Theta Graph Report

Xinyu Luo (xl3147)

I. History:

The story of spanners is actually started by a bunch of people who wanted to simplify the complete graph. In 1986, Chew tried to simplify a graph to a planar graph, and he succeeded with a transmission distance between each pair of vertices at most $\sqrt{10}$ times of the euclidean distance based on Delaunay Triangulation, and an algorithm with time complexity $O(nlogn)$ to build this graph [5]. Well, in 1988, J. Keil mentioned an approach to simplify the complete graph which is better than Delaunay Approximating, this approach is based on the theta graph [3]. In 1991, J. Ruppert and R. Seidel came with another good bound for approximating complete euclidean graphs, which is also based on theta graph but with a completely different proving method, and the graph building algorithm was extended to d-dimensional [1]. And, in 1992, J. Keil came back with a summary of approximating the complete Euclidean graph without realizing there was a better bound for theta graph that came out one year ago [2]. Then, the spanners topic was staying in silence for 15 years.

With the flourishing of the mobile industry, the requirement for networks became higher, so people cared about this topic again. In 2007, a textbook by G. Narasimhan et. called "Geometric Spanner Networks" came out, then wiki named the definition in this book as the official definition for spanners [10]. Then, in these years many researchers came with tighter upper bounds for spanners and clearer proof about the previous results. The new era for this topic was coming [6, 7, 8, 9].

II. Different proofs:

The different classes of spanners can be divided by the polygon approach and theta graph. The first one is mainly the Chew algorithm which is already shown in course notes. For the second class, there are some different proofs J. Keil (1988), J. Ruppert and R. Seidel (1991), clearer proof for J. Keil is from G. Narasimhan (2007), clearer proof for J. Ruppert and R. Seidel is from a lot of people my reference is from A. Verdonschot (2015) [1, 2, 3, 5 , 10].

A. J. Keil (1988) and G. Narasimhan (2007) proof for spanning ratio $\frac{1}{cos\,\theta - sin\,\theta}$

They got this result for $k > 8$. This constraint is because when $k > 8$ it can make sure that the new distance is smaller than the old distance to the target proved by Trig. Thus, there will be no cycle on the path.

The original version is J. Keil version, this proof is 3 pages long, the clearer proof is given by G. Narasimhan and it is one page.

B. J. Ruppert and R. Seidel (1991) and A. Verdonschot (2015) proof for spanning ratio $\frac{1}{1-2sin\frac{\theta}{2}}$

The spanning ratio is improved from $\frac{1}{cos\,\theta - sin\,\theta}$ to $\frac{1}{1-2sin\frac{\theta}{2}}$, based on another proof based on triangle inequality. And also, they improved $k > 8$ to $k > 6$. They found this is because Keil's constructing theta graph algorithm required $k > 6$. They said Keil said this, but I didn't find this was told in Keil's paper. I will write some of my ideas about why it is $k > 6$ at the end of the proof.

J. Ruppert and R. Seidel (1991):

Construct a path from p to q: greedy algorithm (PS: In the original paper, it said if there is a direct edge between p_i and q connect it, otherwise, connect the vertex in the cone which p_i *and* q belongs to has the closest orthogonal projection onto the axis of that cone. I think it is the greedy algorithm.) [1]

Def. Let the path between p and q is $p_0 p_1 \cdots q$, the ith p is p_i , e_i is the edge from p_{i-1} to p_i. Let l_i be the euclidean distance between p_i and q ($dist(p_i,\ q)$), l_{i-1} be the $dist(p_{i-1},\ \ q)$, there are m edges on the path.

We get the formula 1 $|e_i| \ + \ l_i \ \le \ l_{i-1} + 2|e_i|sin\frac{\theta}{2}$, then by telescoping sum, we get routing ratio $\frac{1}{1-2sin\frac{\theta}{2}}$.

Prove formula 1:
If $p_i = q$, trivially this hold.

Otherwise, there are three cases.

Def. Let α be the angle between ray $p_{i-1}p_i$ and the cone axis. Let β be the angle between $p_{i-1}q$ and the cone axis. r is the symmetry vertex of p_i from the cone axis.

