



Sharif University of Technology

Scientia Iranica

Transactions D: Computer Science & Engineering and Electrical Engineering

<http://scientiairanica.sharif.edu>

A degree 3 plane 5.19 -spanner for points in convex position

D. Bakhshesh^{a,*} and M. Farshi^b

a. Department of Computer Science, University of Bojnord, Bojnord, Iran.

b. Department of Mathematical Sciences, Combinatorial and Geometric Algorithms Lab., Yazd University, Yazd, P.O. Box 89195-741, Iran.

Received 18 August 2020; received in revised form 17 March 2021; accepted 19 July 2021

KEYWORDS

Plane spanner;
Stretch factor;
Greedy spanner;
Minimum spanning tree;
Traveling salesperson tour.

Abstract. Let S be a set of n points in the plane that is in convex position. Using the well-known path-greedy spanner algorithm, this study presents an algorithm that constructs a plane $\frac{3+4\pi}{3}$ -spanner G of degree 3 on the point set S . Recently, Biniaz et al. [Biniaz, A., Bose, P., De Carufel, J.-L., Gavoille, C., Maheshwari, A., and Smid, M. “Towards plane spanners of degree 3”, *Journal of Computational Geometry*, **8**(1), pp. 11–31 (2017).] proposed an algorithm that constructs a degree 3 plane $\frac{3+4\pi}{3}$ -spanner G' for S . It was found that there was no upper bound with a constant factor in the total weight of G' , but the total weight of G was asymptotically equal to that of the minimum spanning tree of S .

© 2021 Sharif University of Technology. All rights reserved.

1. Introduction

Let S be a set of points in the plane. A weighted graph G with vertex set S is called *geometric* if any edge (p, q) of G is the straight line between p and q and its weight is $|pq|$, which is the Euclidean distance between p and q . The total weight of the graph G is the sum of the weights of all edges of G and is denoted by $wt(G)$. Let $t > 1$ be a real number. The geometric graph G is called t -spanner for S , if for any two vertices p and q in G , there exists a path P between p and q in G such that $|P| \leq t|pq|$, where $|P|$ denotes the length of the path P which is the sum of the weight of all edges on P . For any two points u and v in a geometric graph G , let $\delta_G(u, v)$ be the length of the shortest path between

u and v in G . The *stretch factor* (dilation) between u and v is defined as the ratio $\frac{\delta_G(u, v)}{|uv|}$ and we denote it by $SF_G(u, v)$. The stretch factor $SF(G)$ of a graph G is defined as:

$$SF(G) = \max_{u, v \in G} SF_G(u, v).$$

Note that when a geometric graph G is t -spanner, clearly $SF(G) \leq t$. We refer the reader to the book [1] and the papers [2–9] for an overview of t -spanners and the related algorithms.

A *plane spanner* of bounded *degree* is a spanner whose edges do not cross each other and whose maximum degree is bounded by a constant. In Table 1, some of the results related to the plane spanner of bounded degree are summarized. Note that since the stretch factor of a Hamiltonian path through a set of points arranged in a grid is $\Omega(\sqrt{n})$ (see [1]), the lower bound on the maximum degree of a t -spanner is 3. The points of the set S are said to be in *convex position* if all points of S are the vertices of the convex hull of S . The lower bound of the maximum degree of t -spanners

*. Corresponding author. Tel.: +98-58-32201000;
Fax: +98-58-32284605
E-mail addresses: d.bakhshesh@ub.ac.ir (D. Bakhshesh);
mfarshi@yazd.ac.ir (M. Farshi)

Table 1. Results of bounded degree plane spanners.

Points in non-convex or convex position		
Reference	Degree	Upper bound on the stretch factor
Bose et al. [12]	27	≈ 8.27
Li and Wang [13]	23	≈ 6.43
Bose et al. [14]	17	≈ 23.56
Perkovic and Kanj [15]	14	≈ 2.91
Bonichon et al. [16]	6	6
Bose et al. [17]	6	≈ 81.66
Bonichon et al. [18]	4	≈ 156.82
Kanj et al. [10]	4	20
Points in convex position		
Kanj et al. [10]	3	20
Biniiaz et al. [19]	3	≈ 5.19

for points in the convex position is also 3 (see [10]). Das and Heffernan [11] proved that spanners of maximum degree 3 always exist.

One of the famous algorithms for constructing a t -spanner on a given point set S is the *path-greedy spanner* algorithm or *greedy spanner* algorithm for short. The algorithm is given as follows. First, the algorithm sorts all pairs of points in nondecreasing order of their Euclidean distance. Assume that the sorted data are stored in a list L . Let E be the edge set of the graph computed by the algorithm. First, the edge set E is considered empty. Next, the algorithm processes the pairs of points in L in order. Suppose that the algorithm wants to process the pair $(p, q) \in L$. If the length of the shortest path between p and q in the graph computed so far is greater than $t|pq|$, the algorithm adds the pair (p, q) to E ; otherwise, the algorithm processes the next pair of points in L . The computed graph by the algorithm is called the *path-greedy spanner* or the *greedy spanner*. Algorithm 1, PATHGREEDY(S, t), shows the

```

1  $G := (S, E)$ ;
2 foreach pair  $(u, v) \in L$  (in sorted order) do
3   if SHORTESTPATH( $G, u, v$ )  $> t \cdot |uv|$  then
4      $E := E \cup \{(u, v)\}$ ;
5   end
6 end
7 return  $G(S, E)$ ;

