



Pergamon

Computers Math. Applic. Vol. 31, No. 11, pp. 121–131, 1996

Copyright ©1996 Elsevier Science Ltd

Printed in Great Britain. All rights reserved

0898-1221/96 \$15.00 + 0.00

S0898-1221(96)00069-7

An Average Case Analysis of a Greedy Algorithm for the On-Line Steiner Tree Problem

YING TEH TSAI, CHUAN YI TANG* AND YUNN YEN CHEN

Department of Computer Science, National Tsing Hua University

Hsinchu, 300, Taiwan, R.O.C.

cytang@cs.nthu.edu.tw

(Received May 1995; accepted June 1995)

Abstract—This paper gives the average distance analysis for the Euclidean tree constructed by a simple greedy but efficient algorithm of the on-line Steiner tree problem. The algorithm accepts the data one by one following the order of input sequence. When a point arrives, the algorithm adds the shortest edge, between the new point and the points arriving already, to the previously constructed tree to form a new tree. We first show that, given n points uniformly on a unit disk in the plane, the expected Euclidean distance between a point and its j^{th} ($1 \leq j \leq n-1$) nearest neighbor is less than or equal to $(5/3)\sqrt{j/n}$ when n is large. Based upon this result, we show that the expected length of the tree constructed by the on-line algorithm is not greater than 4.34 times the expected length of the minimum Steiner tree when the number of input points is large.

Keywords—Analysis of algorithms, On-line algorithms, Average case analysis, On-line Steiner tree problems, Euclidean space.

1. INTRODUCTION

Given n points in the Euclidean plane, the minimum Steiner tree problem is to construct a tree which connects the n points and whose Euclidean length is the minimum one. One on-line version of this problem can be described as follows. Suppose that the n points are revealed one by one. When a point is revealed, all the edges between the new point and the old points are also revealed, and we must make a decision to add an edge from the new point to one or more of the old points, so as to have a network connecting all the points seen so far. No edge can be deleted after it has been added and our goal is to minimize the length of the tree we find. We call such a problem *the on-line Steiner tree problem* [1,2].

Imase and Waxman [1] proposed a greedy on-line algorithm, Algorithm *Greedy*, for computing a Euclidean tree. The algorithm is presented as follows:

ALGORITHM *Greedy*

INPUT: $n(\geq 3)$ points v_1, \dots, v_n on Euclidean space

OUTPUT: A Euclidean tree T

Begin

$T = \phi;$

*Author to whom all correspondence should be addressed.

We are most grateful to R. C. T. Lee for his critical comments and encouragement during the preparation of this paper.

```

    input( $v_1$ );
    input( $v_2$ );
    Add the edge between  $v_1$  and  $v_2$  to  $T$ ;
    For  $k = 3$  to  $n$  do
        input( $v_k$ );
        Add the shortest edge between  $v_k$  and  $v_1, \dots, v_{k-1}$  to  $T$ ;
    Endfor
    output( $T$ );
End

```

The algorithm accepts the data one by one in their input sequence. While a new point arrives, we only add the the shortest edge between the new point and the points arriving already to the previous tree to form a new tree. In each examining step, it takes $O(m)$ time, where m is the total number of the points accepted so far.

Here, we are interested in the solutions obtained by on-line algorithms. To analyze the behavior of an on-line algorithm is to compare the solution resulting from this on-line algorithm with the off-line optimal solution. *The competitive analysis* first introduced by Sleator and Tarjan [3] is widely used to evaluate the performance of an on-line algorithm. A deterministic on-line algorithm A is called c -competitive if there is a constant b such that

$$C_A(\sigma) \leq c \cdot C_{\text{opt}}(\sigma) + b,$$

where $C_A(\sigma)$ denotes the cost incurred by the algorithm A with input σ , and $C_{\text{opt}}(\sigma)$ denotes the cost incurred by the off-line optimal algorithm with input σ . We call c the competitive ratio.

Imase and Waxman [1] have shown that the competitive ratio of Algorithm *Greedy* for the on-line Steiner tree problem is $\lceil \log_2 n \rceil$. Alon and Azar [2] showed that the lower bound of competitive ratio of any on-line deterministic algorithm for the on-line Steiner tree problem is $\Omega(\log n / \log \log n)$. In this paper, we focus on the average distance analysis of the tree constructed by Algorithm *Greedy*. In our analysis, we assume that the points are distributed uniformly and independently on a unit disk in the plane. It is shown that, given n points uniformly on a unit disk in the plane, the expected Euclidean distance between a point and its j^{th} ($1 \leq j \leq n-1$) nearest neighbor is less than or equal to $(5/3)\sqrt{j/n}$ when n is large. The main result of this paper is that, under this assumption, the ratio of the expected length of a tree constructed by Algorithm *Greedy* to the expected length of a minimum Steiner tree is not greater than 4.34 when the number of input points is large.

The rest of this paper is organized as follows. Section 2 describes how we analyze the average length of the tree constructed by Algorithm *Greedy* and the analysis is provided in Section 3. In Section 4, we give the further direction of this research.

