A $\frac{13}{9}$-approximation of the average-$\frac{2\pi}{3}$-MST

Ahmad Biniaz[1] Prosenjit Bose[2] Patrick Devaney[3]

Canadian Conference on Computational Geometry 2022

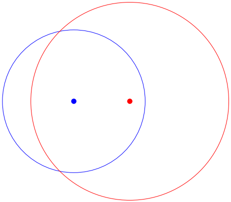
[1]School of Computer Science, University of Windsor, ahmad.biniaz@gmail.com
[2]School of Computer Science, Carleton University, jit@scs.carleton.ca
[3]School of Computer Science, University of Windsor, devaneyp@uwindsor.ca

Table of Contents

Introduction

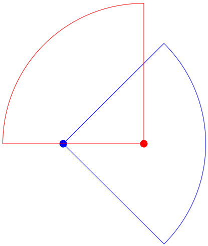
A wireless communication network can be represented as a geometric graph in the plane.

- Each antenna is represented by a point p
- Transmission range is represented by a disk with radius r centered at p
- Edge between two points if they are within each other's transmission ranges



Introduction – Motivation

- Replacing omni-directional antennas with directional antennas.
- Directional antennas provide several advantages over omni-directional antennas
    - less potential for interference
    - lower power consumption
    - reduced area where communications could be maliciously intercepted



α-Spanning Tree

Motivated by this problem, Aschner and Katz introduced the *α-Spanning Tree* (α-ST).

Definition: α-ST (Aschner and Katz, 2017)

A spanning tree of the complete Euclidean graph in the plane where all incident edges of each point p lie in a wedge of angle α with apex p.

Average-α-Minimum Spanning Tree

In [6], Biniaz et al. extended this concept to an average-α-minimum spanning tree ($\overline{\alpha}$-MST).

Definition: $\overline{\alpha}$-MST (Biniaz et al., 2022)

An α-MST with the relaxed restriction that the average angle of all the wedges is at most α

- $A\left(\frac{2\pi}{3}\right)$: the smallest ratio of the length of the $\overline{\frac{2\pi}{3}}$-MST to the length of the standard MST.
- In [6]: $\frac{4}{3} \leq A\left(\frac{2\pi}{3}\right) \leq \frac{3}{2}$
- We show $A\left(\frac{2\pi}{3}\right) \leq \frac{13}{9}$

Notation

- *Maximal path*: a path with at least two edges where all internal vertex degrees are 2, and the end vertex degrees are not 2.
- *Charge*: The angle that the incident edges of a vertex in an $\overline{\alpha}$-MST are allowed to fall within.
  - Can be redistributed between vertices.

Outline

Algorithm of [6] for the $\overline{\frac{2\pi}{3}}$-MST

- Starts with a standard MST.
- Considers the MST with all maximal paths contracted.
- Introduces edge shortcuts in each contracted path.

By exploiting additional geometric properties we save some charge from path vertices. The saved charges allow us to introduce fewer shortcuts than the original algorithm, resulting in a shorter $\overline{\frac{2\pi}{3}}$-ST.

Table of Contents

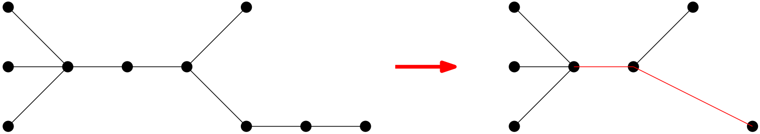
The Algorithm of [6]

We refer to the algorithm of [6] as "Algorithm 1".

- Starts by computing a degree-5 minimum spanning tree T of the point set
- Each vertex holds a charge of $\frac{2\pi}{3}$
- Two phases that redistribute the charges
  - Also modify the tree

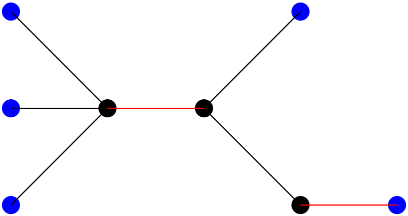
The Algorithm of [6] – First Phase

- All maximal paths of T are contracted
  - No vertices of degree 2
  - All other vertices having the same degree as in T



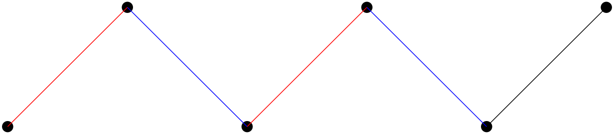
The Algorithm of [6] – First Phase

- Charge from the leaves are then redistributed among the internal vertices
- Charge of each internal vertex with degree n is at least $\left(1 - \frac{1}{n}\right) 2\pi$
  - Covers any set of n edges
  - All vertices can cover their incident edges



