# Planar L-Shaped Point Set Embeddings of Trees*

Oswin Aichholzer† Thomas Hackl† Manfred Scheucher†


## Abstract

In this paper we consider planar L-shaped embeddings of trees in point sets, that is, planar drawings where the vertices are mapped to a subset of the given points and where every edge consists of two axis-aligned line segments. We investigate the minimum number $m$, such that any $n$ vertex tree with maximum degree 4 admits a planar L-shaped embedding in any point set of size $m$.

First we give an upper bound $O(n^c)$ with $c = \log_2 3 \approx 1.585$ for the general case, and thus answer the question by Di Giacomo et al. [4] whether a subquadratic upper bound exists.

Then we introduce the saturation function for trees and show that trees with low saturation can be embedded even more efficiently. In particular, we improve the upper bound for caterpillars and extend the class of trees that require only a linear number of points. In addition, we present some probabilistic results for either randomly chosen trees or randomly chosen point sets.


## 1 Introduction

A *point set embedding* of a given graph $G$ in a given point set $P$ is a drawing where all vertices are drawn as points of $P$. In general, the decision problem whether a graph admits a planar straight line point set embedding in a given point set is NP-complete [3], while for trees and outerplanar graphs there are efficient embedding algorithms; see for example [2]. Kaufmann and Wiese [7] have investigated a relaxation of this problem, namely point set embeddings where edges can be drawn as polylines. They proved that the decision problem remains NP-complete if at most one bend per edge is allowed, and that any planar graph admits a planar point set embedding with at most 2 bends per edge. Katz et al. [6] introduced *orthogeodesic* point set embeddings, i.e., drawings where edges have minimal $L^1$-length and are drawn as unions of axis parallel line segments. They proved that deciding whether a graph admits a planar orthogeodesic point set embedding is NP-complete. Di Giacomo et al. [4] introduced *L-shaped* point set embeddings, i.e., orthogeodesic point set embeddings where every edge has at most one bend, and investigated orthogeodesic and L-shaped point set embeddings of trees; in particular *caterpillars*, i.e., trees where the removal of all leaves results in a path.

As in [4, 6], throughout this paper we assume that every two points in any set of $m$ points have distinct $x$- and distinct $y$-coordinates. Moreover, we will only consider planar L-shaped embeddings, and thus we can further assume that for every set $P$ of $m$ points, the $x$- and $y$-coordinates of every point in $P$ are in $\{1, \dots, m\}$, i.e., that $P = \{(i, \pi(i))\}_{i=1}^m$ holds for a permutation $\pi : \{1, \dots, m\} \to \{1, \dots, m\}$.

For a tree $T$, let $f(T)$ be the minimum number $m$ such that $T$ admits a planar L-shaped embedding in any point set of size $m$, and let further $f_d(n) := \max f(T)$ where the maximum is over all trees $T$ on $n$ vertices with maximum degree at most $d$. Point sets admitting an embedding (of a certain type) of every $n$ vertex graph (of a certain class) are referred to as *universal point sets*, see for example [4]. Obviously, only trees with maximum degree at most 4 admit planar L-shaped embeddings, and moreover, any $n$ vertex path admits a trivial embedding in every set of $n$ points. Therefore, only trees with maximum degree 3 or 4 are of interest. The previously best known upper bound on $f_4(n)$ is quadratic [4]. The best lower bound so far is $f_d(n) \geq n$ for $d = 3, 4$.

In Section 2, we prove $f_4(n) = O(n^{\log_2 3})$, using a recursive embedding algorithm, and hence answer the question stated by Di Giacomo et al. [4] whether a subquadratic upper bound on $f_4(n)$ exists.

In Section 3, we introduce the *saturation* function $\sigma$ for trees and prove that $f(T) \leq 2^{\sigma(T)} n$ holds for any $n$ vertex tree $T$. For trees with saturation bounded by a constant this clearly gives a linear upper bound, which enlarges the set of graphs that can be embedded in point sets of linear size. In particular, for caterpillars we improve the upper bound $f(T) \leq 3n - 2$ provided in [4] to $2n$, which can be further improved to $(4/3 + \varepsilon)n + O(1)$ for $\varepsilon > 0$. We further show that $f(T) = O(n^{1.5+\varepsilon})$ holds with probability at least $\frac{2\varepsilon}{1+2\varepsilon}$ if the tree $T$ is chosen uniformly at random among all rooted $n$ vertex trees with maximum degree at most 4.