```

Algorithm 2. MODIFIEDPATHGREEDY(S, E, L, t).

pseudocode of the greedy spanner algorithm. Now, we describe a modified version of the greedy spanner algorithm that we need later. In PATHGREEDY(S, t), the algorithm starts with a sorted list L and empty edge set E . The modified version of this algorithm, MODIFIEDPATHGREEDY(S, E, L, t) (see Algorithm 2), takes two extra parameters: an edge set E and a sorted list L . If $E = \emptyset$ and L is the sorted list of all $\binom{n}{2}$ pairs of points of S in non-decreasing order of their distances, then both algorithms PATHGREEDY(S, t) and MODIFIEDPATHGREEDY(S, E, L, t) generate the same graph.

In 2017, Biniiaz et al. [19] presented an algorithm that constructs a plane $\frac{3+4\pi}{3}$ -spanner of maximum degree at most 3 for any set S of points in the plane that is in convex position. Let \mathcal{P} be a convex polygon. If a vertex or a side of \mathcal{P} is removed, then the resulting chain is called a *convex chain*. The algorithm proposed by Biniiaz et al. [19] works as follows. Let $CH(S)$ be the boundary of the convex hull of S . At first, $CH(S)$ is added to the spanner. Then, it selects the farthest pair (p, q) of points of S . Then, it adds a special matching between two convex chains obtained by removing p and q from $CH(S)$ (see Algorithm 3). The matching for the two convex chains is computed as follows. First, they compute the closest pair between two convex chains that are separated by a line. Given this closest pair, we split the two chains and recurse on both sides.

In this paper, we focus on constructing a bounded-degree plane spanner for points in the convex position of degree at most 3. Using the algorithm MODIFIEDPATHGREEDY, we propose an algorithm

input: a set S of n points in \mathbb{R}^d and a real number $t > 1$.
output: t -spanner $G(S, E)$.

```

1 Sort  $\binom{n}{2}$  pairs of points in non-decreasing order of their distances (ties are
  broken arbitrarily), and store them in list  $L$ ;
2  $E := \emptyset$ ;
3  $G := (S, E)$ ;
4 foreach pair  $(u, v) \in L$  (in sorted order) do
5   if SHORTESTPATH( $G, u, v$ )  $> t \cdot |uv|$  then
6      $E := E \cup \{(u, v)\}$ ;
7   end
8 end
9 return  $G(S, E)$ ;

```

Algorithm 1. PATHGREEDY(S, t).

input: A non-empty finite set S of points in the plane that is in convex position

output: A plane degree-3 spanner of S .

- 1 $(p, q) :=$ a farthest pair of points of S ;
 - 2 $C_1, C_2 :=$ the two chains obtained by removing p and q from $CH(S)$;
 - 3 $E' := CH(S) \cup \text{MATCHING}(C_1, C_2)$;
 - 4 **return** $G' = (S, E')$;
-

Algorithm 3. DEG3PLANE SPANNER(S) [19].

that constructs a plane $\frac{3+4\pi}{3}$ -spanner of degree at most 3 for points in the convex position. Then, it is shown that the proposed plane spanner can be computed in $O(n^2 \log n)$ time. In [19], Biniaz et al. did not mention the time complexity of their algorithm (Algorithm 3). In [20], Biniaz and Smid presented an $O(n \log^7 n)$ -time algorithm that computes the plane spanner generated by Algorithm 3. We demonstrate that there is no upper bound with a constant factor on the total weight of the spanner proposed by Biniaz et al. [19], but using the concept of *generalized leapfrog property* (see [1]), our study shows that for any set S of points in the plane that is in convex position, the total weight of our proposed plane spanner is asymptotically equal to the weight of the minimum spanning tree of S ($MST(S)$).

2. Preliminaries

In this section, some definitions and notations used in the following sections are presented. Throughout this paper, it is assumed that S is a set of n points in the plane that is in a convex position. The farthest pair (p, q) of points of S is called a *diametral pair*, p and q are called *diametral points*, and the Euclidean distance $|pq|$ is called the *diameter* of S . We assume, without loss of generality, that the diametral pair (p, q) of S is horizontal and p is to the left of q . We denote the set of all points of $S \setminus \{p, q\}$, which are above the line segment pq and below pq by **upper** and **lower**, respectively. Let D_p and D_q be two closed disks with radius $|pq|$ centered at p and q , respectively. The intersection of D_p and D_q is denoted by $L(p, q)$ and is called the *lune* of p and

q . In the following sections, we use the notation G to refer to the plane spanner proposed in the current paper and G' to refer to the plane spanner generated by Algorithm 3. In the graphs G or G' , an edge (a, b) is called a *shortcut edge* if $a \in \text{upper}$ and $b \in \text{lower}$ or $a \in \text{lower}$ and $b \in \text{upper}$.

