Journal of Graph Algorithms and Applications
http://jgaa.info/ vol. 24, no. 3, pp. 343–369 (2020)
DOI: 10.7155/jgaa.00537

On L-shaped point set embeddings of trees: first non-embeddable examples

Torsten Mütze[1] *Manfred Scheucher*[2]

[1]Department of Computer Science, University of Warwick, United Kingdom
[2]Institut für Mathematik, Technische Universität Berlin, Germany

Abstract

An *L-shaped* embedding of a tree in a point set is a planar drawing of the tree where the vertices are mapped to distinct points and every edge is drawn as a sequence of two axis-aligned line segments. There has been considerable work on establishing upper bounds on the minimum cardinality of a point set to guarantee that any tree of the same size with maximum degree 4 admits an L-shaped embedding on the point set. However, no non-trivial lower bound is known to this date, i.e., no known n-vertex tree requires more than n points to be embedded. In this paper, we present the first examples of n-vertex trees for $n \in \{13, 14, 16, 17, 18, 19, 20\}$ that require strictly more points than vertices to admit an L-shaped embedding. Moreover, using computer help, we show that every tree on $n \leq 12$ vertices admits an L-shaped embedding in every set of n points. We also consider embedding *ordered* trees, where the cyclic order of the neighbors of each vertex in the embedding is prescribed. For this setting, we determine the smallest non-embeddable ordered tree on $n = 10$ vertices, and we show that every ordered tree on $n \leq 9$ or $n = 11$ vertices admits an L-shaped embedding in every set of n points. We also construct an infinite family of ordered trees which do not always admit an L-shaped embedding, answering a question raised by Biedl, Chan, Derka, Jain, and Lubiw.

Submitted:	Reviewed:	Revised:	Accepted:
November 2019	March 2020	May 2020	June 2020
	Final:	Published:	
	June 2020	July 2020	

Article type: Regular paper
Communicated by: G. Liotta

Torsten Mütze is also affiliated with Charles University, Faculty of Mathematics and Physics, and was supported by GACR grant GA 19-08554S, and by DFG grant 413902284. Manfred Scheucher was partially supported by DFG Grant FE 340/12-1. An extended abstract of this paper appeared in the Proceedings of the 26th International Symposium on Graph Drawing and Network Visualization [11].

E-mail addresses: torsten.mutze@warwick.ac.uk (Torsten Mütze) scheucher@math.tu-berlin.de (Manfred Scheucher)

1 Introduction

An *L-shaped* embedding of a tree in a point set is a planar drawing of the tree where the vertices are mapped to distinct points of the set and every edge is drawn as a sequence of two axis-aligned line segments; see Figure 1. Here and throughout this paper, all point sets are such that no two points have the same x- or y-coordinate. The investigation of L-shaped embeddings was initiated in [5, 6, 7]. In particular, Di Giacomo et al. [5] showed that $O(n^2)$ points are always sufficient to embed any n-vertex tree. Note that an L-shaped embedding requires that the maximum degree of the tree is at most 4. Moreover, if the maximum degree is 2, then the tree is a path and can be embedded greedily on any point set of the same size. Formally, let $f_d(n)$ denote the minimum number N of points such that every n-vertex tree with maximum degree $d \in \{3, 4\}$ admits an L-shaped embedding in every point set of size N.

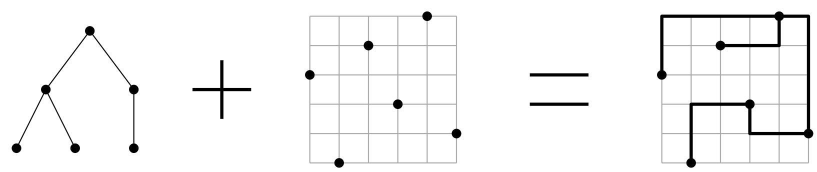


Figure 1: An L-shaped embedding of a tree in a point set.