In Section 4, we show that a given $n$-vertex com-

*This work is based on the Master's thesis of Manfred Scheucher [8]. Available from http://www.ist.tugraz.at/scheucher/publ/masters_thesis_2015.pdf. Partially supported by the ESF EUROCORES programme EuroGIGA – CRP ComPoSe, Austrian Science Fund (FWF): I648-N18 and FWF project P23629-N18 'Combinatorial Problems on Geometric Graphs'.

†Institute of Software Technology, Graz University of Technology, {oaich,thackl,mscheuch}@ist.tugraz.at

plete binary tree $T$ can be embedded in a point set $P$ with probability at least $1/2$, if $P$ was chosen uniformly at random among all point sets of size $O(n \log^2 n)$. A generalization to arbitrary trees, including an improved bound on the size of the point sets, can be found in [8]. Even though the question by Di Giacomo et al. [4], whether a linear upper bound on $f_3(n)$ exists, remains open, these results give more insight into the problem.

## 2 A Subquadratic Bound for the General Case

For this section, we define the integer sequence $(u_n)_{n\in\mathbb{N}_0}$ recursively by

$$u_0 := 0,\ u_{n+1} := \max_{\substack{a\geq b\geq c\\ a+b+c=n}} 1 + u_a + 2u_b + 2u_c,$$

where $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$ denotes the set of non-negative integers. We use a recursive approach to find a planar L-shaped embedding of a given tree $T$ with $n$ vertices and maximum degree at most 4 in a given point set $P$ of size $u_n$. In the recursive step subtrees of sizes $a$, $b$, and $c$ will be embedded in sub point sets of sizes $u_a$, $u_b$, and $u_c$, respectively. As $u_n \leq n^{\log_2 3}$ holds for every $n$, this will give a proof for $f_4(n) \leq n^{\log_2 3}$.

**Lemma 1** *$u_n \leq n^{\log_2 3}$ holds for any $n \in \mathbb{N}_0$.*

**Proof.** Let $g(x) = x^{\log_2 3}$ on $[0, \infty)$. We give a proof by induction that $u_n \leq g(n)$ holds for $n \in \mathbb{N}_0$.

For the induction base, $u_0 = 0$ and $u_1 = 1$ clearly fulfill the inequality. For the induction step, let $n \geq 1$. We assume that $u_k \leq g(k)$ holds for any $k \leq n$, and prove that $u_{n+1} \leq g(n+1)$ holds. Let

$$S = \{(x, y, z) \in \mathbb{R}^3 \colon x \geq y \geq z \geq 0,\ x + y + z = n\}.$$

By definition, $u_{n+1} = 1 + u_a + 2u_b + 2u_c$ holds for some integers $a \geq b \geq c \geq 0$ with $a + b + c = n$. By the induction assumption, we can write

$$u_{n+1} \leq 1 + g(a) + 2g(b) + 2g(c),$$

and since $(a, b, c) \in S$,

$$u_{n+1} \leq \max_{(x,y,z)\in S} \underbrace{1 + g(x) + 2g(y) + 2g(z)}_{=h(x,y,z)}.$$

As $g$ is a convex function on $[0, \infty)$, $h$ is a convex function on $S$. Moreover, $S$ is a convex set since $S$ is spanned by $s_1 = (n, 0, 0)$, $s_2 = (\frac{n}{2}, \frac{n}{2}, 0)$, and $s_3 = (\frac{n}{3}, \frac{n}{3}, \frac{n}{3})$; a proof can be found in [8, Lemma 10].

According to the Maximum Principle, $h$ attains its maximum over $S$ in $s_1$, $s_2$, or $s_3$. We now show that $h(s_i) \leq g(n+1)$ holds for $i = 1, 2, 3$:

$s_1$: Due to the Mean Value Theorem it holds that $g(n+1) - g(n) = g'(\xi)$ for some $\xi \in (n, n+1)$. Since $\log_2 3 > 1$ and $1 \leq \xi$, we have $g'(\xi) = (\log_2 3)\xi^{\log_2 3 - 1} \geq 1$, and thus $h(s_1) = 1 + g(n) \leq g(n+1)$.