3. A degree 3 plane spanner for points in convex position

In this section, an algorithm that constructs a plane $\frac{3+4\pi}{3}$ -spanner $G = (S, E)$ of degree at most 3 for S is proposed. The idea of the algorithm is as follows. The algorithm starts with $E = CH(S)$. Then, we run MODIFIEDPATHGREEDY(S, E, L, t), where $t = \frac{3+4\pi}{3}$ and L contains all pairs of points (a, b) with $a \in \text{upper}$ and $b \in \text{lower}$ that are sorted by the non-decreasing function of the Euclidean distance $|ab|$ (see Figure 1). The graph G consists of $CH(S)$ and the output of MODIFIEDPATHGREEDY(S, E, L, t) (see Algorithm 4).

Now, it is proven that the output of algorithm GREEDYPLANE SPANNER(S) is $\frac{3+4\pi}{3}$ -spanner. We start with the following lemmas, which are needed later:

Lemma 1. *Let S be a finite set of at least two points in the plane and let (p, q) be any diametral pair of S . Then, the points of S lie in $L(p, q)$.*

Lemma 2 [19]. *Let C be a convex chain with endpoints p and q . If C is in $L(p, q)$, then the stretch factor of C is at most $\frac{2\pi}{3}$.*

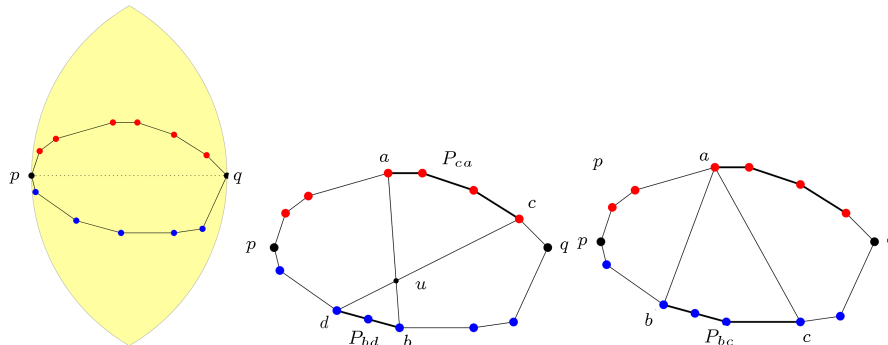


Figure 1. (a) The points with red color forming U , and the points with blue color forming L , and the yellow region as $L(p, q)$. (b) Illustrating the proof of Theorem 2. (c) Illustrating the proof of Theorem 3.

input: a set S of n points in the plane that is in convex position.
output: a plane $\frac{3+4\pi}{3}$ -spanner G of degree at most 3.
1 $(p, q) :=$ a diametral pair of points of S ;
2 **upper** $:=$ set of all points of S which are above the line segment pq ;
3 **lower** $:=$ set of all points of S which are below the line segment pq ;
4 $L :=$ list of all pairs of points (a, b) with $a \in \mathbf{upper}$ and $b \in \mathbf{lower}$ and sorted by non-decreasing the Euclidean distance $|ab|$;
5 $E := CH(S) \cup \text{MODIFIEDPATHGREEDY}(S, E, L, \frac{3+4\pi}{3})$;
6 **return** $G = (S, E)$;

Algorithm 4. GREEDYPLANESPANNER(S).

Theorem 1. *The graph G generated by GREEDY PLANESPANNER(S) is a $\frac{3+4\pi}{3}$ -spanner for S .*

Proof. Let $t = \frac{3+4\pi}{3}$ and let a and b be two arbitrary distinct points in S . To prove the theorem, it is sufficient to prove that $SF_G(a, b) \leq t$. Let (p, q) be a diametral pair of points selected in Line 1 of Algorithm 4. We consider two cases:

- **Case 1.** $a, b \in \mathbf{upper} \cup \{p, q\}$ or $a, b \in \mathbf{lower} \cup \{p, q\}$. We prove this case for $a, b \in \mathbf{upper} \cup \{p, q\}$ (the other case is symmetric). Let C_u be the convex chain connecting p to q obtained by removing all points **lower** from $CH(S)$. By Lemma 1, C_u lies in $L(p, q)$. Then, by Lemma 2, $SF_{C_u}(a, b) \leq \frac{2\pi}{3}$. Since G contains $CH(S)$, $SF_G(a, b) \leq SF_{C_u}(a, b)$. Hence, $SF_G(a, b) \leq \frac{2\pi}{3} \leq t$;
- **Case 2.** $a \in \mathbf{upper}$ and $b \in \mathbf{lower}$, or $a \in \mathbf{lower}$ and $b \in \mathbf{upper}$. Suppose, w.l.o.g., that $a \in \mathbf{upper}$ and $b \in \mathbf{lower}$. Consider the edge set E in Algorithm 4. If $(a, b) \in E$, then clearly $SF_G(a, b) = 1 \leq t$. Now, suppose that $(a, b) \notin E$. Then, according to the construction of G , there is a path between a and b of length at most $t \times |ab|$ and, therefore, $SF_G(a, b) \leq t$. This proves the theorem. \square

