Geometric Spanning Trees Minimizing the Wiener Index

Karim Abu-Affash[1] Paz Carmi[2] Ori Luwisch[2]
Joseph S. B. Mitchell[3]

[1]Software Engineering Department, Shamoon College of Engineering, Israel
[2]Department of Computer Science, Ben-Gurion University of Negev, Israel
[3]Department of Applied Mathematics and Statistics, Stony Brook University, USA

The 18th Algorithms and Data Structures Symposium
(WADS 2023)

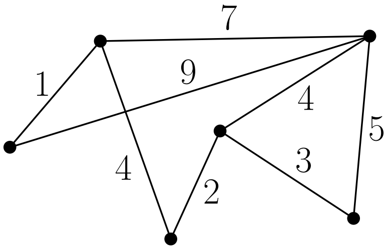
July 31–August 2, 2023

Outline

Outline

Wiener Index in Graphs

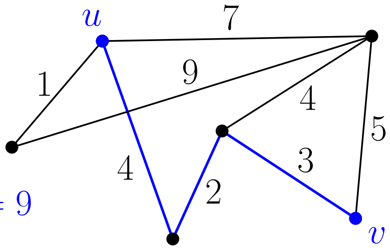
- Let $G = (V, E)$ be a wieghted undirected graph.



Wiener Index in Graphs

- Let $G = (V, E)$ be a wieghted undirected graph.
- Let $\delta_G(u, v)$ denote the **shortest (minimum-wieght) path** between the vertices u and v in G.

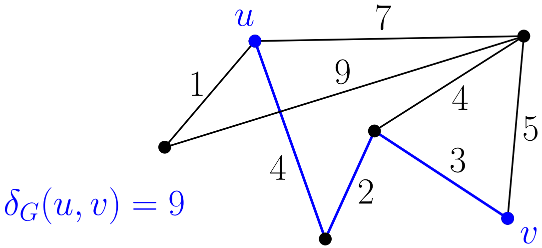
$$\delta_G(u, v) = 9$$



Wiener Index in Graphs

- Let $G = (V, E)$ be a wieghted undirected graph.
- Let $\delta_G(u, v)$ denote the **shortest (minimum-wieght) path** between the vertices u and v in G.
- The **Wiener index** of G, $W(G)$, is defined as the sum of the shortest paths between every pair of vertices in G, i.e.,

$$W(G) = \sum_{u,v \in V} \delta_G(u, v)$$



Outline

Motivation and Related Works

In Chemistry:

- The Wiener index was first introduced by the chemist Haryy Wiener in 1947 to correlate between the boiling point (and later other chemical properties) and the molecule structure.

Motivation and Related Works

In Chemistry:

- The Wiener index was first introduced by the chemist Haryy Wiener in 1947 to correlate between the boiling point (and later other chemical properties) and the molecule structure.
- Molecules structure can be modeled as an undirected graph: each vertex represents an atom and each edge represents a bond between two atoms.

Motivation and Related Works

In Chemistry:

- The Wiener index was first introduced by the chemist Haryy Wiener in 1947 to correlate between the boiling point (and later other chemical properties) and the molecule structure.
- Molecules structure can be modeled as an undirected graph: each vertex represents an atom and each edge represents a bond between two atoms.
- The Wiener index of molecular graphs can be used to predict properties of the corresponding molecules.

Motivation and Related Works

In Chemistry:

- The Wiener index was first introduced by the chemist Haryy Wiener in 1947 to correlate between the boiling point (and later other chemical properties) and the molecule structure.
- Molecules structure can be modeled as an undirected graph: each vertex represents an atom and each edge represents a bond between two atoms.
- The Wiener index of molecular graphs can be used to predict properties of the corresponding molecules.

Most of works related to Wiener index focus on computing and bounding the Wiener index of specific graphs or classes of graphs.

Motivation and Related Works

In Network Design: Given an undirected graph $G = (V, E)$ and a (non-negative) weight function (representing the delay on each edge), the **routing cost** $c(T)$ of a spanning tree T of G is

$$c(T) = \sum_{u,v \in V} \delta_T(u, v)$$

Motivation and Related Works

In Network Design: Given an undirected graph $G = (V, E)$ and a (non-negative) weight function (representing the delay on each edge), the **routing cost** $c(T)$ of a spanning tree T of G is

$$c(T) = \sum_{u,v \in V} \delta_T(u, v)$$

The Minimum Routing Cost Spanning Tree (MRCST) problem

Given a weighted undirected graph $G = (V, E)$, compute a minimum routing cost spanning tree of G.

Motivation and Related Works

In Network Design: Given an undirected graph $G = (V, E)$ and a (non-negative) weight function (representing the delay on each edge), the **routing cost** $c(T)$ of a spanning tree T of G is

$$c(T) = \sum_{u,v \in V} \delta_T(u, v)$$

The Minimum Routing Cost Spanning Tree (MRCST) problem

Given a weighted undirected graph $G = (V, E)$, compute a minimum routing cost spanning tree of G.

- **MRCST** is NP-complete, even if all edge weights are 1 [Johnson et al. 1978].

Motivation and Related Works

In Network Design: Given an undirected graph $G = (V, E)$ and a (non-negative) weight function (representing the delay on each edge), the **routing cost** $c(T)$ of a spanning tree T of G is

$$c(T) = \sum_{u,v \in V} \delta_T(u, v)$$

The Minimum Routing Cost Spanning Tree (MRCST) problem

Given a weighted undirected graph $G = (V, E)$, compute a minimum routing cost spanning tree of G.

- **MRCST** is NP-complete, even if all edge weights are 1 [Johnson et al. 1978].
- There exists a PTAS for **MRCST** [Wu et al. 2000].

Outline

Our Contribution

Problem: Given a set P of n points in the plane, compute a spanning tree on P that minimizes the Wiener index (the edge weights are their Euclidean lengths).

Our Contribution

Problem: Given a set P of n points in the plane, compute a spanning tree on P that minimizes the Wiener index (the edge weights are their Euclidean lengths).

Our results: We show that

1. The spanning tree of P that minimizes the Wiener index is planar.

Our Contribution

Problem: Given a set P of n points in the plane, compute a spanning tree on P that minimizes the Wiener index (the edge weights are their Euclidean lengths).

Our results: We show that

1. The spanning tree of P that minimizes the Wiener index is planar.
2. One can solve the problem in polynomial time when the points of P are in convex position.

Our Contribution

Problem: Given a set P of n points in the plane, compute a spanning tree on P that minimizes the Wiener index (the edge weights are their Euclidean lengths).

Our results: We show that

1. The spanning tree of P that minimizes the Wiener index is planar.
2. One can solve the problem in polynomial time when the points of P are in convex position.
3. Given a cost W and a budget B, computing a spanning tree of P whose Wiener index is at most W and its weight is at most B is (weakly) NP-hard.

Our Contribution

Problem: Given a set P of n points in the plane, compute a spanning tree on P that minimizes the Wiener index (the edge weights are their Euclidean lengths).