2. A GLIMPSE OF THE ANALYSIS

In our analysis, we shall assume that the points are uniformly and independently distributed on a unit disk, which is the same assumption in [4,5]. Let X be a random point which is uniformly distributed on a unit disk. Assume that there is a set V of n points in the plane, which is an independent random sample from X . This probabilistic assumption used in the analysis is the same as that in [4,5]. We denote the minimum Steiner tree on the set V and the tree constructed by Algorithm *Greedy* on the set V by T_S and T_{ONLI} , respectively. Let $L(T_S)$ denote the expected length of the minimum Steiner tree T_S , and $L(T_{\text{ONLI}})$ denote the expected length of the tree T_{ONLI} constructed by Algorithm *Greedy*. The goal of the analysis is to compare $L(T_{\text{ONLI}})$ with $L(T_S)$ where the number of points n tends to infinity.

We first discuss a lower bound for $L(T_S)$. Let $L(T_P)$ denote the expected Euclidean length of the minimum spanning tree on the set V . Du and Hwang [6] have shown that on the Euclidean

plane, the length of minimum Steiner tree on a set of points is larger than or equal to $\sqrt{3}/2$ times that of minimum spanning tree on the set of points. Then

$$L(T_S) \geq \frac{\sqrt{3}}{2} L(T_P).$$

A spanning tree which spans n points has $n - 1$ edges. Let a and b be two points. Let $e(a, b)$ be the edge of a minimum spanning tree linking a and b . Here, the length of $e(a, b)$ must be greater than or equal to the distance between a and its nearest neighbor and the distance between b and its nearest neighbor. Thus

$$L(T_P) \geq (n - 1)E(d(n, 1)),$$

where $E(d(n, 1))$ denotes the expected distance of the edge between a point and its nearest neighbor in the set V of n points. So we have

$$L(T_S) \geq \frac{\sqrt{3}(n - 1)}{2} E(d(n, 1)). \tag{1}$$

Now we discuss $L(T_{\text{ONLI}})$, the expected length of a tree produced by Algorithm *Greedy* on the set V . There are $n!$ kinds of input sequences for n input points. Let $S_1, S_2, \dots, S_{n!}$ be the $n!$ distinct sequences of n points. Let $T(S_i)$ denote the tree produced by Algorithm *Greedy* when the input sequence is S_i . If $|T(S_i)|$ is the length of the tree $T(S_i)$, the expected length of a tree produced by Algorithm *Greedy* on the set V is

$$L(T_{\text{ONLI}}) = \frac{\sum_{i=1}^{n!} |T(S_i)|}{n!}.$$

It is obvious that we can hardly find all $|T(S_i)|$'s and sum them up. Fortunately, we have a mechanism to find $\sum_{i=1}^{n!} |T(S_i)|$ through an elegant method. We shall explain our mechanism informally by the following example.

Consider the three points shown in Figure 1. Let $e(a, b)$ be the edge from point a to point b and $|e(a, b)|$ the Euclidean distance of $e(a, b)$. Here are six distinct input sequences, and their corresponding expected lengths of spanning trees produced by Algorithm *Greedy* are shown as follows.

S_i	$ T(S_i) $
1, 2, 3	$ e(2, 1) + e(3, 1) = 3 + 4 = 7$
1, 3, 2	$ e(3, 1) + e(2, 1) = 4 + 3 = 7$
2, 1, 3	$ e(1, 2) + e(3, 1) = 3 + 4 = 7$
2, 3, 1	$ e(3, 2) + e(1, 2) = 5 + 3 = 8$
3, 1, 2	$ e(1, 3) + e(2, 1) = 4 + 3 = 7$
3, 2, 1	$ e(2, 3) + e(1, 2) = 5 + 3 = 8$

The total sum of $|T(S_i)|$ is, therefore, $(7 + 7 + 7 + 8 + 7 + 8) = 44$ and the average length of a spanning tree produced by Algorithm *Greedy* in Figure 1 is $44/6$.

We note that $|e(1, 2)|$, $|e(2, 1)|$, and $|e(3, 1)|$ are the distances between points 1, 2, and 3 and their nearest neighbors, respectively. Furthermore, $|e(1, 3)|$, $|e(2, 3)|$, and $|e(3, 2)|$ are the distances between points 1, 2, and 3 and their second nearest neighbors, respectively. It will be proved later in Section 3 that when we compute the sum of $|T(S_i)|$, the distance between point i , $i = 1, 2$, and 3, and its nearest neighbor appears three times and the distance between point i , $i = 1, 2$, and 3, and its second nearest neighbor appears once. Thus, the total sum of $|T(S_i)|$ is

$$\begin{aligned} & 3 \times (|e(1, 2)| + |e(2, 1)| + |e(3, 1)|) + (|e(1, 3)| + |e(2, 3)| + |e(3, 2)|) \\ & = 3 \times (3 + 3 + 4) + (4 + 5 + 5) = 44. \end{aligned}$$

