defined on points $(0,0)$, $(6,0)$, p, and q. For each Type we show the resulting Wiener index of the two paths.

In the following two theorems, we prove that the problem of computing a Hamiltonian path on P that minimizes the Wiener index is NP-hard and that the Wiener index of any

Hamiltonian path on P can be $\Omega(\sqrt{n})$ times the Wiener index of the complete Euclidean graph over P.

▶ **Theorem 6.** *It is NP-hard to compute a Hamiltonian path on a point set P that minimizes the Wiener index.*

Proof. We reduce from Hamiltonicity in a grid graph, which is a graph whose vertices are integer grid points and whose edges join pairs of grid points at distance 1 from each other; see Figure 7). It is well known (and readily computed) that the Wiener index of a Hamiltonian path of n points, where each edge is of length one, is $\binom{n+1}{3}$ (see [21]). Thus, it is easy to see that a grid graph $G = (P, E)$ has a Hamiltonian path if and only if there exists a Hamiltonian path in the complete graph over P of Wiener index $\binom{n+1}{3}$. ◀

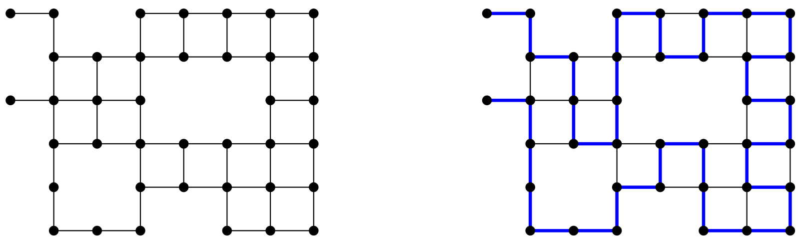

Figure 7 A grid graph G and a Hamiltonian path with Wiener index $\binom{n+1}{3}$ in G.

▶ **Theorem 7.** *There exists a set P of n points in the plane such that the Wiener index of any Hamiltonian path on P is $\Omega(\sqrt{n})$ times the Wiener index of the complete Euclidean graph over P.*

Proof. Let P be a set of n points located on a $\sqrt{n} \times \sqrt{n}$ integer grid. The Wiener index of any Hamiltonian path of P is at least $\binom{n+1}{3}$, which is the Wiener index of a Hamiltonian path whose $(n-1)$ edges are each of length 1. Thus, the Wiener index of any Hamiltonian path of P is $\Theta(n^3)$. On the other hand, since the distance between any two points of P is at most $\sqrt{2n}$, and there are $\binom{n}{2}$ pairs of points, the Wiener index of the complete graph over P is $O(n^{2.5})$. Therefore, the Wiener index of any Hamiltonian path of P is $\Omega(\sqrt{n})$ times the Wiener index of the complete Euclidean graph over P. ◀

6 Concluding remarks

In this paper, we study the problem of computing a spanning tree of a set P of n points in the plane that minimizes the Wiener index when the underlying graph is the complete Euclidean graph. We proved that any spanning tree of P that minimizes the Wiener index has non-crossing edges. Then, we used this fact to devise an $O(n^4)$-time algorithm that constructs a spanning tree of minimum Wiener index when the points of P are in convex position. Improving the running time of the algorithm is an interesting question.

Another challenging open question is to determine the complexity of computing a spanning tree of minimum Wiener index for a set P of n points in general position: Is this problem NP-hard? In this paper, we considered a variant of this problem when the total weight of the tree is bounded, and we proved that this variant is NP-hard.