Our results: We show that

1. The spanning tree of P that minimizes the Wiener index is planar.
2. One can solve the problem in polynomial time when the points of P are in convex position.
3. Given a cost W and a budget B, computing a spanning tree of P whose Wiener index is at most W and its weight is at most B is (weakly) NP-hard.
4. The Hamiltonian path of P that minimizes the Wiener index is not necessarily planar.

Our Contribution

Problem: Given a set P of n points in the plane, compute a spanning tree on P that minimizes the Wiener index (the edge weights are their Euclidean lengths).

Our results: We show that

1. The spanning tree of P that minimizes the Wiener index is planar.
2. One can solve the problem in polynomial time when the points of P are in convex position.
3. Given a cost W and a budget B, computing a spanning tree of P whose Wiener index is at most W and its weight is at most B is (weakly) NP-hard.
4. The Hamiltonian path of P that minimizes the Wiener index is not necessarily planar.
5. Computing a Hamiltonian path of P that minimizes the Wiener index is NP-hard.

Outline

Optimal Tree is Planar

Let P be a set of n points in the plane and let T be a tree that minimizes the Wiener index.

Lemma 1

T is planar.

Optimal Tree is Planar

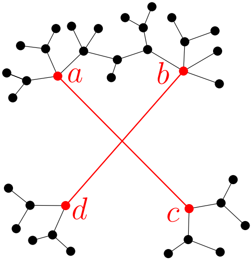
Let P be a set of n points in the plane and let T be a tree that minimizes the Wiener index.

Lemma 1

T is planar.

Proof:

- Assume towards a contradiction that there are two crossing edges (a, c) and (b, d) in T.



Optimal Tree is Planar

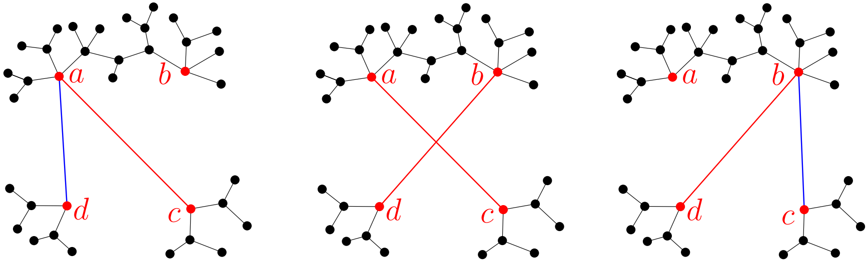
Let P be a set of n points in the plane and let T be a tree that minimizes the Wiener index.

Lemma 1

T is planar.

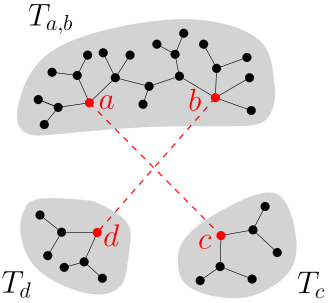
Proof:

- Assume towards a contradiction that there are two crossing edges (a, c) and (b, d) in T.



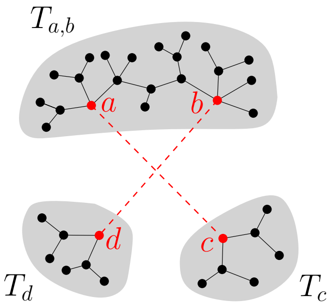
Optimal Tree is Planar (Proof of Lemma 1)

- Let T_{ab}, T_c, and T_d be the sub-trees obtained by removing the edges (a, c) and (b, d) from T.



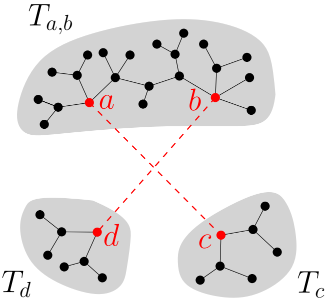
Optimal Tree is Planar (Proof of Lemma 1)

- Let T_{ab}, T_c, and T_d be the sub-trees obtained by removing the edges (a, c) and (b, d) from T.
- Let n_{ab}, n_c, and n_d be the number of points in T_{ab}, T_c, and T_d, respectively.



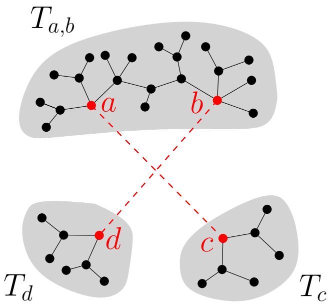
Optimal Tree is Planar (Proof of Lemma 1)

- Let T_{ab}, T_c, and T_d be the sub-trees obtained by removing the edges (a, c) and (b, d) from T.
- Let n_{ab}, n_c, and n_d be the number of points in T_{ab}, T_c, and T_d, respectively.
- Let $\delta_a(T_{ab}) = \sum_{p \in T_{ab}} \delta_T(a, p)$ denote the total weight of the paths from a to every point in T_{ab}



Optimal Tree is Planar (Proof of Lemma 1)

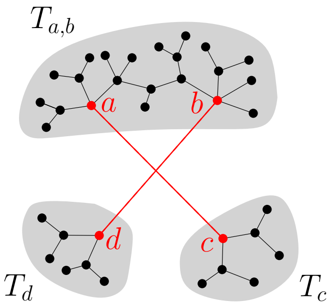
- Let T_{ab}, T_c, and T_d be the sub-trees obtained by removing the edges (a, c) and (b, d) from T.
- Let n_{ab}, n_c, and n_d be the number of points in T_{ab}, T_c, and T_d, respectively.
- Let $\delta_a(T_{ab}) = \sum_{p \in T_{ab}} \delta_T(a, p)$ denote the total weight of the paths from a to every point in T_{ab} (Similarly, $\delta_b(T_{ab})$, $\delta_c(T_c)$, $\delta_d(T_d)$).



Optimal Tree is Planar (Proof of Lemma 1)

Thus,

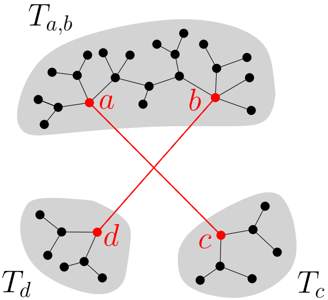
$$W(T) = W(T_{ab}) + n_c \cdot \delta_a(T_{ab}) + n_d \cdot \delta_b(T_{ab})$$



Optimal Tree is Planar (Proof of Lemma 1)

Thus,

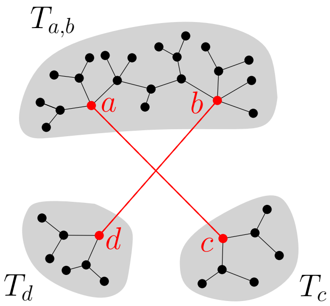
$$W(T) = W(T_{ab}) + n_c \cdot \delta_a(T_{ab}) + n_d \cdot \delta_b(T_{ab}) + W(T_c) + (n_{ab} + n_d) \cdot \delta_c(T_c)$$