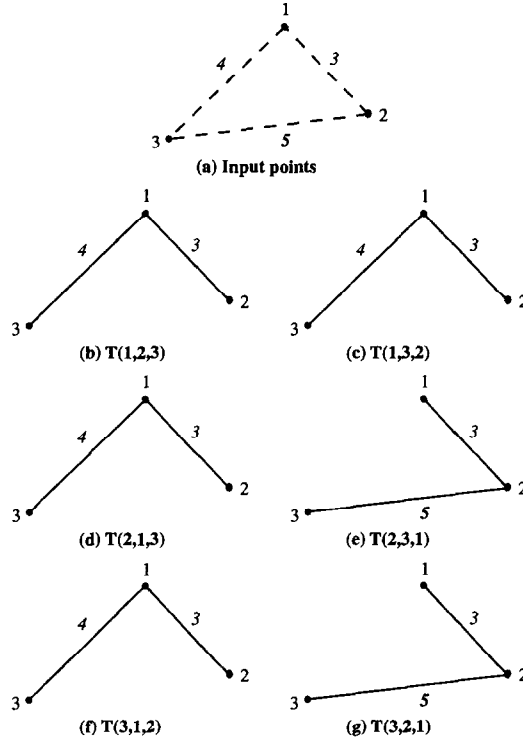


Figure 1. An example for Algorithm *Greedy*.

Let the points in set V be $1, 2, \dots, n$. Let $d_{i,j}$ denote the distance between point i and its j^{th} nearest neighbor. Let $N_{i,j,n}$ denote the number of times that $d_{i,j}$ appears in $\sum_{i=1}^{n!} |T(S_i)|$. Then, obviously,

$$\sum_{i=1}^{n!} |T(S_i)| = \sum_{i=1}^n \sum_{j=1}^{n-1} d_{i,j} N_{i,j,n}.$$

Because we can prove that $N_{1,j,n} = N_{2,j,n} = \dots = N_{n,j,n}$ for all j in the end of this section, we shall have

$$\sum_{i=1}^{n!} |T(S_i)| = \sum_{j=1}^{n-1} N_{1,j,n} \left(\sum_{i=1}^n d_{i,j} \right) = \sum_{j=1}^{n-1} (nN_{1,j,n}) \left(\frac{\sum_{i=1}^n d_{i,j}}{n} \right).$$

In the above formula, $\sum_{i=1}^n d_{i,j}/n$ denotes the expected distance between an arbitrary point and its j^{th} nearest point on the set V , and $(nN_{1,j,n})$ denotes the total number of times that $d_{1,j}, d_{2,j}, \dots$ and $d_{n,j}$ appear in $\sum_{i=1}^{n!} |T(S_i)|$. In the rest of this paper, we use $M_{n,j}$ and $E(d(n, j))$ to denote $N_{1,j,n}$ and $\sum_{i=1}^n d_{i,j}/n$, respectively. Here we have the following formula:

$$L(T_{\text{ONLI}}) = \frac{\sum_{i=1}^{n!} |T(S_i)|}{n!} = \frac{\sum_{j=1}^{n-1} (M_{n,j} E(d(n, j)))}{(n-1)!}. \tag{2}$$

The above discussion indicates that our main job is to find the expected distance between a point and its j^{th} nearest neighbor and the number of times that their distance appears in $\sum_{i=1}^{n!} E(T(S_i))$. Now we show that $N_{1,j,n} = N_{2,j,n} = \dots = N_{n,j,n}$ for all j .

LEMMA 1. *Given points $1, 2, \dots, n$ in the plane, consider the sequence $S_a = \{\tau(1), \tau(2), \dots, \tau(n)\}$, where τ is a permutation on integers $\{1, 2, \dots, n\}$. For every point i and $k, i \neq k$, let $r(i, k)$ denote $\tau(k)$'s rank in the nearest neighbors of $\tau(i)$. For every point $h \neq \tau(i)$, let $S_b = \{\tau'(1), \tau'(2), \dots, \tau'(n)\}$ be a sequence, where τ' is a permutation on integers $\{1, 2, \dots, n\}$ and $\tau' \neq \tau$, such that $\tau'(i) = h$ and $\tau'(k)$ is the $r(i, k)^{\text{th}}$ nearest neighbor of $\tau'(i)$. Then, the number of times*

that $d_{\tau(i),j}$ appears in $|T(S_a)|$ is equal to the number of times that $d_{\tau'(i),j}$ appears in $|T(S_b)|$ for all j . ■

LEMMA 2. For $1 \leq i_1, i_2 \leq n$ and $1 \leq j \leq n - 1$, $N_{i_1,j,n} = N_{i_2,j,n}$.