References

1. M. Abrahamsen and V. Fredslund-Hansen. Degree of convexity and expected distances in polygons. *arXiv preprint arXiv:2208.07106*, 2022.
2. D. Bonchev. The Wiener number: Some applications and new developments. In *Topology in Chemistry*, pages 58–88, 2002.
3. D. Bonchev and N. Trinajstić. Information theory, distance matrix, and molecular branching. *The Journal of Chemical Physics*, 67(10):4517–4533, 1977.
4. P. Bose, P. Carmi, and L. Chaitman-Yerushalmi. On bounded degree plane strong geometric spanners. *Journal of Discrete Algorithms*, 15:16–31, 2012.
5. P. Bose, J. Gudmundsson, and M. H. M. Smid. Constructing plane spanners of bounded degree and low weight. In *ESA*, pages 234–246, 2002.
6. P. Bose, D. Hill, and M. H. M. Smid. Improved spanning ratio for low degree plane spanners. *Algorithmica*, 80(3):935–976, 2018.
7. J. Cardinal, S. Collette, and S. Langerman. Local properties of geometric graphs. *Computational Geometry*, 39(1):55–64, 2008.
8. P. Carmi and L. Chaitman-Yerushalmi. Minimum weight Euclidean t-spanner is np-hard. *Journal of Discrete Algorithms*, 22:30–42, 2013.
9. S.-W. Cheng, C. Knauer, S. Langerman, and M. H. M. Smid. Approximating the average stretch factor of geometric graphs. *Journal of Computational Geometry*, 3(1):132–153, 2012.
10. K. Das, I. Gutman, and B. Furtula. Survey on geometric-arithmetic indices of graphs. *Match (Mulheim an der Ruhr, Germany)*, 65:595–644, 01 2011.
11. K. Dhamdhere, A. Gupta, and R. Ravi. Approximation algorithms for minimizing average distortion. *Theory of Computing Systems*, 39(1):93–111, 2006.
12. A. A. Dobrynin, R. Entringer, and I. Gutman. Wiener index of trees: Theory and applications. *Acta Applicandae Mathematica*, 66(3):211–249, 2001.
13. A. Filtser and S. Solomon. The greedy spanner is existentially optimal. In *Proceedings of the 2016 ACM Symposium on Principles of Distributed Computing (PODC)*, pages 9–17, 2016.
14. M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., USA, 1979.
15. T. F. Gonzalez. *Handbook of Approximation Algorithms and Metaheuristics, Chapter 59*. Chapman & Hall/CRC, 2007.
16. A. Graovac and T. Pisanski. On the wiener index of a graph. *Journal of Mathematical Chemistry*, 8:53–62, 1991.
17. F. Harary. *Graph Theory*. Addison-Wesley, 1969.
18. T. C. Hu. Optimum communication spanning trees. *SIAM Journal on Computing*, 3(3):188–195, 1974.
19. D. S. Johnson, J. K. Lenstra, and A. H. G. Rinnooy Kan. The complexity of the network design problem. *Networks*, 8:279–285, 1978.
20. L. Kier. *Molecular connectivity in chemistry and drug research*. Elsevier, 1976.
21. M. Knor, R. Škrekovski, and A. Tepeh. Mathematical aspects of Wiener index. *Ars Mathematica Contemporanea*, 11:327–352, 2015.
22. X. Y. Li and Y. Wang. Efficient construction of low weighted bounded degree planar spanner. *International Journal of Computational Geometry & Applications*, 14(1-2):69–84, 2004.
23. O. Mekenyan, D. Bonchev, and N. Trinajstić. Structural complexity and molecular properties of cyclic systems with acyclic branches. *Croatica Chemica Acta*, 56(2):237–261, 1983.
24. J. S. B. Mitchell and W. Mulzer. Proximity algorithms. In Jacob E. Goodman Csaba Tóth, Joseph O'Rourke, editor, *Handbook of Discrete and Computational Geometry (3rd Edition)*, chapter 32, pages 849–874. Chapman & Hall/CRC, Boca Raton, FL, 2017.
25. B. Mohar and T. Pisanski. How to compute the wiener index of a graph. *Journal of Mathematical Chemistry*, 2(3):267, 1988.
26. G. Narasimhan and M. Smid. *Geometric Spanner Networks*. Cambridge University Press, 2007.

27 S. Ronghua. The average distance of trees. *Journal of Systems Science and Complexity*, 6:18–24, 1993.

28 R. Sitters. Polynomial time approximation schemes for the traveling repairman and other minimum latency problems. *SIAM Journal on Computing*, 50(5):1580–1602, 2021.

29 N. Trinajstić. *Mathematical and Computational Concepts in Chemistry*. Ellis Horwood, Chichester, 1986.

30 E. Weiszfeld and F. Plastria. On the point for which the sum of the distances to n given points is minimum. *Annals of Operations Research*, 167(1):7–41, 2009.

31 H. Wiener. Structural determination of paraffin boiling points. *Journal of the American Chemical Society*, 69(1):17–20, 1947.

32 B. Ye Wu, K.-M. Chao, and C. Yi Tang. Approximation algorithms for some optimum communication spanning tree problems. *Discrete Applied Mathematics*, 102(3):245–266, 2000.

33 B. Ye Wu, G. Lancia, V. Bafna, K.-M. Chao, R. Ravi, and C. Y. Tang. A polynomial-time approximation scheme for minimum routing cost spanning trees. *SIAM Journal on Computing*, 29(3):761–778, 2000.

34 K. Xu, M. Liu, K. Das, I. Gutman, and B. Furtula. A survey on graphs extremal with respect to distance-based topological indices. *Match (Mulheim an der Ruhr, Germany)*, 71:461–508, 02 2014.