Case 1: $\beta < \alpha$ (Figure 1.)
First, p_{i-1} is not directly connected to q which means q has to be at the right side of p_i.
Then,
$|p_{i-1}r| \ \le |p_{i-1}w| \ + \ |wr| \ and \ |p_iq| \ \le |p_iw| \ + \ |wq|$
$\Rightarrow |p_{i-1}r| \ + \ |p_iq| \ \le |p_{i-1}q| \ + \ |p_ir|$
$\Rightarrow |e_i| \ + \ l_i \ \le \ l_{i-1} + 2|e_i|sin\alpha \ \le \ l_{i-1} + 2|e_i|sin\frac{\theta}{2}$

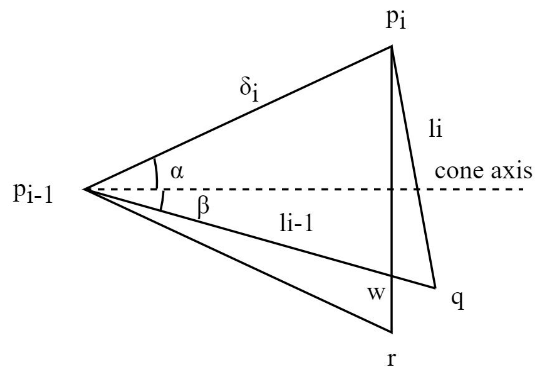

Figure 1. Case 1

Case 2: $\beta \geq \alpha$, p_i *and* q *are on the same side* (Figure 2.)
First, make a circle C_0 with the center p_{i-1} and radius $|p_{i-1}p_i|$, the intersection point with ray $p_{i-1}q$ is r, then s is an image cross cone axis of r. Make a circle C with the center q and radius $|p_iq|$.

If we get $|p_{i-1}p_i| + |p_iq| \leq |p_{i-1}q| + |rs|$, *then* $|e_i| + l_i \leq l_{i-1} + 2|e_i|sin\beta \leq l_{i-1} + 2|e_i|sin\frac{\theta}{2}$

We have $|p_{i-1}p_i| = |p_{i-1}q| - |rq|$, *we need to prove* $|p_iq| \leq |rq| + |rs|$.

For $|qs| \leq |rq| + |rs|$, *only need to prove* $|p_iq| \leq |qs|$.

We notice that p_{i-1} *and* q *are the center for the two circle, so the intersect points of* C *and* C_0 p_i *and* w *should be on the different side of ray* $p_{i-1}q$, *and one of the intersection is* p_i, *thus,* s *must outside the circle* C. *We get* $|p_iq| \leq |qs|$.

Case 3: $\beta \geq \alpha$, p_i *and* q *are on the different side* (Figure 2.)
The proof is the same as case 2.

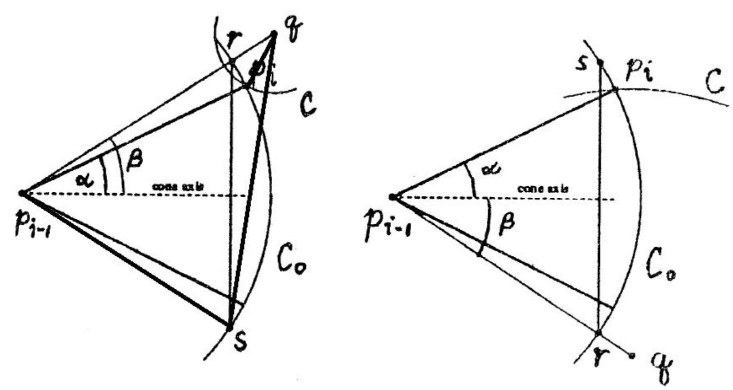

Figure 2. Case 2 and Case 3

About why $k > 6$?
To be honest, the author thought out this spanning ratio based on $k > 6$. We already know p_i is on the left side of q and we have $\frac{\angle p_{i-1}p_iq}{p_{i-1}q} = \frac{\angle p_ip_{i-1}q}{p_iq}$, thus,

$\angle p_{i-1}p_i q > \angle p_i p_{i-1} q$ when $k > 6$, so $|p_i q| < |p_{i-1} q|$. There will not be a cycle. Also, the spanning ratio becomes infinity when $k = 6$ and negative when $k < 6$.

Clearer proof in some recent paper (2015) which is very different from 1991 paper:

This proof comes from a proof for Yao's graph. [4]

Def. A path from a to b, c is the only intermediate point. Rotate c towards ab, using a as the center of the circle and $|ac|$ as the radius for θ degrees, then get c', as figure 3 showed on the next page. We call the arc from c to c' is $arc\ C$.