PROOF. According to Lemma 1, consider i_1 and i_2 where $i_1 \neq i_2$. Supposing that $d_{i_1,j}$ appears in $|T(S_a)|$ x times, then there exists a sequence S_b such that $d_{i_2,j}$ also appears in $|T(S_b)|$ x times, for all j . Therefore, for all $n!$ input sequences, the number of times that $d_{i_1,j}$ appears in $\sum_{i=1}^{n!} |T(S_i)|$ is the same as the number of times that $d_{i_2,j}$ appears in $\sum_{i=1}^{n!} |T(S_i)|$. Thus, $N_{1,j,n} = N_{2,j,n} = \dots = N_{n,j,n}$. ■

In order to get an upper bound of the ratio of $L(T_{\text{ONLI}})$ to $L(T_S)$, we have to know a lower bound of $E(d(n, 1))$, an upper bound of $E(d(n, j))$, and the value of $M_{n,j}$. Then the analysis goes as follows:

1. We derive a lower bound of $E(d(n, 1))$ from [4] and get a lower bound for $L(T_S)$.
2. As shown later, we propose a method to estimate an upper bound of $E(d(n, j))$.
3. We derive $M_{n,j}$ from considering all input sequences.
4. Based upon the results above, we show an upper bound of the ratio of $L(T_{\text{ONLI}})$ to $L(T_S)$ when n is large.

3. THE ANALYSIS

LEMMA 3. $E(d(n, 1)) \geq \sqrt{\pi}/2\sqrt{n}$.

PROOF. Suppose n uniformly and independently distributed points in the d -ball of volume a . By Lemma 1 in [4], the expected length of the nearest neighbor of a random point in the region is larger than or equal to

$$\frac{1}{d} \left(\frac{a}{c_d n} \right)^{1/d} \Gamma \left(\frac{1}{d} \right),$$

where $c_d = \pi^{d/2}/\Gamma(d/2 + 1)$. Then we know that

$$E(d(n, 1)) \geq \frac{1}{2} \left(\frac{\pi}{(\pi/\Gamma(2))n} \right)^{1/2} \Gamma \left(\frac{1}{2} \right) = \frac{\sqrt{\pi}}{2\sqrt{n}}. \quad \blacksquare$$

From Lemma 3 and equation (1), we immediately have a lower bound of $L(T_S)$.

THEOREM 1. $L(T_S) \geq ((n - 1)\sqrt{3\pi})/4\sqrt{n}$. ■

Let X_1, X_2, \dots, X_{n-1} and Y denote n independent points which are uniformly distributed within a unit circle with center $(0, 0)$. Assume that Y is located at $(0, b)$, $0 \leq b \leq 1$. Let Z denote the distance between Y and one of X_i s, $1 \leq i \leq n - 1$. Let $F(z)$ and $f(z)$ denote the cumulative distribution function and probability density function of Z , respectively. The following lemma describes functions $F(z)$ and $f(z)$.

LEMMA 4.

$$F(z) = \begin{cases} z^2, & 0 \leq z \leq 1 - b, \\ \frac{1}{\pi} \left[\arccos \left(\frac{1+b^2-z^2}{2b} \right) + z^2 \arccos \left(\frac{b^2+z^2-1}{2bz} \right) - b \sqrt{1 - \left(\frac{1+b^2-z^2}{2b} \right)^2} \right], & 1 - b < z \leq 1 + b, \\ 1, & z > 1 + b, \end{cases}$$

$$f(z) = \begin{cases} 2z, & 0 \leq z \leq 1 - b, \\ \frac{1}{\pi} \left(2z \arccos \left(\frac{b^2 + z^2 - 1}{2bz} \right) + \frac{z}{b \sqrt{1 - \frac{(1+b^2-z^2)^2}{4b^2}}} \right. \\ \left. - \frac{z(1+b^2-z^2)}{2b \sqrt{1 - \frac{(1+b^2-z^2)^2}{4b^2}}} - \frac{z^2 - b^2 + 1}{2b \sqrt{1 - \frac{(b^2+z^2-1)^2}{4b^2 z^2}}} \right), & 1 - b < z \leq 1 + b, \\ 0, & z > 1 + b. \end{cases}$$

PROOF. Let A denote the area of the intersection of the unit circle and a circle with radius z and center $(0, b)$, where $0 \leq b \leq 1$. Then $F(z) = \Pr(Z \leq z) = A/\pi$, where

$$A = \begin{cases} z^2\pi, & 0 \leq z \leq 1 - b, \\ \arccos\left(\frac{1+b^2-z^2}{2b}\right) + z^2 \arccos\left(\frac{b^2+z^2-1}{2bz}\right) \\ \quad - b\sqrt{1 - \left(\frac{1+b^2-z^2}{2b}\right)^2}, & 1 - b < z \leq 1 + b, \\ \pi, & z > 1 + b. \end{cases}$$

Finally we obtain function $f(z)$ by differentiating $F(z)$ with z . ■

Let $L(n, j)$, $1 \leq j \leq n - 1$, be the expected value of distance between Y and its j^{th} nearest neighbor among X_i s. Then

$$E(d(n, j)) = \int_0^1 \frac{2\pi b}{\pi} L(n, j) db = \int_0^1 2b L(n, j) db. \tag{3}$$