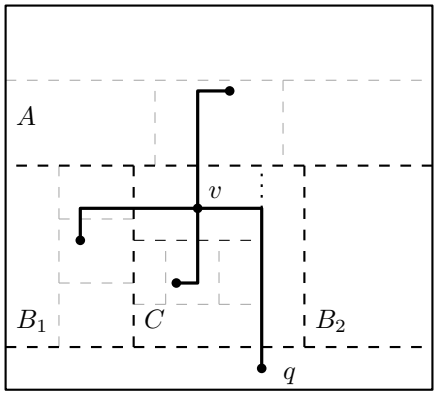


Figure 1: The recursive embedding: lines partitioning $P$ are drawn dashed black; the dotted line illustrates how to choose $C'$; lines partitioning $A$, $B$, and $C$ in the next recursion are drawn dashed gray. Only points used in the first and second recursive step are shown.

$s_2$: $h(s_2) = 1 + 3(n/2)^{\log_2 3} = 1 + n^{\log_2 3} = 1 + g(n) \leq g(n+1)$.

$s_3$: $h(s_3) = 1 + 5(n/3)^{\log_2 3} \leq 1 + g(n) \leq g(n+1)$ holds, since $5/3^{\log_2 3} < 1$.

As a consequence, we have $u_{n+1} \leq g(n+1)$. □

From the definition of $u_n$, we derive the following algorithm:

1. Let $T$ be an $n$ vertex tree with maximum degree at most 4, rooted at some degree-1-vertex $r$, and let $P$ be a set of $u_n$ points.
2. We place $r$ at the bottommost point $q$ of $P$.
3. Let $v$ be the child of $r$, and let $T_A, T_B, T_C$ be the subtrees of $v$ of sizes $a$, $b$, and $c$ respectively, with $a \geq b \geq c \geq 0$ and $a + b + c + 1 = n - 1$.
4. Recall that $u_n \geq u_{n-1} + 1$ holds by definition of $u_n$. We partition $P = A \cup (B_1 \cup C \cup B_2) \cup \{q\}$ such that $A$ contains the topmost $u_a$ points, $q$ is the bottommost point, $|B_1| = |B_2| = u_b$, $|C| = 2u_c + 1$, $B_1$ is on the left of $C$, and $C$ is on the left of $B_2$. Figure 1 gives an illustration.
5. At least $u_c+1$ points in $C$ have $x$-coordinates less (greater) than $q$. We denote this set as $C'$. We embed $v$ as the topmost vertex $q'$ in $C'$. As long as not all subtrees are empty, we continue recursively by embedding $T_A$ in $A$, $T_B$ in $B_1$ (resp., $B_2$), and $T_C$ in $C' \backslash \{q'\}$. The subtrees are embedded with respect to an according rotation as illustrated in Figure 1.

Together with Lemma 1 we get the following:

**Theorem 2** *$f_4(n) \leq n^{\log_2 3}$.*

For further improvements on the multiplicative factor of the $n^{\log_2 3}$ term we refer to [8, Chapter 3], where $f_3(n) \leq 0.5n^{\log_2 3} + O(n)$ and $f_4(n) \leq cn^{\log_2 3} + O(n)$ with $c \approx 0.508$ have been shown.

## 3 Bounds for Special Cases

In this section, we introduce the saturation function $\sigma$ for trees and show how to handle trees with low saturation in a more efficient way.

**Definition 1 (Saturation)** *Let $T = (V, E)$ be a tree. For the rooted version $T^r$ of $T$ with root $r \in V$, we define $\sigma_r : V \to \mathbb{N}_0$ recursively, such that*

$$\sigma_r(v) = \max\{0, \sigma_r(u_1), \sigma_r(u_2) + 1, \ldots, \sigma_r(u_k) + 1\}$$

*holds for every vertex $v$ with children $u_1, \ldots, u_k$ ($k \geq 0$) and $\sigma_r(u_1) \geq \ldots \geq \sigma_r(u_k)$. We define the* rooted saturation $\sigma(T^r) := \sigma_r(r) = \max_{v \in V} \sigma_r(v)$. *For the unrooted tree $T$, we define the* saturation $\sigma(T) := \min_{r \in V} \sigma(T^r)$.