Proof for spanning ratio $\frac{1}{1-2sin\frac{\theta}{2}}$:
If cc' and ab are intersected,
we get $|ac'| \le |as| + |sc'|$ and $|bc| \le |bs| + |sc|$.
$\Rightarrow$ $|ac| + |bc| \le |as| + |sc'| + |bs| + |sc| = |ab| + |cc'| = |ab| + 2sin\frac{\theta}{2}|ac|$,
$Then,\ it\ is\ the\ same\ as\ the\ formula\ 1,\ we\ get\ |e_i| + l_i \le l_{i-1} + 2|e_i|sin\frac{\theta}{2}$.

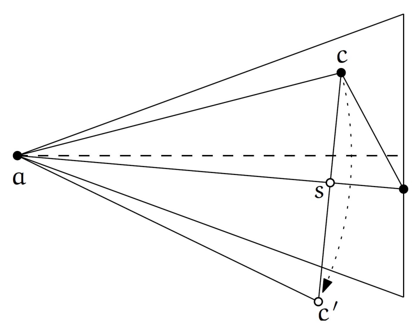

Figure 3. Rotating proof (figure from [4])

We need to prove cc' and ab is always intersected. (This proof in the paper is not very clear, the proof below is rewritten by myself).

There will be 2 cases: $|ac| \le |ab|$ and $|ac| > |ab|$.

Case 1: $|ac| \le |ab|$.
We know that $arc\ C$ will intersect with ab. And the $arc\ C$'s chord cc' must intersect with ab for the rotating angle θ less than π.

Case 2: $|ac| > |ab|$, this time $arc\ C$ will not intersect with ab.
We select c instead of b is because c is on the left side of b. Thus, Case 2 could happen if and only if the angle σ between ac and cone axis is larger than the angle γ between ab and the cone axis, as shown in the figure 4.

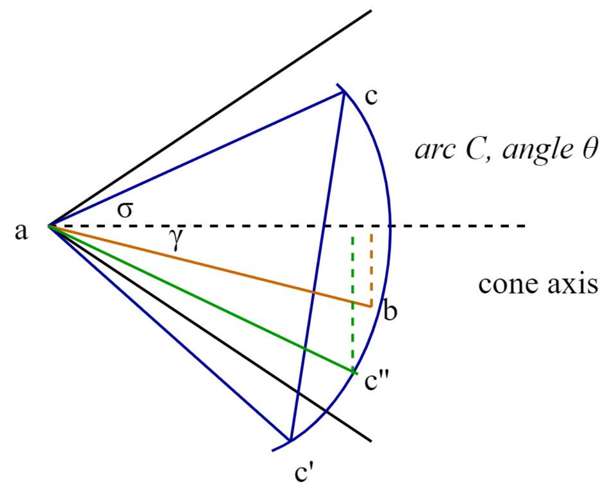

Figure 4.

Thus, if we make a mirror point c'' of cone axis for c. cc'' is must on the left of b, and $\angle c''ac = 2\sigma < \theta$, thus, c' is on the left of b, and the different side of ab. Also, cc' will on the right side of a when $\theta < \pi$. Thus, cc' has to intersect with ab .

C. $Theta-4$, $Theta-5$, $Theta-6$

We have already got all the spanning ratio for $k > 6$. And for Θ_2 and Θ_3, they are proved to be not a spanner by the translate prove for $Yao-2$ and $Yao-3$, Y_2 and Y_3 are not spanners is proved by Nawar M. El Molla in his Ph.D. thesis in 2009. For $\Theta_4\Theta_5\Theta_6$, they are all spanners, but they are proved different ways.

- **Θ_4 is a spanner:**

  Proved by L. Barba, P. Bose, et al on paper "On the stretch factor of the Theta-4 graph" in 2012. The approximating spanning ratio for Θ_4 is very loose, it is around 237. [7]

- **Θ_5 is a spanner - this is a hard topic, it is the last one solved in $\Theta_4\Theta_5\Theta_6$:**

  First mentioned by J. Ruppert and R. Seidel (1991), the same persons who get spanning ratio $\frac{1}{1-2sin\frac{\theta}{2}}$. They said in the conclusion of their paper: "In the planar case, some improvement can be made on the constants. In particular, when k is odd, there is an asymmetry between the cones at point p and q that we can take advantage of by growing paths from both ends. Interestingly, this asymmetry allows us to prove a bound near 10 on the path lengths even for the case $k = 5$. For even k, similar improvements can be made by altering the apertures of some cones at each point." Unfortunately, no details are given.