Let $z_i = |X_i - Y|$. Let \bar{z}_1 be the smallest of these z_i , \bar{z}_2 the next z_i in order of magnitude, ..., and \bar{z}_{n-1} the largest z_i . If $g_j(\bar{z}_j)$ is the marginal p.d.f. of \bar{z}_j , then by order statistics [7],

$$g_j(\bar{z}_j) = \begin{cases} \frac{(n-1)!}{(j-1)!(n-1-j)!} [F(\bar{z}_j)]^{j-1} [1 - F(\bar{z}_j)]^{n-1-j} f(\bar{z}_j), & 0 \leq \bar{z}_j \leq 1 + b, \\ 0, & \text{else.} \end{cases}$$

Therefore, we have

$$\begin{aligned} L(n, j) &= \int_0^{1+b} \bar{z}_j g_j(\bar{z}_j) d\bar{z}_j \\ &= \frac{(n-1)!}{(j-1)!(n-1-j)!} \int_0^{1+b} \bar{z}_j [F(\bar{z}_j)]^{j-1} [1 - F(\bar{z}_j)]^{n-1-j} f(\bar{z}_j) d\bar{z}_j \\ &= \frac{(n-1)!}{(j-1)!(n-1-j)!} \int_0^{1+b} z [F(z)]^{j-1} [1 - F(z)]^{n-1-j} f(z) dz. \end{aligned} \tag{4}$$

The theorem below shows an upper bound of $E(d(n, j))$.

THEOREM 2. $\lim_{n \rightarrow \infty} E(d(n, j)) \leq (5/3)\sqrt{j/n}$, $1 \leq j \leq n - 1$.

PROOF. Let $g(i) = \int_0^{1+b} zF(z)^{n-2-i} [1 - F(z)]^i f(z) dz$ and $h(i) = \int_0^{1+b} F(z)^i [1 - F(z)]^{n-1-i} dz$. Then

$$\begin{aligned} g(0) &= \int_0^{1+b} zF(z)^{n-2} f(z) dz \\ &= \frac{1}{n-1} zF(z)^{n-1} \Big|_0^{1+b} - \frac{1}{n-1} \int_0^{1+b} F(z)^{n-1} dz \\ &= \frac{1}{n-1} (1+b - h(0)), \\ g(i) &= \int_0^{1+b} zF(z)^{n-2-i} [1 - F(z)]^i f(z) dz \\ &= \frac{1}{n-1-i} z[1 - F(z)]^i F(z)^{n-1-i} \Big|_0^{1+b} \\ &\quad - \frac{1}{n-1-i} \int_0^{1+b} [[1 - F(z)]^i - iz[1 - F(z)]^{i-1} f(z)] F(z)^{n-1-i} dz \end{aligned}$$

$$\begin{aligned}
 &= \frac{i}{n-1-i} \int_0^{1+b} zF(z)^{n-1-i} [1-F(z)]^{i-1} f(z) dz \\
 &\quad - \frac{1}{n-1-i} \int_0^{1+b} F(z)^{n-1-i} [1-F(z)]^i dz \\
 &= \frac{i}{n-1-i} g(i-1) - \frac{1}{n-1-i} h(i), \quad \text{and} \\
 h(i) &= \int_0^{1+b} F(z)^i [1-F(z)]^{n-1-i} dz \\
 &= \int_0^1 x^i (1-x)^{n-1-i} \frac{1}{f(F^{-1}(x))} dx, \quad (\text{Let } x = F(z)) \\
 &\geq \int_0^1 x^i (1-x)^{n-1-i} dx, \quad (\text{because } 0 \leq f(F^{-1}(x)) \leq 1) \\
 &= \frac{i!(n-1-i)!}{n!}.
 \end{aligned}$$

So we have the following equations:

$$g(0) = \frac{1}{n-1} (1+b-h(0)), \tag{5}$$

$$g(i) = \frac{i}{n-1-i} g(i-1) - \frac{1}{n-1-i} h(i), \tag{6}$$

$$h(i) \geq \frac{i!(n-1-i)!}{n!}. \tag{7}$$

Since

$$\begin{aligned}
 L(n, j) &= \frac{(n-1)!}{(j-1)!(n-j-1)!} \int_0^{1+b} zF(z)^{j-1} [1-F(z)]^{n-j-1} f(z) dz \quad (\text{by equation (4)}) \\
 &= \frac{(n-1)!}{(j-1)!(n-j-1)!} g(n-1-j) \\
 &= \frac{(n-1)!}{(j-1)!(n-j-1)!} \left(\frac{n-1-j}{j} \times \frac{n-2-j}{j+1} \times \dots \times \frac{2}{n-3} \right. \\
 &\quad \times \frac{1}{n-2} \times \frac{1}{n-1} (1+b-h(0)) \\
 &\quad - \frac{n-1-j}{j} \times \frac{n-2-j}{j+1} \times \dots \times \frac{2}{n-3} \times \frac{1}{n-2} h(1) \\
 &\quad - \frac{n-1-j}{j} \times \frac{n-2-j}{j+1} \times \dots \times \frac{3}{n-4} \times \frac{1}{n-3} h(2) \\
 &\quad \vdots \\
 &\quad \left. - \frac{n-1-j}{j} \times \frac{1}{j+1} h(n-2-j) \right) \quad (\text{by equations (5),(6)}) \\
 &= 1+b-h(0) - (n-1)h(1) - \frac{(n-1)(n-2)}{2} h(2) - \frac{(n-1)(n-2)(n-3)}{3 \times 2 \times 1} h(3) \\
 &\quad - \dots - \frac{(n-1)!}{(j+1)!(n-2-j)!} h(n-2-j) - \frac{(n-1)!}{j!(n-1-j)!} h(n-1-j) \\
 &= 1+b - \sum_{i=0}^{n-1-j} \frac{(n-1)!}{i!(n-1-i)!} h(i),
 \end{aligned}$$