The Algorithm of [6] – Second Phase

- Edges of each path contracted in phase 1 are split into two matchings M_1 and M_2
  - Equal number of edges
  - Last edge in odd-length path is not in either matching



The Algorithm of [6] – Second Phase

- $w(G)$ denotes the *weight*, or total edge length, of a graph G
- Edges of the matching with the larger weight are removed
- A set of new edges called *shortcuts* are introduced
- Handled in four cases
    - Heavier matching
    - Even or odd path length

The Algorithm of [6] – Second Phase

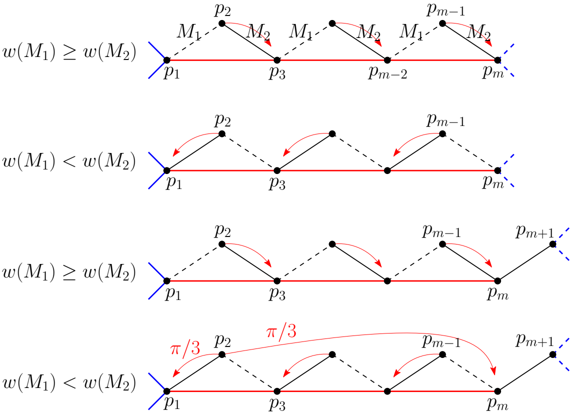


Figure: Borrowed from [6].

The Algorithm of [6] – Weight of Resulting Tree

- M_1' and M_2': union of edges in smaller and larger matchings
- T': final tree
- T: original tree
- E: edges not in $M_1' \cup M_2'$
- S: shortcuts

By the triangle equality $w(S) \leq w(M_1') + w(M_2')$.
Since $w(M_2') \geq w(M_1')$:

$$\begin{aligned} w(T') &= w(E) + w(M_1') + w(S) \\ &\leq w(E) + w(M_1') + w(M_1') + w(M_2') \\ &= w(T) + w(M_1') \leq \frac{3}{2} w(T) \end{aligned}$$

Table of Contents

The Improved Algorithm

- We modify phase 2 of Algorithm 1
- Show that the 3 edges incident to new degree-3 vertices can be covered by $\frac{5\pi}{4}$ charge
  - *Save* remaining $\frac{\pi}{12}$ charge
- Use saved charge to achieve a better approximation w.r.t. original MST

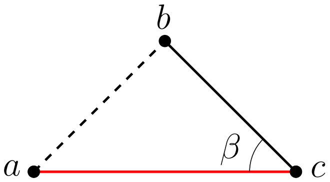
Lemma 1

Lemma 1

It is possible to save at least $\frac{\pi}{12}$ charge from every shortcut performed by phase 2 of Algorithm 1.

Lemma 1 – Proof

- Consider a shortcut ac on a contracted path.
- WLOG assume that ab is in M_2
  - Removed in phase 2
- Since (a, b, c) is part of the MST, ac is the largest edge of $\triangle abc$
- $\angle abc$ is its largest angle.
- Therefore $\beta \leq \frac{\pi}{2}$.

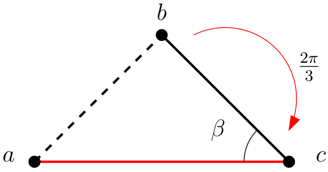


Lemma 1 – Proof

After replacement of ab by the shortcut ac

- Charge assigned to a remains enough to cover its incident edges
- Since b has degree 1, its $\frac{2\pi}{3}$ charge is free

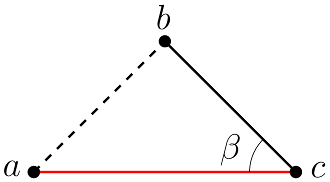
Algorithm 1 transfers this free charge to c to cover its new edge.



Lemma 1 – Proof

We show how to cover all edges incident to c while saving $\frac{\pi}{12}$ charge.