Optimal Tree is Planar (Proof of Lemma 1)

Thus,

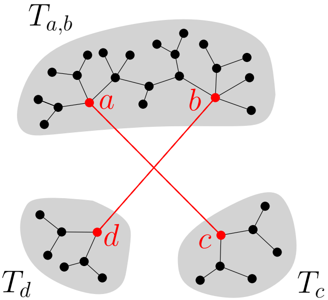
$$
\begin{aligned}
W(T) = {} & W(T_{ab}) + n_c \cdot \delta_a(T_{ab}) + n_d \cdot \delta_b(T_{ab}) \\
& + W(T_c) + (n_{ab} + n_d) \cdot \delta_c(T_c) \\
& + W(T_d) + (n_{ab} + n_c) \cdot \delta_d(T_d)
\end{aligned}
$$



Optimal Tree is Planar (Proof of Lemma 1)

Thus,

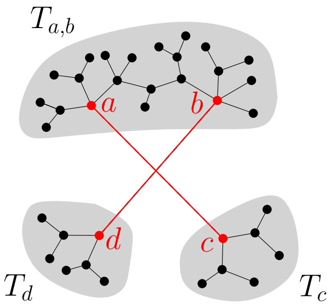
$$\begin{aligned} W(T) = {} & W(T_{ab}) + n_c \cdot \delta_a(T_{ab}) + n_d \cdot \delta_b(T_{ab}) \\ & + W(T_c) + (n_{ab} + n_d) \cdot \delta_c(T_c) \\ & + W(T_d) + (n_{ab} + n_c) \cdot \delta_d(T_d) \\ & + n_c(n_{ab} + n_d) \cdot |ac| \end{aligned}$$



Optimal Tree is Planar (Proof of Lemma 1)

Thus,

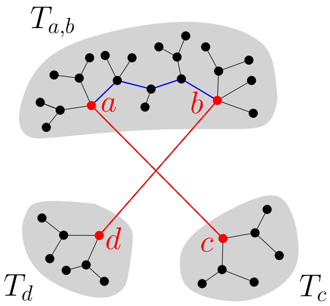
$$W(T) = W(T_{ab}) + n_c \cdot \delta_a(T_{ab}) + n_d \cdot \delta_b(T_{ab})$$
$$+ W(T_c) + (n_{ab} + n_d) \cdot \delta_c(T_c)$$
$$+ W(T_d) + (n_{ab} + n_c) \cdot \delta_d(T_d)$$
$$+ n_c(n_{ab} + n_d) \cdot |ac| + n_d(n_{ab} + n_c) \cdot |bd|$$



Optimal Tree is Planar (Proof of Lemma 1)

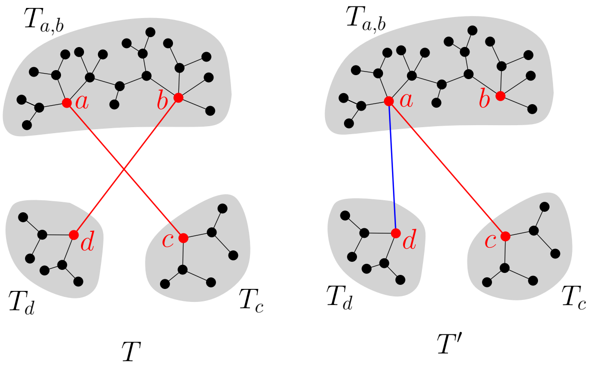
Thus,

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



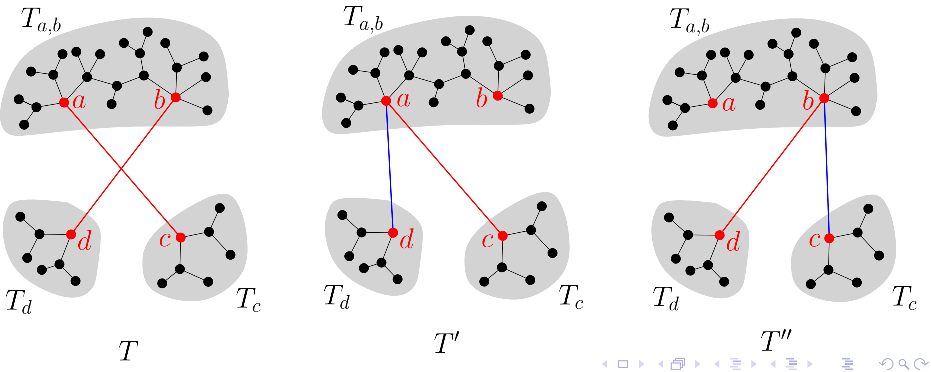
Optimal Tree is Planar (Proof of Lemma 1)

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



Optimal Tree is Planar (Proof of Lemma 1)

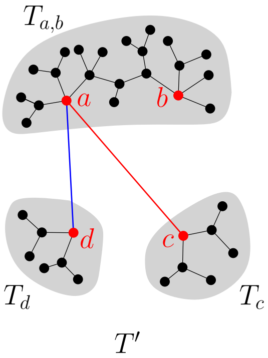
- Let T' be the spanning tree of P obtained from T by replacing the edge (b, d) by the edge (a, d).
- Let T'' be the spanning tree of P obtained from T by replacing the edge (a, c) by the edge (b, c).



Optimal Tree is Planar (Proof of Lemma 1)

Thus,

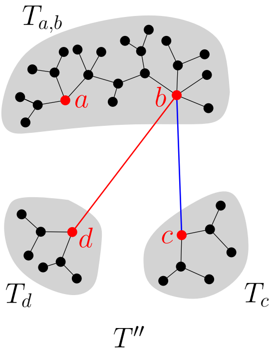
$$\begin{aligned} W(T') = {} & W(T_{ab}) + (n_c + n_d) \cdot \delta_a(T_{ab}) \\ & + W(T_c) + (n_{ab} + n_d) \cdot \delta_c(T_c) + n_c(n_{ab} + n_d) \cdot |ac| \\ & + W(T_d) + (n_{ab} + n_c) \cdot \delta_d(T_d) + n_d(n_{ab} + n_c) \cdot |ad| \end{aligned}$$



Optimal Tree is Planar (Proof of Lemma 1)

and

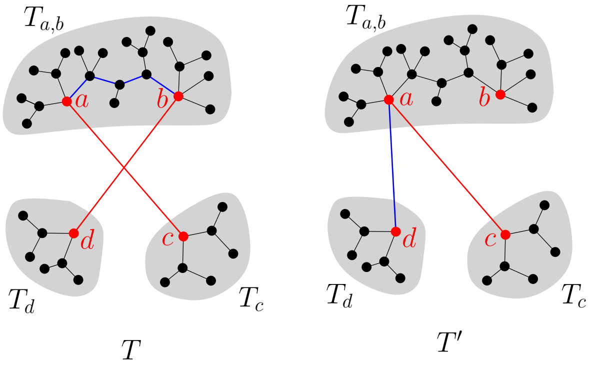
$$\begin{aligned} W(T'') = {} & W(T_{ab}) + (n_c + n_d) \cdot \delta_b(T_{ab}) \\ & + W(T_c) + (n_{ab} + n_d) \cdot \delta_c(T_c) + n_c(n_{ab} + n_d) \cdot |bc| \\ & + W(T_d) + (n_{ab} + n_c) \cdot \delta_d(T_d) + n_d(n_{ab} + n_c) \cdot |bd| \end{aligned}$$