we can rewrite $E(d(n, j))$ as follows:

$$\begin{aligned} E(d(n, j)) &= \int_0^1 2bL(n, j) db \quad (\text{by equation (3)}) \\ &= \int_0^1 2b \left(1 + b - \sum_{i=0}^{n-j-1} \frac{(n-1)!}{i!(n-1-i)!} h(i) \right) db \\ &= \frac{5}{3} - \int_0^1 2b \left(\sum_{i=0}^{n-j-1} \frac{(n-1)!}{i!(n-1-i)!} h(i) \right) db \\ &= \frac{5}{3} \left(1 - \frac{3}{5} \int_0^1 2b \left(\sum_{i=0}^{n-j-1} \frac{(n-1)!}{i!(n-1-i)!} h(i) \right) db \right). \end{aligned}$$

Note that

$$\frac{5}{3} \sqrt{\frac{j}{n}} = \frac{5}{3} \left(1 - \frac{n-j}{n + \sqrt{nj}} \right).$$

Therefore,

$$\begin{aligned} \frac{5}{3} \sqrt{\frac{j}{n}} - E(d(n, j)) &= \frac{3}{5} \int_0^1 2b \left(\sum_{i=0}^{n-j-1} \frac{(n-1)!}{i!(n-1-i)!} h(i) \right) db - \frac{n-j}{n + \sqrt{nj}} \\ &\geq \frac{3}{5} \int_0^1 2b \times \frac{n-j}{n} db - \frac{n-j}{n + \sqrt{nj}} \quad (\text{by equation (7)}) \\ &= \frac{3}{5} \times \frac{n-j}{n} - \frac{n-j}{n + \sqrt{nj}} \\ &= \frac{n-j}{5} \left(\frac{3\sqrt{j} - 2\sqrt{n}}{\sqrt{n}(n + \sqrt{nj})} \right). \end{aligned}$$

If

$$\frac{4n}{9} \leq j \leq n-1, \quad \frac{3\sqrt{j} - 2\sqrt{n}}{\sqrt{n}(n + \sqrt{nj})} \geq 0 \quad \text{and} \quad E(d(n, j)) \leq \frac{5}{3} \sqrt{\frac{j}{n}}.$$

For $1 \leq j < 4n/9$,

$$\frac{3\sqrt{j} - 2\sqrt{n}}{\sqrt{n}(n + \sqrt{nj})} \geq \frac{3 - 2\sqrt{n}}{\sqrt{n}(n + \sqrt{n})}.$$

Since $\lim_{n \rightarrow \infty} (3 - 2\sqrt{n})/(\sqrt{n}(n + \sqrt{n})) = 0$, we have $\lim_{n \rightarrow \infty} E(d(n, j)) \leq (5/3)\sqrt{j/n}$ when $1 \leq j < 4n/9$. Thus, we conclude that

$$\lim_{n \rightarrow \infty} E(d(n, j)) \leq \frac{5}{3} \sqrt{\frac{j}{n}}, \quad 1 \leq j \leq n-1. \quad \blacksquare$$

For $x, y > 0$, let $P_x^y = y!/x!$ if $y \geq x$, 0 otherwise. We give the formula of $M_{n,j}$ here.

LEMMA 5.

$$M_{n,j} = (n-2)! + \sum_{i=2}^{n-j} (i \cdot P_{j-1}^{n-i-1} \cdot (n-j-1)!), \quad 1 \leq j \leq n-1.$$

PROOF. Let point 1 be an arbitrary point of the n points and point i , $2 \leq i \leq n$, be the $(i-1)^{\text{st}}$ nearest point of point 1. Consider the $n!$ distinct cases. Now we evaluate in how many cases that edge $e(1, j+1)$ is added to T_{ONLI} in the $n!$ input sequences. For all cases that the point 1 is the first input point, no edge will be added to T_{ONLI} . If the point 1 is given in the second place, there are $(n-2)!$ cases that the point $j+1$ is the first input point and edge $e(1, j+1)$ will be

added to T_{ONLI} . If the point 1 is given in the i^{th} ($3 \leq i \leq n$) place, edge $e(1, j + 1)$ is added to T_{ONLI} only if point 1 is not given after the $(n - j + 1)^{\text{st}}$ input point arrives. Edge $e(1, j + 1)$ will be added to T_{ONLI}^m if and only if the following conditions are satisfied:

1. Point $j + 1$ is one of the first $i - 1$ input points.
2. Point x , $2 \leq x \leq j$, is not one of the first $i - 1$ input points; that is, point x should be given after the i^{th} input point arrives. If not, the edge whose distance is smaller than that of edge $e(1, j + 1)$ will be added to T_{ONLI}^m .
3. Other $n - j - 1$ points are present in the remaining $n - j - 1$ places.