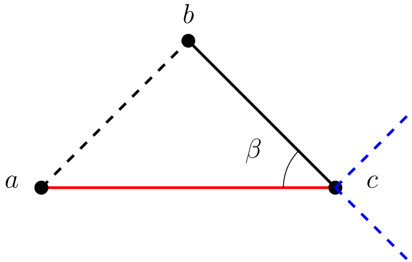
- If c's original degree was 4 or 5 then it carries at least 2π charge
- Sufficient to cover its edges



Lemma 1 – Proof

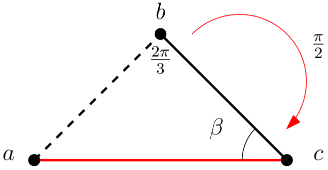
We may assume that the original degree of c is 1, 2, or 3

- Holds a charge of 0, $\frac{2\pi}{3}$, or $\frac{4\pi}{3}$, respectively.
- Thus the new degree of c (after phase 2) is 2, 3, or 4.



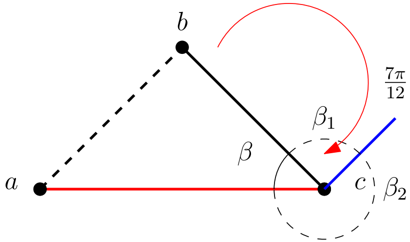
Lemma 1 – Proof

- If $\deg(c) = 2$: the two incident edges of c are ac and bc.
  - We can cover these edges by a charge of β ($\leq \frac{\pi}{2}$).
  - We transfer $\frac{\pi}{2}$ charge from b to c.
  - Save $\frac{\pi}{6}$.



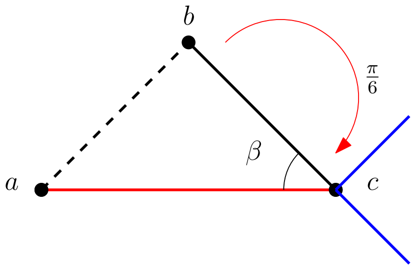
Lemma 1 – Proof

- If $\deg(c) = 3$: cover β and the smaller of the other two angles at c.
  - Three incident edges to c can be covered by charge of $\beta + \left(\frac{2\pi - \beta}{2}\right) = \frac{2\pi + \beta}{2} \leq \frac{2\pi + \frac{\pi}{2}}{2} = \frac{5\pi}{4}$.
  - Transferring $\frac{7\pi}{12}$ from b to c gives it charge of $\frac{5\pi}{4}$.
  - Save $\frac{\pi}{12}$.



Lemma 1 – Proof

- If $\deg(c) = 4$: transfer $\frac{\pi}{6}$ charge from b to c
  - c now holds $\frac{3\pi}{2}$ charge which covers its four incident edges
  - Save the remaining $\frac{\pi}{2}$.



Lemma 1 - Corollary

Corollary 2

It is possible to save $\frac{\pi}{3}$ charge from every four shortcuts that are performed by Algorithm 1.

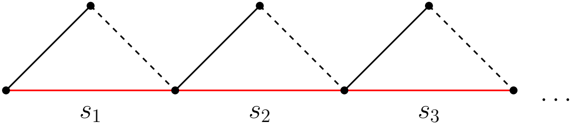
Reversing Shortcuts

Theorem 3

Given a set of *n* points in the plane and an angle $\alpha \geqslant \frac{2\pi}{3}$, there is an $\overline{\alpha}$-spanning tree of length at most $\frac{13}{9}$ times the length of the MST. Furthermore, there is an algorithm to find such an $\overline{\alpha}$-ST that runs in linear time after computing the MST.

Proof of Theorem 3

- T: the (degree-5) minimum spanning tree of a point set
  - T': the $\overline{\frac{2\pi}{3}}$-spanning tree obtained from T by Algorithm 1
- Split the sequence of all shortcuts introduced by Algorithm 1 into nine sets $S_0, \ldots, S_8$.
  - No two adjacent shortcuts in the same contracted path in the same set S_i.
  - Number of shortcuts in any two sets S_i and S_j differ by at most 1.



Proof of Theorem 3

Recall: edges of each contracted path in Algorithm 1 are split into two matchings M_1 and M_2.

- M'_1: the union of the smaller-weight matchings from each contracted path
- M'_2: the set of edges in the heavier matchings
- S_8: the set whose corresponding edges in M'_1 have the largest total weight among $S_0, \ldots, S_8$.

Proof of Theorem 3

Our plan: reverse the shortcuts in S_8 by replacing them by their corresponding edges in M_2'.

- C: the pool of charges that is obtained after phase 1 of Algorithm 1.
- Add charge from each shortcut saved by Lemma 1 to C

We will show that to reverse each shortcut from S_8 it suffices to take $\frac{2\pi}{3}$ charge from C.