Then, P. Bose, P. Morin, et al, proved this in 2015. Actually, this paper “The theta-5 graph is a spanner” actually showed in a conference in 2013 and another conference in 2015.[9]

➢ Θ_6 **is a spanner:**

Proved by N. Bonichon, et al in the paper “Connections between theta-graphs, Delaunay triangulations, and orthogonal surfaces”. He found the connection between theta-graphs and DT. The DT part was mentioned in Chew’s paper in 1986.[8]

PS: It is very interesting that when Y_k graph doesn’t fail as a spanner, Θ_k never fail as well, it is still an opening question to prove when Y_k graph is a spanner and Θ_k graph is a spanner as well. [4]

III. O(nlogn) time complexity algorithm for building theta graph in 2-dimensional (plane swipe algo):

The O(nlogn) algorithm in 2-dimensional space is first mentioned by J. Keil (1988), using the 2-3 tree. And this algorithm is transferred to fit high dimensional space by J. Ruppert and R. Seidel (1991), using the range tree, time complexity $O(n(logn)^{d-1})$. In the book of G. Narasimhan (2007), they said using skip lists makes the algorithm more clear for the high dimensional space. [3, 10]

(You may think the algorithm below is super easy, but you can never imagine how abstract this is described in the original paper… and the textbook has some very misleading parts…)

To explain this easier, we assume that the cone we are looking at the cone axis is the x-coordinate (we can do it by rotation for other cones). Some def. will be shown in Figure 5.

Def. The point set on this plane is $A = \{a_1 ... a_n\}$, we sort A by x-coordinate (it is the same as sorting by the project of the cone axis).

Def. For a cone original from the point a, the upper bound ray for this cone is h_1, the lower bound ray is h_2. D_1 is the orthogonal vector of h_1, D_2 is the orthogonal vector of h_2. The point project on D_1 is $D_1(a_i)$, the point project on D_2 is $D_2(a_i)$.

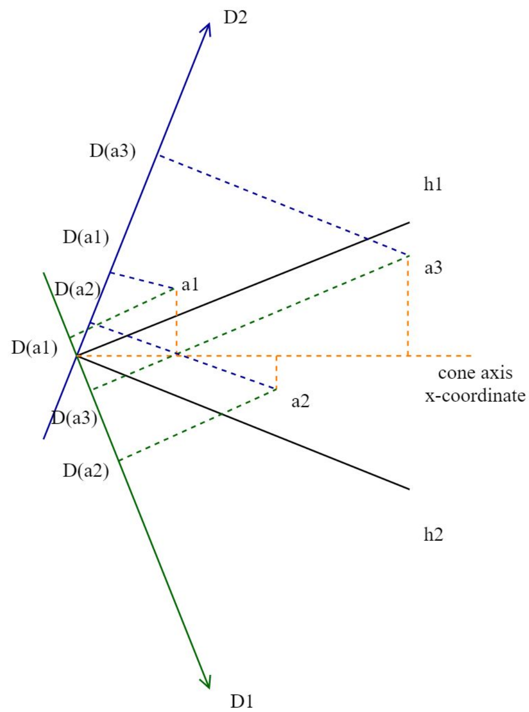

Figure 5. Project on axis

Def. P a point set equal to A but sorted by $D_1(a_i)$, Q is another point set equal to A but sorted by $D_2(a_i)$. For an example of sorting, $D_1(a_1) < D_1(a_3) < D_1(a_2)$, then we got the order for P.

Def. BST T, the leaf nodes of T represent points in Q, the rank of the leaf nodes is the same rank as Q. All the nodes of T augmenting two values, assume a node u and u has both subtrees: 1) value z_1, info for searching operation: $u's$ left subtree's rightmost leaf 2) value z_2: the point has the lowest rank in A of u's subtree. The example is in the figure below.