**Theorem 3** *$f(T) \leq 2^{\sigma(T)}n$ holds for every tree $T$ with $n$ vertices and maximum degree at most 4.*

**Proof.** We slightly modify the algorithm proposed in Section 2. For the recursion, we embed a subtree with maximum saturation in the top area instead of the largest subtree. By definition of $\sigma(T)$, $2^{\sigma(T)}$ copies of every point are sufficient for the algorithm to succeed, and thus, the statement follows. □

We remark that it is straightforward to show that $\sigma(T) \leq \log_2(n+1) - 1$ holds for every $n$ vertex tree $T$ (see [8, Chapter 3.3.2]), which directly gives an alternative proof for $f_4(n) = O(n^2)$.

Di Giacomo et al. [4] have already proven that, for caterpillars with maximum degree at most 4, any point set of size $3n - 2$ is sufficient to find a planar L-shaped embedding. Since caterpillars have saturation at most 1, $2n$ is an upper bound. Moreover, this upper bound can be improved to $(4/3 + \varepsilon)n + O(1)$ for any fixed $\varepsilon > 0$; we refer to [8, Chapter 5.2].

### 3.1 Probabilistic Analysis

In this subsection, we make use of the *register function* $\rho$ (see e.g. Auber et al. [1]) to handle the saturation function $\sigma$: $\rho(T)$ is defined analogously to the saturation $\sigma(T)$, where the expression $\max\{0, \sigma_r(u_1), \sigma_r(u_2)+1, \sigma_r(u_3)+1, \ldots, \sigma_r(u_k)+1\}$ is changed to $\max\{0, \sigma_r(u_1), \sigma_r(u_2) + 1, \sigma_r(u_3) + 2, \ldots, \sigma_r(u_k)+k-1\}$. By definition, the register function gives an upper bound on the saturation function.

For $n \in \mathbb{N}$ let $T_n$ denote the set of rooted trees with $n$ vertices and maximum degree at most 4. If we suppose that every rooted tree in $T_n$ is equally likely, then $T_n$ can be interpreted as a random variable.

Drmota and Prodinger [5] have shown that the expected value of the random variable $\rho(T_n)$ fulfills $\mathbb{E}(\rho(T_n)) = \log_4 n + O(1)$. It is straightforward to deduce that a constant $c \in \mathbb{R}$ exists such that $\mathbb{E}(\sigma(T_n)) \leq \log_4 n + c$ holds for every $n$.

**Theorem 4** *Let $\varepsilon > 0$. Then the probability $\mathbb{P}[f(T_n) = O(n^{1.5+\varepsilon})]$ is at least $p = \frac{2\varepsilon}{1+2\varepsilon} > 0$.*

**Proof.** Let $s_n = \mathbb{E}(\sigma(T_n))$. According to Markov's inequality we have $\mathbb{P}[\sigma(T_n) \geq s_n(1 + 2\varepsilon)] \leq \frac{1}{1+2\varepsilon}$, or equivalently, $\mathbb{P}[\sigma(T_n) \leq s_n(1 + 2\varepsilon)] \geq p$.

Since $s_n \leq \log_4 + c_1$ holds for a constant $c_1$, $2^{\sigma(T_n)} \leq 2^{s_n(1+2\varepsilon)} \leq 2^{(\log_4 n + c_1)(1+2\varepsilon)} = c_2 n^{0.5+\varepsilon}$ holds with probability at least $p$, where $c_2 = 2^{c_1(1+2\varepsilon)}$. The statement follows from Theorem 3. □

## 4 Probabilistic Approach for Trees

For a tree $T$ let $f^{1/2}(T)$ be the minimum number $m$ such that $T$ admits a planar L-shaped embedding in at least half of all point sets of size $m$. Furthermore, let $f_d^{1/2}(n) := \max f^{1/2}(T)$ where the maximum is over all trees $T$ on $n$ vertices with maximum degree at most $d$.