Proof of Theorem 3

- *ac* – any shortcut from S_8 between two consecutive edges of a contracted path.
- *Reverse* this shortcut by replacing *ac* with the removed edge *ab*.
  - Reclaim charge transferred from *b* to *c*.

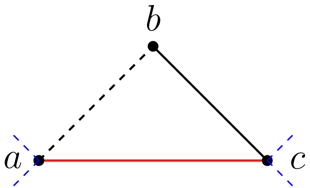


Figure: Left: The tree T' before reversing shortcut *ac*. Right: The tree T'' after reversing *ac*.

Proof of Theorem 3

- No two shortcuts in S_8 adjacent in the same contracted path
  - Can analyze a reverse operation independently of others.
- Reverse operation does not change the degree of *a*
  - Charge remains sufficient to cover its edges.

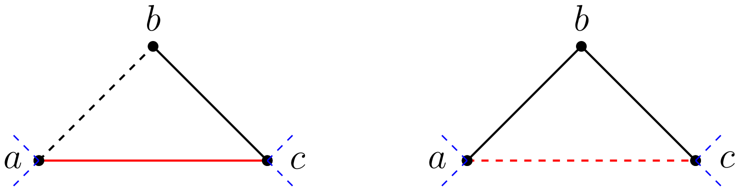
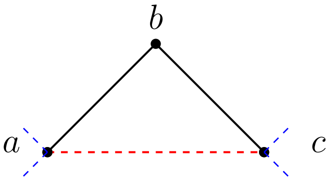


Figure: Left: The tree T' before reversing shortcut *ac*. Right: The tree T'' after reversing *ac*.

Proof of Theorem 3

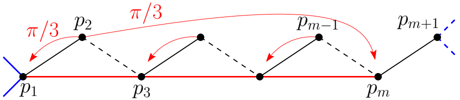
- Take $\frac{\pi}{3}$ charge from C for b to bring it to a charge of π
  - Covers its two incident edges.
- If $\deg(c) = 1$ or $\deg(c) \geq 3$, its charge is sufficient to cover its edges.
- If $\deg(c) = 2$ then we take an additional charge of $\frac{\pi}{3}$ from C for c to cover its two incident edges.



Proof of Theorem 3

- Problematic case: $w(M_1) < w(M_2)$ and the path has odd number of edges
- $p_2 = b$ holds $\frac{\pi}{3}$ charge
  - Take $\frac{2\pi}{3}$ from C for p_2 for its two incident edges

$w(M_1) < w(M_2)$



- $p_1 = c$ is of degree 1 or at least 3 (as the contracted path is maximal)
  - Charge is sufficient to cover its edges

Proof of Theorem 3

- Worst case: we take $\frac{2\pi}{3}$ from C to reverse every shortcut.
- After reversing all shortcuts in S_8, C is left with at least $\frac{2\pi}{3}$.
  - Can be distributed among the leaves of the resulting tree.

Proof of Theorem 3 – Weight of the Resulting Tree

- T'': the $\frac{2\pi}{3}$-ST obtained from T' after reversing all shortcuts in S_8.
- E: edges of T'' not in $M_1' \cup M_2'$.
- E': all edges of $M_1' \cup M_2'$ that correspond to the shortcuts in S_8.
- $M_1'' = M_1' \setminus E'$ and $M_2' = M_2' \setminus E'$

Then,

$$\begin{aligned} w(T'') &= w(E) + w(E') + w(S') + w(M_1'') \\ &\leq w(E) + w(E') + w(M_1'') + w(M_2'') + w(M_1'') \\ &= w(T) + w(M_1''). \end{aligned}$$

Proof of Theorem 3 – Weight of the Resulting Tree

Since S_8 has the largest corresponding M_1' weight, $w(M_1'') \leq \frac{8}{9}w(M_1') \leq \frac{8}{9} \cdot \frac{1}{2}w(T) = \frac{4}{9}w(T)$. Thus,

$$w(T'') \leq w(T) + \frac{4}{9}w(T) = \frac{13}{9}w(T).$$

Theorem 3 – Corollary

Corollary 4

$\frac{4}{3} \leq A\left(\frac{2\pi}{3}\right) \leq \frac{13}{9}$.

Table of Contents

Conclusions

- Open problem: Gap between the upper bound of $\frac{13}{9}$ and lower bound of $\frac{4}{3}$ for $A\left(\frac{2\pi}{3}\right)$.
  - A new algorithm with a better approximation factor
  - A new set of points whose $\overline{\frac{2\pi}{3}}$-MST must have a weight of more than $\frac{4}{3}$ times that of the MST.