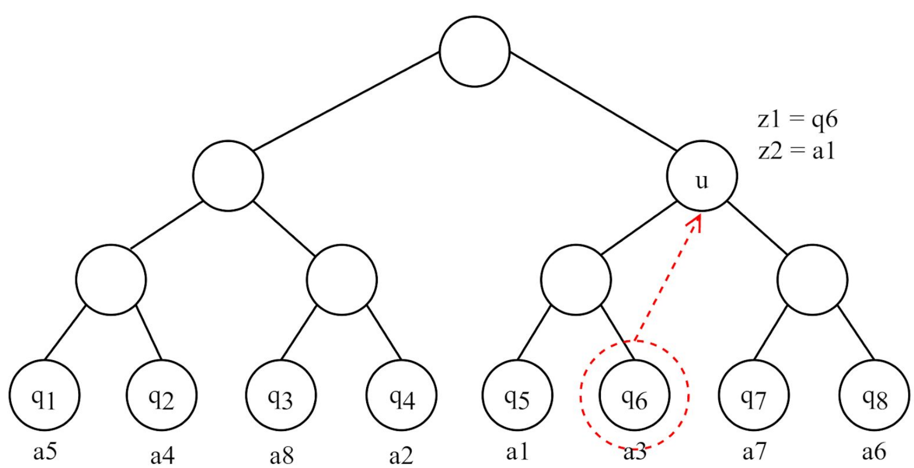

Figure 6. The structure of T

1. Initially, we have an empty BST T.
2. Iterate on P from p_n to p_1: (for example, we are dealing with p_i)
    1) Insert p_i in T (update T at the same time)
    2) Search p_i on T. For this searching path from the root, if we need to turn left on one node v, we record $v's$ right child to a new set M (clear M each iteration). This step will be shown in figure 7 (M *is the nodes in red circle*). Then, we notice that we can only collect $O(logn)$ nodes in M, and these nodes in M represent all nodes on the upper side of $p_i's$ h_2. (Because you can only turn left at node v only when you are sure all the points on the right subtree of v is projected on the positive part of D_2, which mean upper than $p_i's$ h_2).

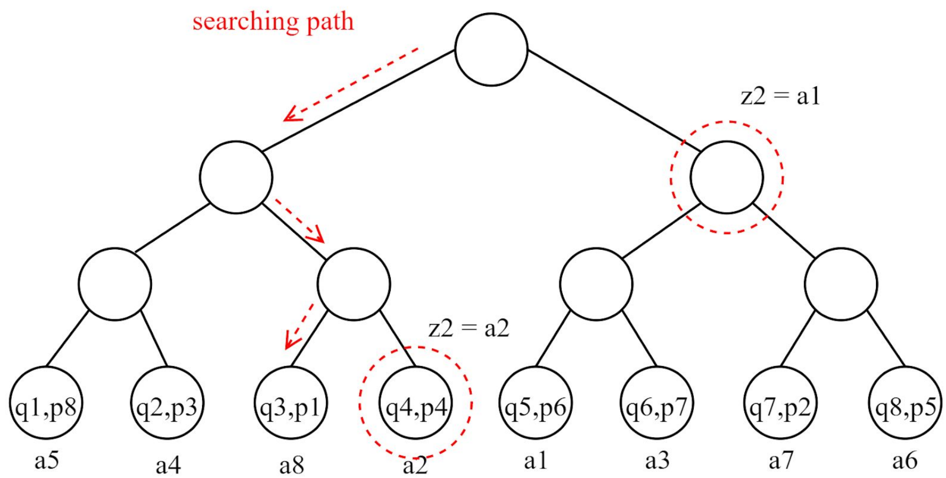

Figure 7.

    3) If M is not empty, select the minimum z_2 (for example a_t) in all nodes in M. This step helps as to select the point with the minimum x-coordinate in all points which are upper than $p_i's$ h_2. (It is all points because non-leaf nodes can represent the minimum z_2 of its subtrees.)
    4) Connect p_i with a_t. Continue the next iteration.

Extra proof for why this is correct:
We use the reverse order in P, this makes sure that when we are going to connect the edge of p_i in the cone, we make sure that $p_{i+1}...p_n$ has to be below $p_i's$ h_1, otherwise $p_{i+1}...p_n$ won't be larger than p_i on D_1. When we insert p_i to T, we get all nodes on the upper side of $p_i's$ h_2, and the nodes in the T has to be $p_{i+1}...p_n$. Thus, we get all points in between $p_i's$ h_1 and h_2 and stored in M. After that, we select the point which has minimum x-coordinate, which means the closest point of p_i in that cone.