Theorem 2. *The graph G , generated by GREEDY PLANESPANNER(S), is plane.*

Proof. According to the construction of G , any two edges (a, b) and (c, d) in G with $a, b, c, d \in \mathbf{upper}$ or $a, b, c, d \in \mathbf{lower}$ do not cross each other. Then, to prove the theorem, it is sufficient to prove that any two shortcut edges (a, b) and (c, d) in G do not cross each other. Suppose, for contradiction, that (a, b) and (c, d) cross each other. Suppose, w.l.o.g., that the pair (a, b) is processed before the pair (c, d) by the algorithm MODIFIEDPATHGREEDY. Hence, $|ab| \leq |cd|$. Suppose, w.l.o.g., that $a, c \in \mathbf{upper}$ and $b, d \in \mathbf{lower}$. Let u be the intersection of (a, b) and (c, d) (see Figure 1). Then, based on triangle inequality, we have $|ca| \leq |au| + |uc|$ and $|bd| \leq |bu| + |ud|$. Hence, we have:

$$\begin{aligned} |ac| + |bd| &\leq |au| + |uc| + |bu| + |ud| = |ab| \\ &+ |cd| \leq 2|cd|. \end{aligned} \quad (1)$$

Let P_{ca} be the convex path between c and a on $CH(S)$ using the points on **upper**, and let P_{bd} be the convex path between b and d on $CH(S)$ using the points on **lower**. By Lemma 2, $|P_{ca}| \leq \frac{2\pi}{3}|ca|$ and $|P_{bd}| \leq \frac{2\pi}{3}|bd|$. Now, consider the path $Q := P_{ca} \cup (a, b) \cup P_{bd}$. Then, we have:

$$\begin{aligned} |Q| &= |P_{ca}| + |ab| + |P_{bd}| \leq \frac{2\pi}{3}|ca| + |ab| + \frac{2\pi}{3}|bd| \\ &= \frac{2\pi}{3}(|ca| + |bd|) + |ab|. \end{aligned}$$

Since $|ab| \leq |cd|$ and using Eq. (1), we have:

$$\begin{aligned} |Q| &\leq \frac{2\pi}{3}(|ca| + |bd|) + |ab| \leq \frac{4\pi}{3}|cd| + |cd| \\ &= \frac{3+4\pi}{3}|cd| = t|cd|. \end{aligned}$$

Hence, the algorithm does not add the edge (c, d) to G , which is a contradiction. Then, (a, b) and (c, d) do not cross each other. Hence, G is plane. \square

Now, we prove that the maximum degree of the graph G is at most 3.

Theorem 3. *The maximum degree of the graph G generated by GREEDYPLANESPANNER(S) is at most 3.*

Proof. Let a be a point in S . We show that the degree of a in G is at most 3. Note that if $a = p$ or $a = q$, where (p, q) is the diametral pair of points which is selected by the algorithm, then clearly the degree of a is 2 since $CH(S) \subseteq G$ and $p, q \notin \mathbf{upper} \cup \mathbf{lower}$. Now, suppose that $a \neq p, q$. Suppose, w.l.o.g., that $a \in \mathbf{upper}$. Since G contains $CH(S)$, then the degree of a is at least two. Suppose, for contradiction, that the degree of a is greater than 3. Hence, there exist two shortcut edges adjacent to point a . Now, suppose that (a, b) and (a, c) are two edges of G such that $b, c \in \mathbf{lower}$ (see Figure 1). Suppose, w.l.o.g., that the algorithm adds

the edge (a, b) before the edge (a, c) . Then, $|ab| \leq |ac|$. Now, let P_{bc} be the convex path between b and c on $CH(S)$ using the points of **lower**. By Lemma 2, we have: $|P_{bc}| \leq \frac{2\pi}{3}|bc|$. Let $Q = (a, b) \cup P_{bc}$. Then, we have $|Q| = |ab| + |P_{bc}| \leq |ac| + \frac{2\pi}{3}|bc|$. By the triangle inequality, we have $|bc| \leq |ab| + |ac| \leq 2|ac|$. Hence, by combining the two previous inequalities, we have $|Q| \leq \frac{3+4\pi}{3}|ac|$. Then, the algorithm does not add the edge (a, c) to G , which is a contradiction. Hence, the degree of a is at most 3. \square

4. Time complexity

We know that the running time of Dijkstra's single-source shortest paths algorithm for a weighted graph is $O(n \log n + m)$, where n is the number of vertices and m is the size of the graph. Moreover, sorting the list L in Algorithm 4 takes $O(n^2 \log n)$ time. Hence, a direct implementation of Algorithm 4 using Dijkstra's single-source shortest paths algorithm has the running time $O(n^3 \log n)$. In this section, we show that the proposed plane spanner G computed by Algorithm 4 can be computed in $O(n^2 \log n)$ time.