Conditions (1)–(3) have $i - 1$, P_{j-1}^{n-i} and $(n - j - 1)!$ cases, respectively. Then, the number of times that edge $e(1, j + 1)$ is added to T_{ONLI}^m while point 1 is the i^{th} ($3 \leq i \leq n - j + 1$) input point is

$$(i - 1) \cdot P_{j-1}^{n-i} \cdot (n - j - 1)!.$$

So, for $1 \leq j \leq n - 1$, we have

$$\begin{aligned} M_{n,j} &= (n - 2)! + \sum_{i=3}^{n-j+1} ((i - 1) \cdot P_{j-1}^{n-i} \cdot (n - j - 1)!) \\ &= (n - 2)! + \sum_{i=2}^{n-j} (i \cdot P_{j-1}^{n-i-1} \cdot (n - j - 1)!). \end{aligned}$$

Here, we prove two expressions to be used in the main theorem.

LEMMA 6.

(1)

$$\frac{M_{n,j}}{(n - 1)!} = \frac{n}{j(j + 1)}.$$

(2)

$$\sum_{j=1}^{n-1} \frac{n}{j(j + 1)} \sqrt{\frac{j}{n}} \leq 2\sqrt{n} \left(1 - \frac{1}{\sqrt{n - 1}}\right).$$

PROOF.

(1)

$$\begin{aligned} \frac{M_{n,j}}{(n - 1)!} &= \frac{1}{n - 1} + \frac{(n - j - 1)!}{(n - 1)!} \sum_{i=2}^{n-j} \frac{i(n - i - 1)!}{(n - i - j)!} \quad (\text{by Lemma 5}) \\ &= \frac{1}{n - 1} + \frac{(n - j - 1)!j!}{j(n - 1)!} \sum_{i=2}^{n-j} \frac{i(n - i - 1)!}{(j - 1)!(n - i - j)!} \\ &= \frac{1}{n - 1} + \frac{1}{j \binom{n-1}{j}} \sum_{i=2}^{n-j} \left[i \binom{n - i - 1}{j - 1} \right]. \end{aligned}$$

We have

$$\begin{aligned} \sum_{i=2}^{n-j} \left[i \binom{n - i - 1}{j - 1} \right] &= 2 \binom{n - 3}{j - 1} + 3 \binom{n - 4}{j - 1} + \dots + (n - j) \binom{j - 1}{j - 1} \\ &= 2 \left[\binom{n - 3}{j - 1} + \binom{n - 4}{j - 1} + \dots + \binom{j - 1}{j - 1} \right] \\ &\quad + \left[\binom{n - 4}{j - 1} + \binom{n - 5}{j - 1} + \dots + \binom{j - 1}{j - 1} \right] \\ &\quad + \dots + \left[\binom{j}{j - 1} + \binom{j - 1}{j - 1} \right] + \binom{j - 1}{j - 1} \end{aligned}$$

$$\begin{aligned}
 &= 2 \sum_{k=0}^{n-j-2} \binom{j-1+k}{k} + \sum_{k=0}^{n-j-3} \binom{j-1+k}{k} \\
 &\quad + \dots + \sum_{k=0}^1 \binom{j-1+k}{k} + \sum_{k=0}^0 \binom{j-1+k}{k} \\
 &\quad \left(\text{by } \sum_{k=0}^n \binom{m+k}{k} = \binom{m+n+1}{n} \right) \\
 &= 2 \binom{n-2}{n-j-2} + \binom{n-3}{n-j-3} + \dots + \binom{j+1}{1} + \binom{j}{0} \\
 &= \binom{n-2}{j} + \sum_{k=0}^{n-j-2} \binom{j+k}{k} \\
 &= \binom{n-2}{j} + \binom{n-1}{n-j-2} \\
 &\quad \left(\text{by } \sum_{k=0}^n \binom{m+k}{k} = \binom{m+n+1}{n} \right) \\
 &= \binom{n-2}{j} + \binom{n-1}{j+1}.
 \end{aligned}$$