Optimal Tree is Planar (Proof of Lemma 1)

Therefore,

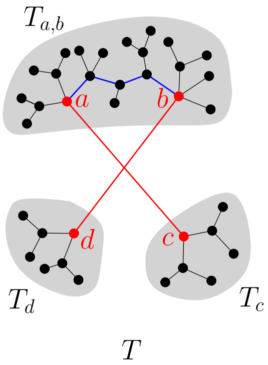
$$W(T) - W(T') = n_d\big(\delta_b(T_{ab}) - \delta_a(T_{ab})\big) + n_d(n_{ab} + n_c)\big(|bd| - |ad|\big) + n_c \cdot n_d \cdot \delta_T(a, b)$$

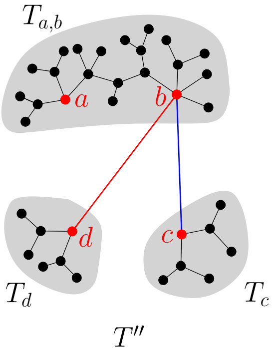


Optimal Tree is Planar (Proof of Lemma 1)

and

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





Optimal Tree is Planar (Proof of Lemma 1)

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

Optimal Tree is Planar (Proof of Lemma 1)

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

$$W(T) - W(T') = n_d\big(\delta_b(T_{ab}) - \delta_a(T_{ab})\big) + n_d(n_{ab} + n_c)\big(|bd| - |ad|\big) + n_c \cdot n_d \cdot \delta_T(a, b) \leq 0$$

and

$$W(T) - W(T'') = n_c\big(\delta_a(T_{ab}) - \delta_b(T_{ab})\big) + n_c(n_{ab} + n_d)\big(|ac| - |bc|\big) + n_c \cdot n_d \cdot \delta_T(a, b) \leq 0$$

Optimal Tree is Planar (Proof of Lemma 1)

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

$$W(T) - W(T') = n_d\big(\delta_b(T_{ab}) - \delta_a(T_{ab})\big) + n_d(n_{ab} + n_c)\big(|bd| - |ad|\big) + n_c \cdot n_d \cdot \delta_T(a, b) \leq 0$$

and

$$W(T) - W(T'') = n_c\big(\delta_a(T_{ab}) - \delta_b(T_{ab})\big) + n_c(n_{ab} + n_d)\big(|ac| - |bc|\big) + n_c \cdot n_d \cdot \delta_T(a, b) \leq 0$$

- Since $n_c > 0$ and $n_d > 0$, we have

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

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

Optimal Tree is Planar (Proof of Lemma 1)

- By summing the two inequalities,

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

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

we have

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

Optimal Tree is Planar (Proof of Lemma 1)

- By summing the two inequalities,

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

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

  we have

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

- That is,

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

Optimal Tree is Planar (Proof of Lemma 1)

- By summing the two inequalities,

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

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

  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,

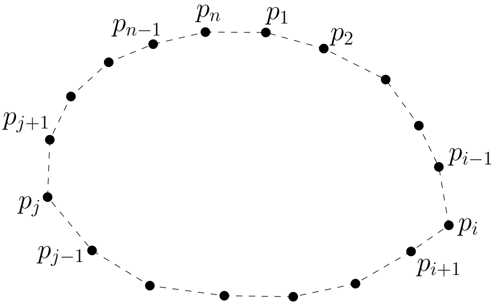
  $$n_{ab}\big(|bd| + |ac| - |ad| - |bc|\big) + n_c\big(|bd| + \delta_T(a,b) - |ad|\big) + n_d\big(|ac| + \delta_T(a,b) - |bc|\big) \le 0$$

- 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. ■

Outline

Optimal Tree of Points in Convex Position

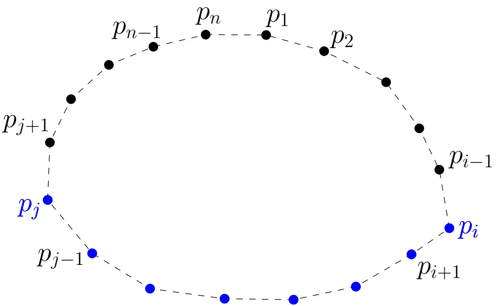
Let $P = \{p_1, p_2, \ldots, p_n\}$ be a set of n points in convex position:



Optimal Tree of Points in Convex Position

Let $P = \{p_1, p_2, \ldots, p_n\}$ be a set of n points in convex position:

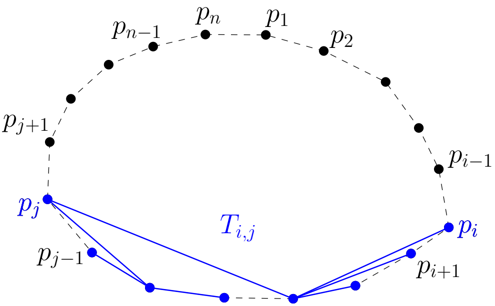
- For each $1 \leq i \leq j \leq n$, let $P[i, j] \subseteq P$ be the set $\{p_i, p_{i+1}, \ldots, p_j\}$.



Optimal Tree of Points in Convex Position

Let $P = \{p_1, p_2, \ldots, p_n\}$ be a set of n points in convex position:

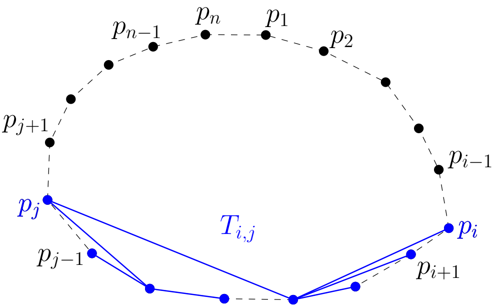
- For each $1 \leq i \leq j \leq n$, let $P[i, j] \subseteq P$ be the set $\{p_i, p_{i+1}, \ldots, p_j\}$.
- Let $T_{i,j}$ be a spanning tree of $P[i, j]$, and let $W(T_{i,j})$ denote its Wiener index.



Optimal Tree of Points in Convex Position

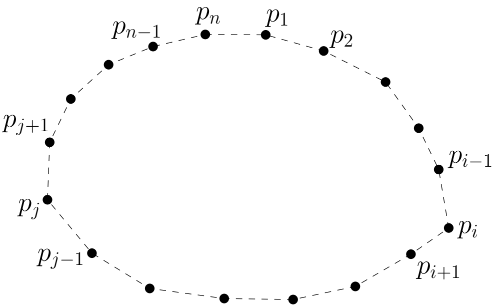
Let $P = \{p_1, p_2, \ldots, p_n\}$ be a set of n points in convex position:

- For each $1 \leq i \leq j \leq n$, 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.
- Let $\delta_i(T_{i,j})$ be the total weight of the paths from p_i to every point of $P[i,j]$ in $T_{i,j}$ (Similarly, $\delta_j(T_{i,j})$).