Time complexity: $O(nlogn)$, for sorting, and n time BST operations, each cone.

Space: $O(n)$ for the BST.

IV. My experiment towards tighter upper bound (a failure for now, will try something more in the future):

I was doing the experiment for finding counterexamples for the routing ratio and the spanning ratio in the table below. I did find the routing ratio for Θ_5 fail some cases in Θ_{4k+5}. But there is a bound about $k \geq 1$, so it is a mistake, to be honest. Then, nothing is different from the table.[6]

Table 1. Different Spanning Ratio & Routing Ratio

	Current Spanning	Current Routing	Previous Spanning & Routing
$\theta_{(4k+2)}$-graph	$1+2\sin\left(\frac{\theta}{2}\right)$	$\frac{1}{1-2\sin\left(\frac{\theta}{2}\right)}$	$\frac{1}{1-2\sin\left(\frac{\theta}{2}\right)}$
$\theta_{(4k+3)}$-graph	$\frac{\cos\left(\frac{\theta}{4}\right)}{\cos\left(\frac{\theta}{2}\right)-\sin\left(\frac{3\theta}{4}\right)}$	$1+\frac{2\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{4}\right)}{\cos\left(\frac{\theta}{2}\right)-\sin\left(\frac{\theta}{2}\right)}$	$\frac{1}{1-2\sin\left(\frac{\theta}{2}\right)}$
$\theta_{(4k+4)}$-graph	$1+\frac{2\sin\left(\frac{\theta}{2}\right)}{\cos\left(\frac{\theta}{2}\right)-\sin\left(\frac{\theta}{2}\right)}$	$1+\frac{2\sin\left(\frac{\theta}{2}\right)}{\cos\left(\frac{\theta}{2}\right)-\sin\left(\frac{\theta}{2}\right)}$	$\frac{1}{1-2\sin\left(\frac{\theta}{2}\right)}$
$\theta_{(4k+5)}$-graph	$\frac{\cos\left(\frac{\theta}{4}\right)}{\cos\left(\frac{\theta}{2}\right)-\sin\left(\frac{3\theta}{4}\right)}$	$1+\frac{2\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{4}\right)}{\cos\left(\frac{\theta}{2}\right)-\sin\left(\frac{\theta}{2}\right)}$	$\frac{1}{1-2\sin\left(\frac{\theta}{2}\right)}$

Reference:

[1] Ruppert J, Seidel R. Approximating the d-dimensional complete Euclidean graph. In Proceedings of the 3rd Canadian Conference on Computational Geometry (CCCG 1991) 1991 (pp. 207-210).
[2] Keil JM, Gutwin CA. Classes of graphs which approximate the complete Euclidean graph. Discrete & Computational Geometry. 1992 Jan 1;7(1):13-28.
[3] Keil JM. Approximating the complete Euclidean graph. InScandinavian Workshop on Algorithm Theory 1988 Jul 5 (pp. 208-213). Springer, Berlin, Heidelberg.
[4] Verdonschot S. Flips and Spanners. arXiv preprint arXiv:1509.02563. 2015 Sep 8.
[5] Chew, L. Paul. "There Is a Planar Graph Almost as 600d as the Complete Graph." (1986).
[6] Bose P, De Carufel JL, Morin P, van Renssen A, Verdonschot S. Towards tight bounds on theta-graphs: More is not always better. Theoretical Computer Science. 2016 Feb 22;616:70-93.
[7] Barba L, Bose P, De Carufel JL, Van Renssen A, Verdonschot S. On the stretch factor of the Theta-4 graph. In Workshop on Algorithms and Data Structures 2013 Aug 12 (pp. 109-120). Springer, Berlin, Heidelberg.
[8] Bonichon N, Gavoille C, Hanusse N, Ilcinkas D. Connections between theta-graphs, Delaunay triangulations, and orthogonal surfaces. InInternational Workshop on Graph-Theoretic Concepts in Computer Science 2010 Jun 28 (pp. 266-278). Springer, Berlin, Heidelberg.
[9] Bose P, Morin P, Van Renssen A, Verdonschot S. The θ5-graph is a spanner. Computational Geometry. 2015 Feb 1;48(2):108-19.
[10] Narasimhan G, Smid M. Geometric spanner networks. Cambridge University Press; 2007 Jan 8.