The main idea to reduce the running time is that we do not use Dijkstra's single-source shortest paths algorithm and, instead, by a quadratic-time preprocessing of S , we use an algorithm whose running time is $O(\log n)$ for each pair. Since there are $O(n^2)$ pairs, the overall running time will be $O(n^2 \log n)$. Note that in [21], Bose et al. demonstrated how to compute the greedy spanner on a given point set in the plane in $O(n^2 \log n)$ time. We could not apply their algorithm here. We think that the application their algorithm might not give the desired results. Now, the algorithm is described in detail.

We number the points of S in the clockwise direction, as depicted in Figure 2. Let x and y be the numbers assigned to p and q , respectively. Let T be a Binary Search Tree (BST) that is initially empty. During the running of the algorithm, upon the addition of an edge (i, j) to the graph, the numbers i and j are added to T . For two numbers $i, j \in \mathbf{upper}$ (or $i, j \in \mathbf{lower}$) with $i < j$, let P_{ij} be the path from i

to j on $CH(S)$ in the clockwise direction. Let A be an $n \times n$ array. For two numbers i and j with $i < j$ and $i, j \in \mathbf{upper}$ (or $i, j \in \mathbf{lower}$), $A[i, j]$ is equal to the length of the path P_{ij} and for other values of i and j , $A[i, j]$ is equal to zero. Since the points of S are in convex position, it is not difficult to know that the array A can be computed in $O(n^2)$ time. Now, it is time to express the algorithm. The algorithm initially adds $CH(S)$ to the edge set E and computes the list L . Suppose that the algorithm wants to process a pair (i, j) (i and j are numbers). Note that $i \in \mathbf{upper}$ and $j \in \mathbf{lower}$. The algorithm searches in T to find the smallest number k , which is greater than i and the greatest number h , which is smaller than i . Note that the number $k(h)$ may not be found in T , in which case, for the sake of simplicity, we assume that $k = 0$ ($h = 0$). It is not surprising that we can determine in $O(\log n)$ time whether the numbers k and h are found in T or not. Suppose that $k \neq 0$ and $h \neq 0$. Then, there exist two numbers $k', h' \in \mathbf{lower}$ such that $(k, k') \in E$ and $(h, h') \in E$. Now, consider two paths $P := P_{ik} \cup (k, k') \cup P_{k'j}$ and $Q := P_{hi} \cup (h, h') \cup P_{h'j}$. We claim that one of the two paths P and Q is the shortest path between i and j . To prove the claim, we first present the following theorem.

Theorem 4 [22]. *If C_1 and C_2 are convex polygonal regions with $C_1 \subseteq C_2$, then the length of the boundary of C_1 is at most the length of the boundary of C_2 .*

According to Theorem 2, the graph G is plane. Hence, through the selection of the numbers k and h , for every convex path \mathcal{C} between i and j , we have $P \subseteq \mathcal{C}$ or $Q \subseteq \mathcal{C}$. Then, by Theorem 4, P or Q is the shortest path between i and j . This proves the claim. Now, using the array A , we have $|P| = A[i, k] + |kk'| + A[k', j]$ and $|Q| = A[h, i] + |hh'| + A[h', j]$. Then, to determine whether the pair (i, j) should be added to the edge set E , it is sufficient to check if $|P| > t \times |ij|$ and $|Q| > t \times |ij|$. In the case of $k = 0$ and $h \neq 0$, it suffices to consider $P := P_{iy} \cup P_{yj}$ and $Q := P_{hi} \cup (h, h') \cup P_{h'j}$. In the cases of $k \neq 0$, $h = 0$, and $k = h = 0$, the paths P and Q are defined similarly.

5. Weight of the spanner

Now, we show that there is no upper bound with a constant factor on the total weight of the plane spanner G' proposed by Biniaz et al. [19]; however, the total weight of the plane spanner G proposed in the current paper is asymptotically equal to the $wt(MST(S))$.

Let S be a set of n points placed at vertices of a regular n -gon. Assume that n is sufficiently large. It is clear that the convex hull edges, except one edge, represent a minimum spanning tree of S . Consider the plane spanner G' on the point set S . According to Algorithm 3, the graph G' is similar to the one shown

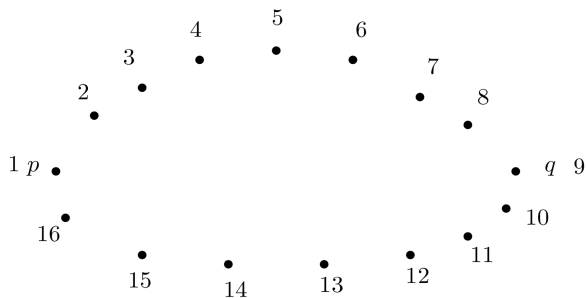


Figure 2. Numbering the point set S .

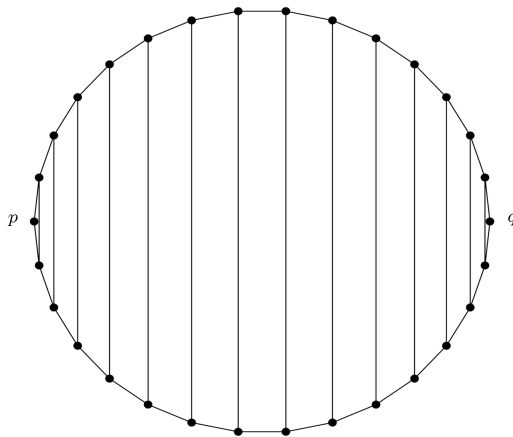


Figure 3. The plane spanner G' on a regular 30-gon.

in Figure 3. Since G' contains many shortcut edges, it is clear that $\lim_{n \rightarrow \infty} wt(G') = \infty$. This shows that the weight of the proposed plane spanner by Biniaz et al. [19] is unbounded.