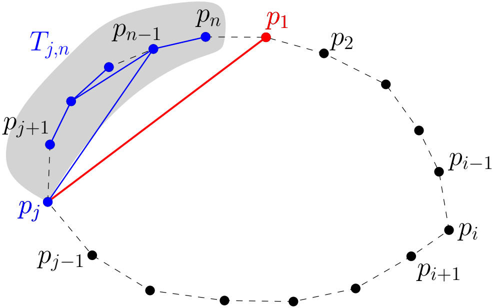
Optimal Tree of Points in Convex Position

- Let T be a (planar) minimum Wiener index spanning tree of P and let $W^* = W(T)$.



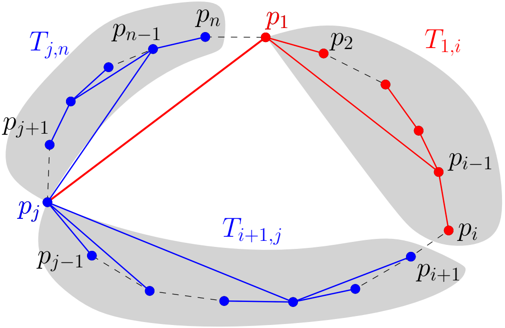
Optimal Tree of Points in Convex Position

- Let T be a (planar) minimum Wiener index spanning tree of P and let $W^* = W(T)$.
- Let p_j be the point with maximum j that is connected to p_1 in T.



Optimal Tree of Points in Convex Position

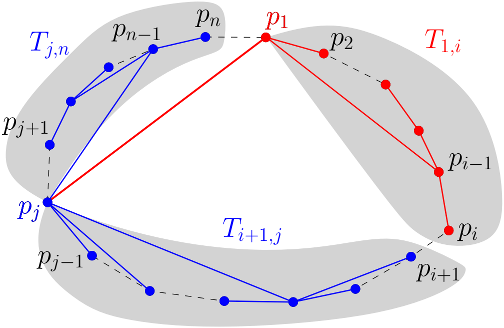
- Let T be a (planar) minimum Wiener index spanning tree of P and let $W^* = W(T)$.
- Let p_j be the point with maximum j that is connected to p_1 in T.
- Moreover, there exists an index $1 \le i < j$ such that all the points in $P[1, i]$ are closer to p_1 than to p_j in T, and all the points in $P[i + 1, j]$ are closer to p_j than to p_1 in T.



Optimal Tree of Points in Convex Position

Hence,

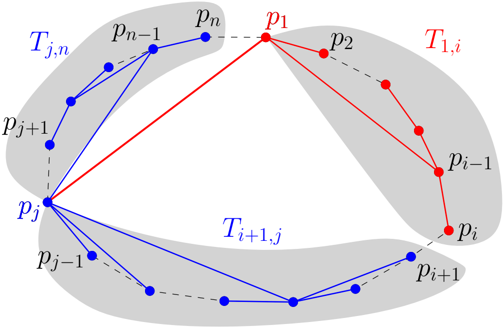
$$W^* = W(T_{1,i}) + (n-i) \cdot \delta_1(T_{1,i})$$



Optimal Tree of Points in Convex Position

Hence,

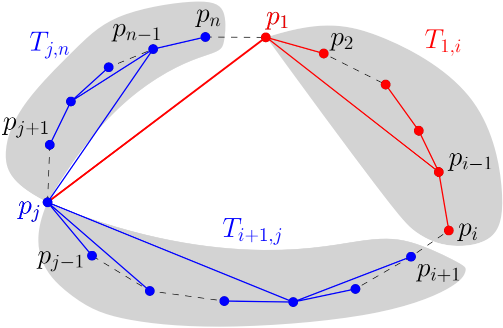
$$W^* = W(T_{1,i}) + (n-i) \cdot \delta_1(T_{1,i}) + W(T_{i+1,j}) + (n-j+i) \cdot \delta_j(T_{i+1,j})$$



Optimal Tree of Points in Convex Position

Hence,

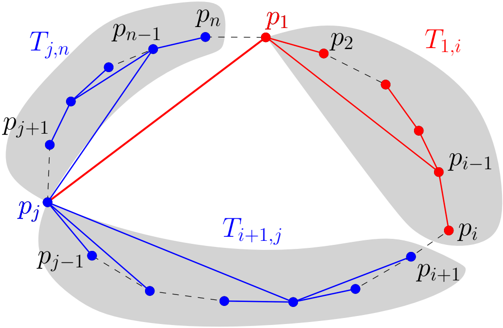
$$W^* = W(T_{1,i}) + (n-i)\cdot\delta_1(T_{1,i}) + W(T_{i+1,j}) + (n-j+i)\cdot\delta_j(T_{i+1,j}) + W(T_{j,n}) + (j-1)\cdot\delta_j(T_{j,n})$$



Optimal Tree of Points in Convex Position

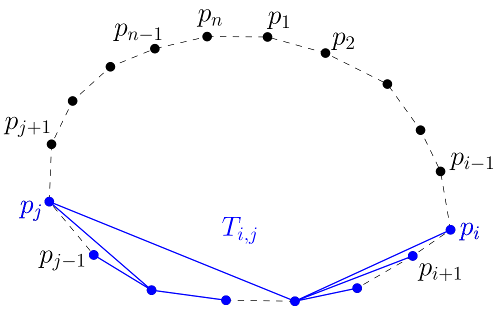
Hence,

$$\begin{aligned} W^* = {} & W(T_{1,i}) + (n-i) \cdot \delta_1(T_{1,i}) \\ & + W(T_{i+1,j}) + (n-j+i) \cdot \delta_j(T_{i+1,j}) \\ & + W(T_{j,n}) + (j-1) \cdot \delta_j(T_{j,n}) \\ & + i(n-i) \cdot |p_1 p_j| \end{aligned}$$



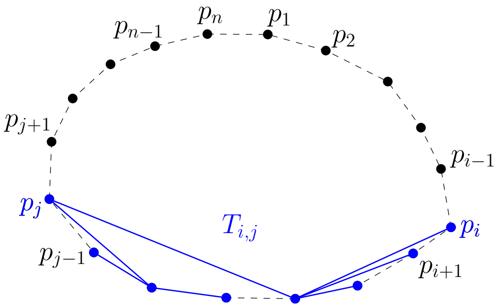
Optimal Tree of Points in Convex Position

- 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.