### 4.1 Complete $k$-Ary Trees

We prove that $f^{1/2}(T) \leq 2(n+1)\log^2(n+1)$ holds if $T$ is a complete binary tree on $n$ vertices. To do so, we consider the following algorithm:

1. Let $T$ be a complete binary tree on $n$ vertices, rooted at $r$, and let $P$ be a set of $2(n+1)\log_2^2(n+1)$ points.

2. Since $T$ is complete, $\tilde{n} := n + 1 = 2^h$ holds with $h := \log_2 \tilde{n} \in \mathbb{N}$. We define $\alpha := 2h$ and write $|P| = \alpha\tilde{n}\log_2 \tilde{n}$.

3. We partition $P = (A \cup B_1 \cup B_2) \cup C$ such that $|A| = |C| = \alpha(\frac{\tilde{n}}{2})$, $|B_1| = |B_2| = \alpha(\frac{\tilde{n}}{2})\log_2(\frac{\tilde{n}}{2})$, $(A \cup B_1 \cup B_2)$ is above $C$, $A$ is to the left of $B_1$, and $B_1$ is to the left of $B_2$. Furthermore, let $B = B_1 \cup B_2$. Figure 2 gives an illustration.

4. If there exists a *candidate point* $q$, i.e., a point in $C$ which is to the left of $B$, we place $r$ in $q$ and continue recursively by embedding the two subtrees in $B_1$ and $B_2$, respectively. Otherwise, no solution is found and the algorithm stops.

If there exists a candidate point in every recursion (step 4), then the algorithm clearly admits a planar L-shaped embedding; one only needs to draw the edges as depicted in Figure 2 after all points have been placed.

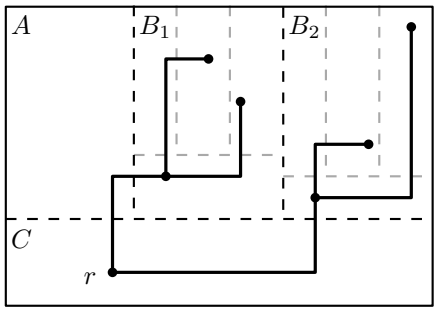


Figure 2: Embedding of a complete binary tree: black dashed lines illustrate the partition of $P$ and gray dashed lines illustrate the partition of $B_1$ and $B_2$ in the next recursion, respectively.

It remains to show that all desired candidate points exist with probability at least 1/2.

The probability that a candidate point exists is exactly $p := 1 - \prod_{i=1}^{|C|} \left(1 - \frac{|A|+1}{|A|+|B|+i}\right)$, because assuming that neither of the points $c_1, \ldots, c_{i-1}$ is placed on the left of $B$, there are $|A|+|B|+i$ positions in which $c_i$ can be placed, and $|A|+1$ of which are to the left of $B$. Since $i \leq |C|$ and by the partition of $P$, we have

$$p \geq 1 - \left(1 - \frac{|A|}{|P|}\right)^{|C|} = 1 - \left(1 - \frac{1}{2\log_2 \tilde{n}}\right)^{\alpha\frac{\tilde{n}}{2}} =: \tilde{p},$$

which we can also write as

$$\tilde{p} = 1 - \left(\left(1 - \frac{1}{2\log_2 \tilde{n}}\right)^{2\log_2 \tilde{n}}\right)^{\alpha\frac{\tilde{n}}{4\log_2 \tilde{n}}}.$$

Recall that the function $g(x) = (1 - \frac{1}{x})^x$ on $[2, \infty)$ fulfills $\frac{1}{4} \leq g(x) \leq \frac{1}{e}$ with $e$ being Euler's number, and that the function $h(x) = \frac{x}{\ln x}$ on $(1, \infty)$ has its minimum at $x = e$ with $h(e) = e$. As a consequence $\frac{\tilde{n}}{4\log_2 \tilde{n}} \geq \frac{e \ln 2}{4} \geq \frac{\ln 2}{2}$ holds, and thus we can bound $p \geq \tilde{p} \geq 1 - (1/e)^{(\alpha \ln 2)/2} = 1 - (1/2)^{\alpha/2}$.