Now, we will analyze the total weight of the plane spanner G . First, the generalized leapfrog property is defined, to be needed later.

Definition 1 (generalized leapfrog property [1]).

Let t_1 and t_2 be real numbers, such that $1 < t_1 < t_2$. A set E of undirected edges in \mathbb{R}^d is said to satisfy the (t_1, t_2) -leapfrog property, if for every $\{p_1, q_1\}, \{p_2, q_2\}, \dots, \{p_k, q_k\}$ of k pairwise distinct edges of E :

$$t_1 |p_1 q_1| < \sum_{i=2}^k |p_i q_i| + t_2 \left(|p_1 p_2| + \sum_{i=2}^{k-1} |q_i p_{i+1}| + |q_k q_1| \right).$$

Now, the *Generalized Leapfrog Theorem* is presented.

Theorem 5 (generalized leapfrog theorem [1]).

There exists a constant ϕ with $0 < \phi < 1$, such that the following holds. Let t_1 and t_2 be real numbers such that $1 < 1 - \phi + \phi t_2 < t_1 < t_2$. Let S be a set of points in \mathbb{R}^d and let E be a set of edges, whose endpoints are from S , and that satisfies the (t_1, t_2) -leapfrog property. Then:

$$wt(E) \leq c_{dt_1 t_2} \cdot wt(MST(S)),$$

where $c_{dt_1 t_2}$ is a real number that depends only on d , t_1 , and t_2 .

According to a graph, the length of the second shortest path between two vertices p and q in the graph is denoted by $\delta_2(p, q)$. If there is only one path between p and q in the graph, then we assume that $\delta_2(p, q) = \infty$. In the following, a sufficient condition for the leapfrog property is given.

Theorem 6 [23]. Let S be a set of n points in \mathbb{R}^d , let $t > 1$ be a real number, and let $G = (S, E)$ be

an undirected t -spanner for S . Assume that $\delta_2(p, q) > t|pq|$, for every edge (p, q) in E . Then, the edge set E satisfies the (t, t^2) -leapfrog property.

Let $G = (S, E)$ be an undirected t -spanner for S and F be a subset of E . By carefully studying the proof of Theorem 6, we find that the proof is correct even when we replace E by F in Theorem 6. Hence, the following result is obtained.

Theorem 7. Let S be a set of n points in \mathbb{R}^d , let $t > 1$ be a real number, and let $G = (S, E)$ be an undirected t -spanner for S and F be a subset of E . Assume that $\delta_2(p, q) > t|pq|$, for every edge (p, q) in F . Then, the edge set F satisfies the (t, t^2) -leapfrog property.

Let S be a set of points in the plane that is in a convex position. Consider the plane spanner $G = (S, E)$ computed by the algorithm GREEDY PLANE SPANNER(S). Now, we prove the following result.

Lemma 3. The set of all shortcut edges in G satisfies the $(\frac{3+4\pi}{3}, (\frac{3+4\pi}{3})^2)$ leapfrog property.

Proof. Let $t = \frac{3+4\pi}{3}$ and F is the set of all shortcut edges in G . By Theorem 7, to prove Lemma 3, it is sufficient to prove that for every edge $(a, b) \in F$, $\delta_2(a, b) > t|ab|$. Let P be a path between a and b in G having length $\delta_2(a, b)$.

Suppose that all edges of P are on $CH(S)$. Notice that, in this case, the path P passes through one of the diametral points. Since G initially includes the convex hull edges and (a, b) is added to the edge set of G , $|P| > t|ab|$ according to the construction of G .

Now, suppose that one of the edges of P is not on $CH(S)$. In other words, P contains a shortcut edge. Let C_1 and C_2 be two convex chains obtained by removing the points p and q from $CH(S)$. Since G is plane and the points of S are in convex position, the path P consists of some edges on C_1 , a shortcut edge (r, s) , and some edges on C_2 (see Figure 4). Suppose, w.l.o.g., that $r \in C_1$ and $s \in C_2$. Note that since G is plane, it is not difficult to see that in the quadrilateral $abrs$, there are no shortcut edges except two edges (a, b) and (r, s) . Now, we show that $|P| > t|ab|$. If the

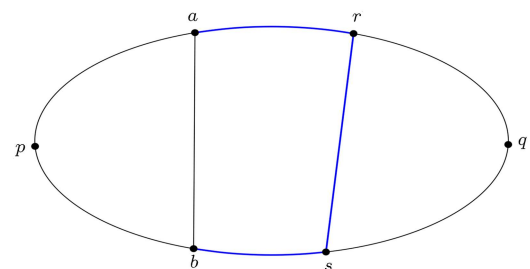


Figure 4. The path P with the gray color in the proof of Lemma 3.