Optimal Tree of Points in Convex Position

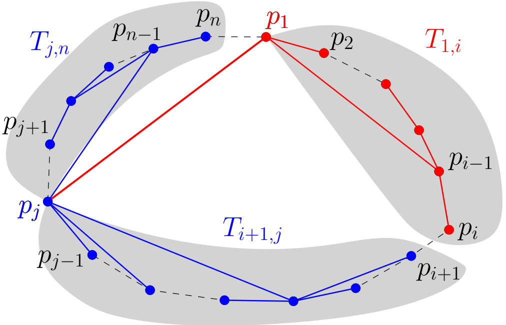
- Let $W_j[i,j] = W(T_{i,j}) + (n - j + i - 1) \cdot \delta_j(T_{i,j})$ be the minimum value obtained by a spanning tree $T_{i,j}$ of $P[i,j]$ rooted at p_j.
- Let $W_i[i,j] = W(T_{i,j}) + (n - j + i - 1) \cdot \delta_i(T_{i,j})$ be the minimum value obtained by a spanning tree $T_{i,j}$ of $P[i,j]$ rooted at p_i.



Optimal Tree of Points in Convex Position

- 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.
- 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

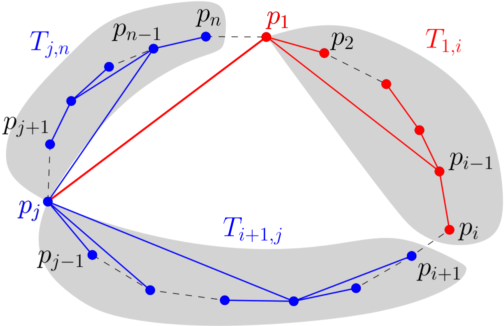
$$W^* = W_1[1,n] = W_1[1,i] + W_j[i+1,j] + W_j[j,n] + i(n-i) \cdot |p_1 p_j|$$



Optimal Tree of Points in Convex Position

Therefore, $W_1[1,n]$ can be recursively computed using

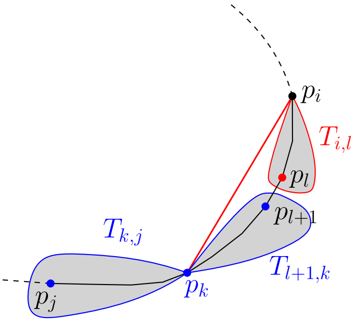
$$W_1[1,n] = \min_{\substack{1<j\leq n \\ 1\leq i<j}} \left\{ W_1[1,i] + W_j[i+1,j] + W_j[j,n] + i(n-i)\cdot|p_1p_j| \right\}$$



Optimal Tree of Points in Convex Position

Sub-problems: For every $1 \leq i < j \leq n$, we recursively compute:

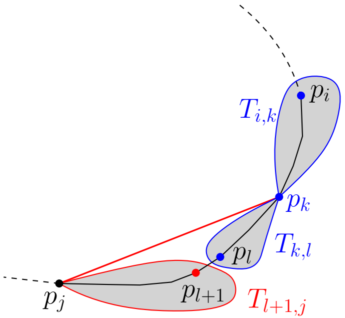
$$W_i[i,j] = \min_{\substack{i<k\leq j \\ i\leq l<k}} \left\{ W_i[i,l] + W_k[l+1,k] + W_k[k,j] + (j-l)(n-j+l)\cdot|p_i p_k| \right\}$$



Sub-problems: For every $1 \leq i < j \leq n$, we recursively compute:

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

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

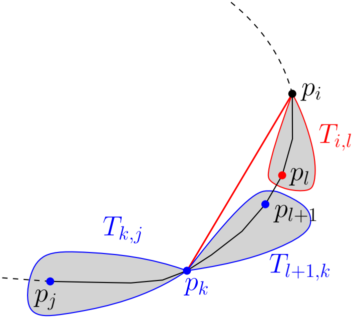


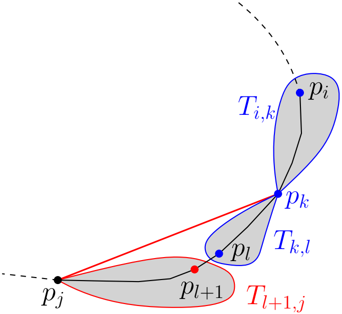
Optimal Tree of Points in Convex Position

Sub-problems: For every $1 \leq i < j \leq n$, we recursively compute:

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

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





Optimal Tree of Points in Convex Position

Dynamic peogramming algorithm: We maintain two tables $\overleftarrow{M}$ and $\overrightarrow{M}$ each of size $n \times n$, such that $\overleftarrow{M}[i, j] = W_i[i, j]$ and $\overrightarrow{M}[i, j] = W_j[i, j]$, for each $1 \leq i < j \leq n$.

Optimal Tree of Points in Convex Position

Dynamic peogramming algorithm: We maintain two tables $\overleftarrow{M}$ and $\overrightarrow{M}$ each of size $n \times n$, such that $\overleftarrow{M}[i,j] = W_i[i,j]$ and $\overrightarrow{M}[i,j] = W_j[i,j]$, for each $1 \leq i < j \leq n$.

Algorithm 2 *ComputeOptimal*(P)

1: **for** each $i \leftarrow 1$ to n **do**

  $\overleftarrow{M}[i,i] \leftarrow 0 \quad , \quad \overrightarrow{M}[i,i] \leftarrow 0$

2: **for** each $i \leftarrow n$ to 1 **do**

  for each $j \leftarrow i$ to n **do**

$$\overleftarrow{M}[i,j] \leftarrow \min_{\substack{i<k\leq j \\ i\leq l<k}} \left\{ \overleftarrow{M}[i,l] + \overrightarrow{M}[l+1,k] + \overleftarrow{M}[k,j] + (j-l)(n-j+l) \cdot |p_i p_k| \right\}$$

$$\overrightarrow{M}[i,j] \leftarrow \min_{\substack{i\leq k<j \\ k\leq l<j}} \left\{ \overrightarrow{M}[i,k] + \overleftarrow{M}[k,l] + \overrightarrow{M}[l+1,j] + (l-i+1)(n-l+i-1) \cdot |p_k p_j| \right\}$$

3: **return** $\overleftarrow{M}[1,n]$

Optimal Tree of Points in Convex Position

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.

Hardness Proof

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

$$W(T) = \sum_{p,q \in P} \delta_T(p,q) \leq W \text{ (the Wiener index of } T\text{), and}$$

$$wt(T) = \sum_{(p,q) \in T} |pq| \leq B \text{ (the weight of } T\text{).}$$

Hardness Proof

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

$$W(T) = \sum_{p,q \in P} \delta_T(p,q) \leq W \quad \text{(the Wiener index of } T\text{), and}$$

$$wt(T) = \sum_{(p,q) \in T} |pq| \leq B \quad \text{(the weight of } T\text{).}$$

Theorem 2

The Euclidean Wiener Index Tree Problem is weakly NP-hard.

Hardness Proof

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

$$W(T) = \sum_{p,q \in P} \delta_T(p,q) \leq W \quad \text{(the Wiener index of } T\text{), and}$$

$$wt(T) = \sum_{(p,q) \in T} |pq| \leq B \quad \text{(the weight of } T\text{).}$$

Theorem 2

The Euclidean Wiener Index Tree Problem is weakly NP-hard.