Obviously, this lower bound on $p$ does not depend on the recursion level but only on $\alpha$, which was chosen depending only on the initial number of points.

Since $T$ is a complete binary tree on $n = \tilde{n} - 1$ vertices, $T$ has $\frac{\tilde{n}}{2} - 1$ inner vertices. Furthermore, since $(1/2)^{\alpha/2} = 1/\tilde{n}$ holds by definition of $\alpha$, the probability for the algorithm to succeed is at least $(1 - 1/\tilde{n})^{\tilde{n}/2-1} \geq (1 - 1/\tilde{n})^{\tilde{n}/2} \geq (1/4)^{1/2} = 1/2$.

This gives a proof of the following:

**Theorem 5** *$f^{1/2}(T) \leq 2(n+1)\log^2(n+1)$ holds if $T$ is a complete binary tree on $n$ vertices.*

In [8, Chapter 4.1], this upper bound is improved to $O(n \log n (\log\log n)^2)$, and in [8, Chapter 4.2] $O(n \log n (\log\log n)^2)$ is shown to be an upper bound for complete ternary trees as well.

### 4.2 The General Case

Unfortunately, the algorithm stated in Section 4.1 can not be applied for arbitrary trees since subtrees might differ heavily in size. In [8, Chapter 4.3] a slightly modified algorithm is proposed, which makes use of a tree's Jordan center to handle this problem. Using that algorithm, they show $f_3^{1/2}(n) = O(n \log n (\log\log n)^2)$.

Even though the question by Di Giacomo et al. [4], whether a linear upper bound on $f_3(n)$ exists, remains open, we gain some more insight by this result: as any tree $T$ admits an embedding in $P$ with probability at least 1/2 if $P$ was chosen uniformly at random among all point sets of size $m = O(n \log n (\log\log n)^2)$, $T$ admits an embedding in $Q$ with probability at least $1-(1/2)^k$ if $Q$ was chosen uniformly at random among all point sets of size $mk$. Thus, we can get arbitrary close to probability 1. In particular, to get probability at least $1 - \varepsilon$ we can choose $k = \lceil \log_2(1/\varepsilon) \rceil$.

Trees with maximum degree 4 are a bit tougher to handle; while $f^{1/2}(T)$ has a quasilinear upper bound when $T$ is a complete ternary tree, the best bound so far for the general case is $f_4^{1/2}(n) = O(n^{c+\varepsilon})$ with $c \approx 1.332$; we refer to [8, Chapter 4.3.3]. The question remains open whether a quasilinear upper bound on $f_4^{1/2}(n)$ exists.

### References

[1] D. Auber, J.-P. Domenger, M. Delest, P. Duchon, and J.-M. Fédou. New strahler numbers for rooted plane trees. In *Mathematics and Computer Science III*, pages 203–215. Birkhäuser Basel, 2004.

[2] P. Bose. On embedding an outer-planar graph in a point set. *Comp. Geom. Theo. Appl.*, 23(3):303–312, 2002.

[3] S. Cabello. Planar embeddability of the vertices of a graph using a fixed point set is NP-hard. *J. Graph Algorithms and Appl.*, 10(2):353–363, 2006.

[4] E. Di Giacomo, F. Frati, R. Fulek, L. Grilli, and M. Krug. Orthogeodesic point-set embedding of trees. *Comp. Geom. Theo. Appl.*, 46(8):929–944, 2013.

[5] M. Drmota and H. Prodinger. The register function for $t$-ary trees. *ACM Trans. on Algorithms*, 2(3):318–334, 2006.

[6] B. Katz, M. Krug, I. Rutter, and A. Wolff. Manhattan-geodesic embedding of planar graphs. In *Proc. on Graph Drawing*, pages 207–218, 2009.

[7] M. Kaufmann and R. Wiese. Embedding vertices at points: Few bends suffice for planar graphs. *J. Graph Algorithms and Appl.*, 6(1):115–129, 2002.

[8] M. Scheucher. Orthogeodesic point set embeddings of outerplanar graphs. Master's thesis, Graz University of Technology, Austria, 2015.