algorithm adds the edge (r, s) before the edge (a, b) , then $|P| > t|ab|$ according to the construction of G . Now, suppose that (r, s) is added to G after the edge (a, b) . Then, $|rs| \geq |ab|$. Let C_{ar} be the part of the path P that is on C_1 and C_{bs} be the part of P that is on C_2 . Consider the path $Q := C_{ar} \cup (a, b) \cup C_{bs}$ between r and s . Since $|rs| \geq |ab|$, $|P| \geq |Q|$. Since (r, s) is an edge of G and Q is a path between r and s , which is created before processing the pair (r, s) , $|Q| > t|rs|$. Hence, we have:

$$|P| \geq |Q| > t|rs| \geq t|ab|.$$

This completes the proof. \square

By Theorem 5 and Lemma 3, $wt(F) = O(1) \cdot wt(MST(\text{lower} \cup \text{upper}))$. Now, in the following, we show that $wt(MST(\text{lower} \cup \text{upper})) \leq 2wt(MST(S))$.

Lemma 4. $wt(MST(\text{lower} \cup \text{upper})) \leq 2wt(MST(S))$.

Proof. Let $TSP(S)$ be the traveling salesperson tour on the point set S . It is clear that $wt(TSP(\text{lower} \cup \text{upper})) \leq wt(TSP(S))$ (see Exercise 1.7 in [1]). Since $wt(MST(\text{lower} \cup \text{upper})) \leq wt(TSP(\text{lower} \cup \text{upper}))$, we have $wt(MST(\text{lower} \cup \text{upper})) \leq wt(TSP(S))$. On the other hand, it is well known that $wt(TSP(S)) \leq 2wt(MST(S))$ (see [1]). Hence, by combining two previous inequalities, $wt(MST(\text{lower} \cup \text{upper})) \leq 2wt(MST(S))$. \square

Now, we conclude this section with the following result.

Theorem 8. For any set S of points in the plane that is in convex position, we have:

$$wt(G) = O(1) \cdot wt(MST(S)),$$

where $G = (S, E)$ is the plane spanner computed by GREEDYPLANE SPANNER(S).

Proof. Let F be the all-shortcut edges in E . Let F' be the set of all edges on $CH(S)$. Clearly, $E = F \cup F'$. By Lemma 4, we have $wt(F) = O(1) \cdot wt(MST(S))$. Now, since $TSP(S)$ only contains the convex hull edges (see [24]), we have $wt(F') = wt(TSP(S))$. Since $wt(TSP(S)) \leq 2wt(MST(S))$ (see [1]), $wt(F') \leq 2wt(MST(S))$. Now, we conclude that $wt(E) = wt(F) + wt(F') = O(1) \cdot wt(MST(S))$. \square

Whether the size (number of the edges) of the proposed plane spanner G is less than or equal to the size of the plane spanner G' proposed by Biniaz et al. [19] or not is an interesting question.

To tackle this problem, one direction is to show that the graph G is a subgraph of G' . We found the following counterexample for this. Assume the points of P placed on the sides of the rectangle $pdqe$ such that $|qd| = |cd| = |pe| = 1$, $|dc| = |qb| = 2\pi/3$ and $|ab| =$



Figure 5. The point set P .

0.4 (see Figure 5). It is quite clear that the graph G contains the edge (a, c) , while G' does not contain (a, c) , and contains the edge (b, c) instead.

6. Conclusion

This study presented an algorithm that constructs a plane $\frac{3+4\pi}{3}$ -spanner of degree at most three in $O(n^2 \log n)$ time for any set of n points in the plane that is in convex position. It was found that the total weight of the proposed plane spanner was asymptotically equal to the total weight of the minimum spanning tree of the points.

We conclude the paper with the following open problems:

1. Is the size (number of the edges) of the proposed plane spanner G less than or equal to the size of the plane spanner G' proposed by Biniaz et al. [19]?
2. Does the algorithm $\text{PATHGREEDY}(S, \frac{3+4\pi}{3})$ output a plane spanner of degree at most three for any set S of points in the plane that is in convex position?

We think that this problem may have a yes-answer. An attempt was made to demonstrate this, but we did not succeed.

Acknowledgment

Some parts of this work were done at the *Open Problem Session*, held at Yazd University, 13–15 July 2019. The first author would like to thank Mohammad Farshi for his warm hospitality at Yazd. We also thank the anonymous referees whose comments helped us improve the presentation of the results.