Proof (sketch): We reduce from the Partition problem.

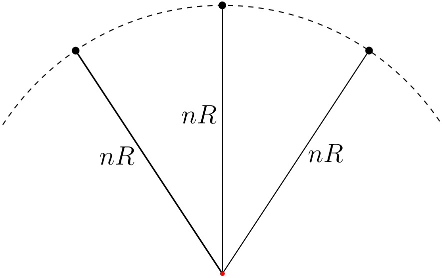
Partition: Given a set $X = \{x_1, x_2, \ldots, x_n\}$ of n positive integers with even $R = \sum_{i=1}^{n} x_i$, decide whether there is a subset $S \subseteq X$, such that $\sum_{x_i \in S} x_i = R/2$.

Hardness Proof

- Given an instance $X = \{x_1, x_2, \ldots, x_n\}$ of Partition, we construct a set P of $m = n^3 + 3n$ points as follows:

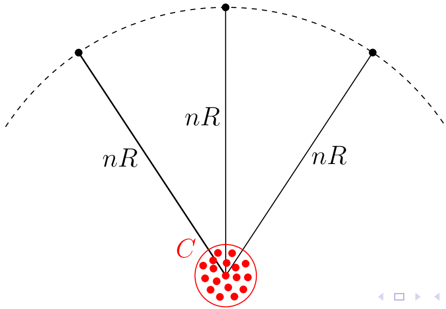
Hardness Proof

- Given an instance $X = \{x_1, x_2, \ldots, x_n\}$ of Partition, we construct a set P of $m = n^3 + 3n$ points as follows:
- Locate n points $p_1, \ldots, p_n$ equally spaced on a circle of radius nR.



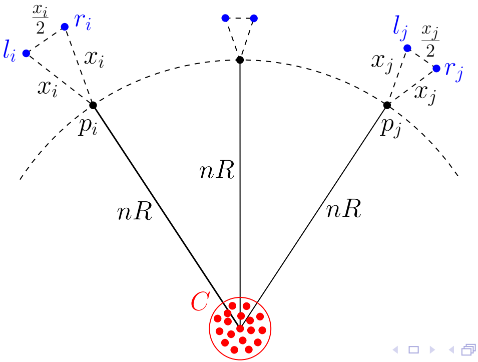
Hardness Proof

- Given an instance $X = \{x_1, x_2, \ldots, x_n\}$ of Partition, we construct a set P of $m = n^3 + 3n$ points as follows:
- Locate n points $p_1, \ldots, p_n$ equally spaced on a circle of radius nR.
- Locate a cluster C of n^3 points on the center of the circle.



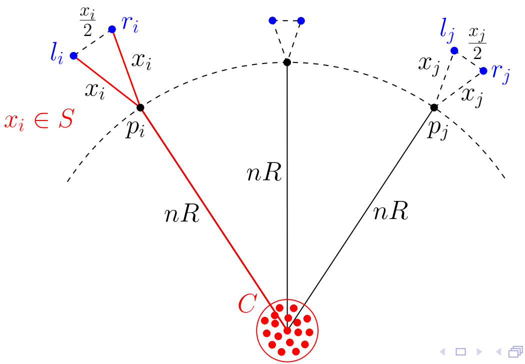
Hardness Proof

- Given an instance $X = \{x_1, x_2, \ldots, x_n\}$ of Partition, we construct a set P of $m = n^3 + 3n$ points as follows:
- Locate n points $p_1, \ldots, p_n$ equally spaced on a circle of radius nR.
- Locate a cluster C of n^3 points on the center of the circle.
- For each $1 \leq i \leq n$, locate two points l_i and r_i both at distance x_i from p_i and the distance between them is $\frac{1}{2}x_i$.



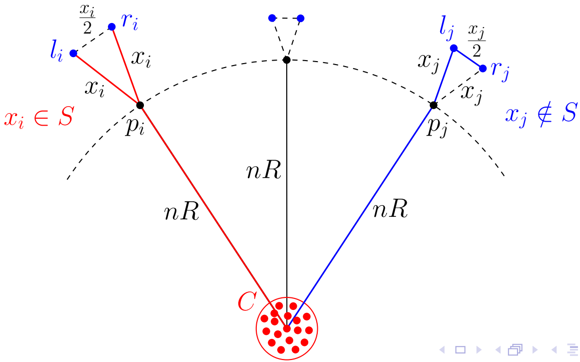
Hardness Proof

- Given an instance $X = \{x_1, x_2, \ldots, x_n\}$ of Partition, we construct a set P of $m = n^3 + 3n$ points as follows:
- Locate n points $p_1, \ldots, p_n$ equally spaced on a circle of radius nR.
- Locate a cluster C of n^3 points on the center of the circle.
- For each $1 \leq i \leq n$, locate two points l_i and r_i both at distance x_i from p_i and the distance between them is $\frac{1}{2}x_i$.



Hardness Proof

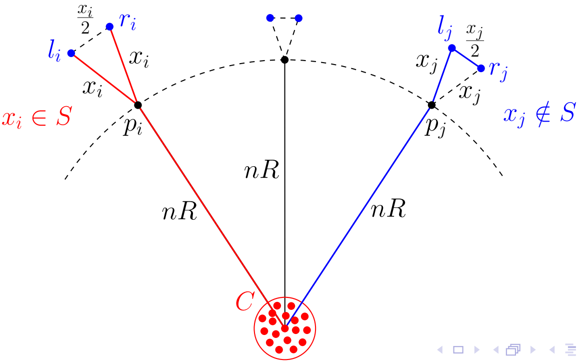
- Given an instance $X = \{x_1, x_2, \ldots, x_n\}$ of Partition, we construct a set P of $m = n^3 + 3n$ points as follows:
- Locate n points $p_1, \ldots, p_n$ equally spaced on a circle of radius nR.
- Locate a cluster C of n^3 points on the center of the circle.
- For each $1 \leq i \leq n$, locate two points l_i and r_i both at distance x_i from p_i and the distance between them is $\frac{1}{2}x_i$.



Hardness Proof

Finally, set $$B = n^2R + R + \frac{3}{4}R = \left(n^2 + \frac{7}{4}\right)R, \text{ and}$$

$$\begin{aligned} W &= 3n^2(m-3)R + \left(\frac{9}{4}m - \frac{13}{4}\right)R \\ &= 3n^5R + \frac{45}{4}n^3R - 9n^2R + \frac{27}{4}nR - \frac{13}{4}R \end{aligned}$$



Wiener Index Paths

Let P be a set of n points.

Theorem 4

The path that minimizes the Wiener index among all Hamiltonian paths of P is not necessarily planar.

Wiener Index Paths

Let P be a set of n points.

Theorem 4

The path that minimizes the Wiener index among all Hamiltonian paths of P is not necessarily planar.