After replacing $\sum_{i=2}^{n-j} [i \binom{n-i-1}{j-1}]$ by $\binom{n-2}{j} + \binom{n-1}{j+1}$, we obtain

$$\begin{aligned}
 \frac{M_{n,j}}{(n-1)!} &= \frac{1}{n-1} + \frac{1}{j \binom{n-1}{j}} \left[\binom{n-2}{j} + \binom{n-1}{j+1} \right] \\
 &= \frac{1}{n-1} + \frac{n-1-j}{j(n-1)} + \frac{n-1-j}{j(j+1)} = \frac{n}{j(j+1)}.
 \end{aligned}$$

$$\begin{aligned}
 (2) \quad \sum_{j=1}^{n-1} \frac{n}{j(j+1)} \sqrt{\frac{j}{n}} &= \sqrt{n} \left(\frac{1}{2} + \frac{1}{3\sqrt{2}} + \frac{1}{4\sqrt{3}} + \frac{1}{5 \times 2} + \sum_{j=5}^{n-1} \frac{1}{\sqrt{j}(j+1)} \right) \\
 &\leq \sqrt{n} \left(1 + \sum_{j=5}^{n-1} j^{-3/2} \right) \\
 &\leq \sqrt{n} \left(1 + \int_4^{n-1} j^{-3/2} dj \right) \\
 &= \sqrt{n} \left(1 + 2 \left(\frac{1}{2} - \frac{1}{\sqrt{n-1}} \right) \right) \\
 &= 2\sqrt{n} \left(1 - \frac{1}{\sqrt{n-1}} \right).
 \end{aligned}$$

We now prove the main theorem.

THEOREM 3. *The limit ratio of $L(T_{\text{ONLI}})$ to $L(T_{\text{MIN}})$ is 4.34.*

PROOF. We derive an upper bound of $L(T_{\text{ONLI}})$ as follows:

$$\begin{aligned}
 \lim_{n \rightarrow \infty} L(T_{\text{ONLI}}) &= \lim_{n \rightarrow \infty} \sum_{j=1}^{n-1} \frac{M_{n,j}^{n-1}}{(n-1)!} E(d(n, j)) \quad (\text{by equation (2)}) \\
 &= \lim_{n \rightarrow \infty} \sum_{j=1}^{n-1} \frac{n}{j(j+1)} E(d(n, j)) \quad (\text{by Lemma 6 (1)})
 \end{aligned}$$

$$\begin{aligned}
&\leq \lim_{n \rightarrow \infty} \frac{5}{3} \sum_{j=1}^{n-1} \frac{n}{j(j+1)} \sqrt{\frac{j}{n}} && \text{(by Theorem 2)} \\
&\leq \lim_{n \rightarrow \infty} \frac{10}{3} \sqrt{n} \left(1 - \frac{1}{\sqrt{n-1}}\right) && \text{(by Lemma 6 (2))} \\
&= \frac{10}{3} \sqrt{n}.
\end{aligned}$$

From Theorem 1, we know that $\lim_{n \rightarrow \infty} L(T_S) \geq (\sqrt{3n\pi})/4$. Finally, we get

$$\lim_{n \rightarrow \infty} \frac{L(T_{\text{ONLI}})}{L(T_S)} \leq \frac{(10/3)\sqrt{n}}{\sqrt{3n\pi}/4} \approx 4.34. \quad \blacksquare$$

4. CONCLUDING REMARKS

In this paper, we discuss the average length of the Euclidean tree constructed by Algorithm *Greedy* when the number of input points tends to infinity. The ratio of the expected length of the tree constructed by Algorithm *Greedy* to the expected length of the minimum Steiner tree is not greater than 4.34 when the number of input points is large. Besides, we propose a method to derive an upper bound of the expected distance between a point and its j^{th} nearest neighbor, which points are distributed uniformly and independently on a unit disk in the plane. The result of this paper can be improved if a better upper bound for $E(d(n, j))$ or a better lower bound for $L(T_S)$ is found. The average competitive ratio of Algorithm *Greedy* for the on-line Steiner tree problem has been shown to be a constant, but the exact value of average ratio is still an open problem.

REFERENCES

1. M. Imase and B.M. Waxman, Dynamic Steiner tree problem, *SIAM J. Disc. Math.* **4** (3), 369–384 (1991).
2. N. Alon and Y. Azar, On-line Steiner trees in the Euclidean plane, Presented at the 8^{th} Annual Symposium on Computational Geometry, (1992).
3. D.D. Sleator and R.E. Tarjan, Amortized efficiency of list update and paging rules, *Commun. ACM* **28** (2), 202–208 (1985).
4. D.J. Bertsimas and G.V. Ryzin, An asymptotic determination of the minimum spanning tree and minimum constants in geometrical probability, *Operations Research Letters* **9**, 223–231 (1990).
5. M.-Y.B. Wang and R.-C. Chang, An upper bound for the average length of the Euclidean minimum spanning tree, *Intern. J. Computer Math.* **30**, 1–12 (1989).
6. D.Z. Du and F.K. Hwang, A proof of the Gilbert-Pollak conjecture on the Steiner ratio, *Algorithmica* **7**, 121–135 (1992).
7. R.V. Hogg and A.T. Craig, *Introduction to Mathematical Statistics*, Macmillan, (1978).