References

1. Narasimhan, G. and Smid, M., *Geometric Spanner Networks*, Cambridge University Press (2007).
2. Abam, M.A., Baharifard, F., Borouny, M., and Zarrabi-Zadeh, H. “Fault-tolerant spanners in networks with symmetric directional antennas”, *Theor. Comput. Sci.*, **704**, pp. 18–27 (2017).
3. Abam, M.A., De Berg, M., and Seraji, M.J.R. “Geodesic spanners for points on a polyhedral terrain”, *SIAM J. Comput.*, **48**(6), pp. 1796–1810 (2019).
4. Abam, M.A. and Qafari, M.S. “Geometric spanner games”, *Theor. Comput. Sci.*, **795**, pp. 398–407 (2019).
5. Bakhshesh, D., Barba, L., Bose, P., De Carufel, J.-L., Damian, M., Fagerberg, R., Farshi, M., van Renssen,

- A., Taslakian, P., and Verdonchot, S. “Continuous Yao graphs”, *Comp. Geom-Theor. App.*, **67**, pp. 42–52 (2018).
6. Bakhshesh, D. and Farshi, M. “Angle-constrained spanners with angle at least $\pi/3$ ”, *Inform. Process. Lett.*, **120**, pp. 44–46 (2017).
7. Bakhshesh, D. and Farshi, M. “Fault tolerancy of continuous Yao graph of angle less than $2\pi/5$ ”, *Inform. Process. Lett.*, **148**, pp. 13–18 (2019).
8. Bakhshesh, D. and Farshi, M. “(Weakly) self-approaching geometric graphs and spanners”, *Comp. Geom-Theor. App.*, **78**, pp. 20–36 (2019).
9. Iranfar, B. and Farshi, M. “On the expected weight of the theta graph on uncertain points”, *Journal of Algorithms and Computation*, **52**(1), pp. 163–174 (2020).
10. Kanj, I., Perković, L., and Turkoglu, D. “Degree four plane spanners: Simpler and better”, *Journal of Computational Geometry*, **8**(2), pp. 3–31 (2017).
11. Das, G. and Heffernan, P.J. “Constructing degree-3 spanners with other sparseness properties”, *International Journal of Foundations of Computer Science*, **07**(02), pp. 121–135 (1996).
12. Bose, P., Gudmundsson, J., and Smid, M. “Constructing plane spanners of bounded degree and low weight”, *Algorithmica*, **42**(3), pp. 249–264 (2005).
13. Li, X.-Y. and Wang, Y. “Efficient construction of low weighted bounded degree planar spanner”, *Int. J. Comput. Geom. Ap*, **14**(01n02), pp. 69–84 (2004).
14. Bose, P., Smid, M., and Xu, D. “Delaunay and diamond triangulations contain spanners of bounded degree”, *Int. J. Comput. Geom. Ap*, **19**(02), pp. 119–140 (2009).
15. Perković, L. and Kanj, I.A. “On geometric spanners of Euclidean and unit disk graphs”, In *Proc. 25th Int. Sympos. on Theoretical Aspects of Computer Science*, pp. 409–420 (2008).
16. Bonichon, N., Gavoille, C., Hanusse, N., and Perković, L. “Plane spanners of maximum degree six”, In *Automata, Languages and Programming*, S. Abramsky, C. Gavoille, C. Kirchner, F. Meyer aufder Heide, and P.G. Spirakis, Eds., Berlin, Heidelberg, Springer Berlin Heidelberg, pp. 19–30 (2010).
17. Bose, P., Carmi, P., and Chaitman-Yerushalmi, L. “On bounded degree plane strong geometric spanners”, *Journal of Discrete Algorithms*, **15**, pp. 16–31 (2012).
18. Bonichon, N., Kanj, I., Perković, L., and Xia, G. “There are plane spanners of degree 4 and moderate stretch factor”, *Discrete Comput. Geom.*, **53**(3), pp. 514–546 (2015).
19. Biniaz, A., Bose, P., De Carufel, J.-L., Gavoille, C., Maheshwari, A., and Smid, M. “Towards plane spanners of degree 3”, *Journal of Computational Geometry*, **8**(1), pp. 11–31 (2017).
20. Biniaz, A. and Smid, M., *Personal Communication* (2019).
21. Bose, P., Carmi, P., Farshi, M., Maheshwari, A., and Smid, M. “Computing the greedy spanner in near-quadratic time”, *Algorithmica*, **58**(3), pp. 711–729 (2010).
22. Benson, R., *Euclidean Geometry and Convexity*, McGraw-Hill (1966).
23. Smid, M. “Errata for the book geometric spanner networks”, (2013).
<http://people.scs.carleton.ca/~michiel/SpannerBook/errata.html>
24. Deineko, V.G., Hoffmann, M., Okamoto, Y., and Woeginger, G.J. “The traveling salesman problem with few inner points”, *Oper. Res. Lett.*, **34**(1), pp. 106–110 (2006).

Biographies

Davood Bakhshesh is an Assistant Professor at the Department of Computer Science at University of Bojnord in Bojnord, Iran and has been working at University of Bojnord since 2012. He received his BS degree in Computer Science from Vali-eAsr University of Rafsanjan in 2007, MS degree in Computer Science from Sharif University of Technology in 2009, and PhD degree in Computer Science from Yazd University in 2017.

Mohammad Farshi is an Associate Professor at the Department of Mathematical Sciences at Yazd University in Yazd, Iran and has been working in the Yazd faculty since 1999. He received his BS in Computer Science from Yazd University in 1996, MS degree in Pure Mathematics from Shiraz University in 1999, and PhD degree in Computer Science from Eindhoven University of Technology (TU/e) in 2008.