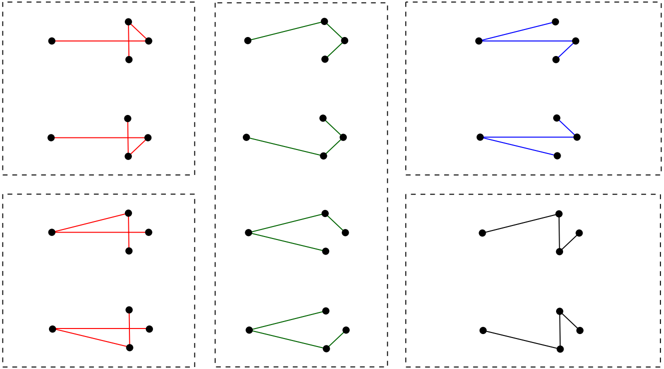
Proof: Consider the set P of $n = 2m + 2$ points located as follows.

Wiener Index Paths

- Since the points in P_l are arbitrarily close to the origin $(0,0)$, any path connecting these points has a Wiener index zero (Similarly for the points in P_r).

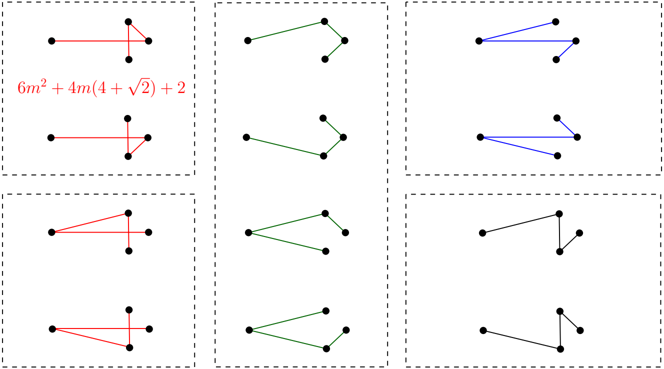
Wiener Index Paths

- Since the points in P_l are arbitrarily close to the origin $(0, 0)$, any path connecting these points has a Wiener index zero (Similarly for the points in P_r).
- Therefore, it is sufficient to consider the 12 possible Hamiltonian paths defined on points $(0, 0)$, $(6, 0)$, p, and q.



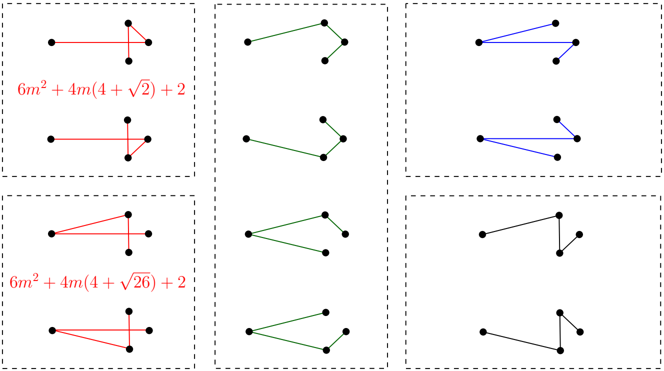
Wiener Index Paths

- Since the points in P_l are arbitrarily close to the origin $(0, 0)$, any path connecting these points has a Wiener index zero (Similarly for the points in P_r).
- Therefore, it is sufficient to consider the 12 possible Hamiltonian paths defined on points $(0, 0)$, $(6, 0)$, p, and q.



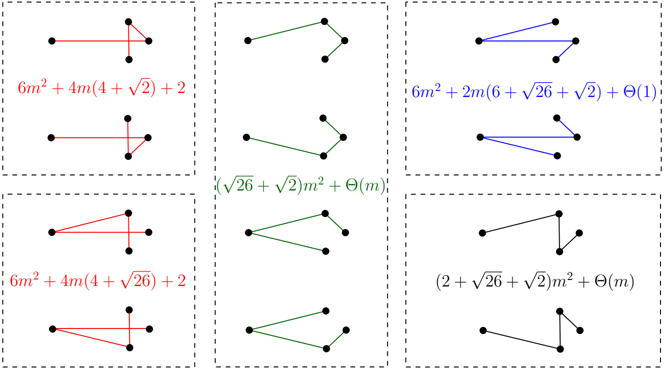
Wiener Index Paths

- Since the points in P_l are arbitrarily close to the origin $(0, 0)$, any path connecting these points has a Wiener index zero (Similarly for the points in P_r).
- Therefore, it is sufficient to consider the 12 possible Hamiltonian paths defined on points $(0, 0)$, $(6, 0)$, p, and q.



Wiener Index Paths

- Since the points in P_l are arbitrarily close to the origin $(0, 0)$, any path connecting these points has a Wiener index zero (Similarly for the points in P_r).
- Therefore, it is sufficient to consider the 12 possible Hamiltonian paths defined on points $(0, 0)$, $(6, 0)$, p, and q.



Wiener Index Paths

Theorem 5

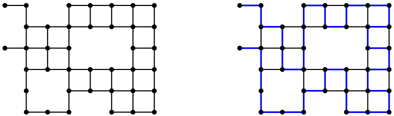
For points in the Euclidean plane, it is NP-hard to compute a Hamiltonian path minimizing the Wiener index.

Wiener Index Paths

Theorem 5

For points in the Euclidean plane, it is NP-hard to compute a Hamiltonian path minimizing the Wiener index.

Proof: We reduce from Hamiltonicity in a grid graph (whose vertices are integer grid points and whose edges join pairs of grid points at distance one).



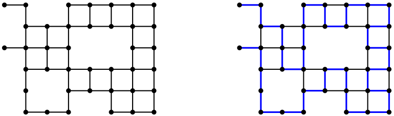
Wiener Index Paths

Theorem 5

For points in the Euclidean plane, it is NP-hard to compute a Hamiltonian path minimizing the Wiener index.

Proof: We reduce from Hamiltonicity in a grid graph (whose vertices are integer grid points and whose edges join pairs of grid points at distance one).

- It is well known that the Wiener index of a Hamiltonian path of n points, where each edge is of length one, is $\binom{n+1}{3}$.



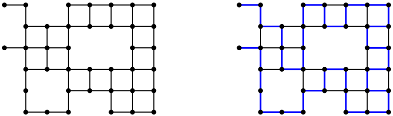
Wiener Index Paths

Theorem 5

For points in the Euclidean plane, it is NP-hard to compute a Hamiltonian path minimizing the Wiener index.

Proof: We reduce from Hamiltonicity in a grid graph (whose vertices are integer grid points and whose edges join pairs of grid points at distance one).

- It is well known that the Wiener index of a Hamiltonian path of n points, where each edge is of length one, is $\binom{n+1}{3}$.
- 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 $\binom{n+1}{3}$. ∎



Summary

Given a set P of points in the plane, we showed that

1. The spanning tree of P that minimizes the Wiener index is planar.
2. One can solve the problem in polynomial time when the points of P are in convex position.
3. Given a cost W and a budget B, computing a spanning tree of P whose Wiener index is at most W and its weight is at most B is (weakly) NP-hard.
4. The Hamiltonian path of P that minimizes the Wiener index is not necessarily planar.
5. Computing a Hamiltonian path of P that minimizes the Wiener index is NP-hard.

Thank you

Questions?