The second author's master's thesis [13] proposed a method to recursively construct an L-shaped embedding of any n-vertex tree in any point set of size $O(n^{1.58})$ (see also [1]). Biedl et al. [3] improved on this, proving that $f_3(n) = O(n^{1.22})$ and $f_4(n) = O(n^{1.55})$ points are enough. To this date, no lower bound besides the trivial bound $f_d(n) \geq n$ is known, i.e., no known n-vertex tree requires more than n points to be embedded. Di Giacomo, Frati, Fulek, Grilli, and Krug [5] specifically asked for a tree and point set that would prove $f_4(n) > n$. The same question was reiterated by Fink, Haunert, Mchedlidze, Spoerhase, and Wolff [6], and by Biedl, Chan, Derka, Jain, and Lubiw [3]. Kano and Suzuki [7] even conjectured that $f_3(n) = n$.

Biedl et al. [3] also considered a more restricted setting of embedding *ordered* trees, where the cyclic order of the neighbors of each vertex in the embedding is prescribed. They presented a 14-vertex ordered tree which does not admit an L-shaped embedding in a point set of size 14[1], and they raised the problem to find an infinite family of such non-embeddable ordered trees.

1.1 Our results

We begin presenting our results for the setting where there are no constraints on the cyclic order in which the neighbors appear around each vertex of the tree.

[1]Specifically, their counterexample is the 14-vertex caterpillar with 6 vertices on the central path and a pending edge on each side of the four inner vertices of the path. The point set is a $(4, 6, 4)$-staircase in our terminology (see Definition 5).

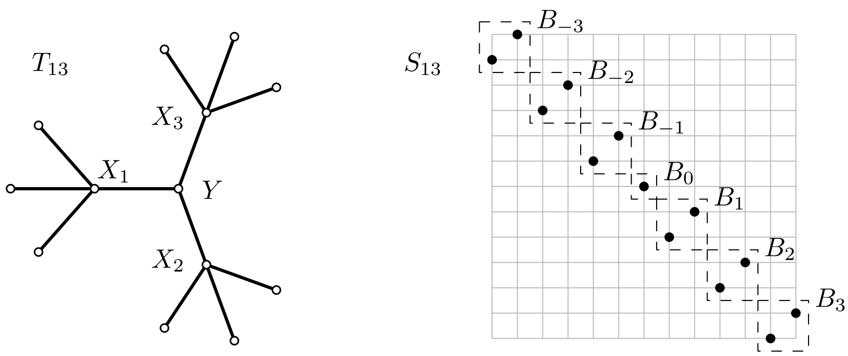


Figure 2: The tree T_{13} (left) does not admit an L-shaped embedding in the $(2, 2, 2, 1, 2, 2, 2)$-staircase point set S_{13} (right). The boxes $B_{-3}, \dots, B_3$ are highlighted by dashed frames.

With brute-force computer search, we verified that all trees on $n \leq 12$ vertices can be embedded in every point set of size n.

Theorem 1 (Computer-based) *Every tree on $n \leq 12$ vertices admits an L-shaped embedding in every set of n points.*

We also formulated a SAT instance to test a given pair of tree and point set for embeddability. This way, we found a 13-vertex tree that does not admit an embedding in a particular point set.

Theorem 2 *The tree T_{13} in Figure 2 does not admit an L-shaped embedding in the point set S_{13} shown in the figure.*

Even though the 13-vertex tree T_{13} was found using the help of a SAT solver, a human-verifiable proof of Theorem 2 is not hard to obtain.

Besides the pair (T_{13}, S_{13}), we also found pairs of trees and point sets that do not admit an embedding for larger values of n. Overall, we found pairs of n-vertex trees and point sets of size n for $n \in \{13, 14, 16, 17, 18, 19, 20\}$. For $n = 15$, however, our computer search did not yield any non-embeddable example (the search was not exhaustive). We remark that all known non-embeddable trees contain T_{13} as a subtree.

We now focus on the more restricted setting of ordered trees introduced in [3], where the cyclic order of the neighbors of each vertex in the embedding is prescribed.

Theorem 3 (Computer-based) *Every ordered tree on $n \leq 9$ vertices or on $n = 11$ vertices admits an L-shaped embedding in every set of n points.*

We also found a 10-vertex tree that does not admit an embedding in a particular point set. This is a smaller non-embeddable instance than the one for $n = 14$ previously presented in [3].

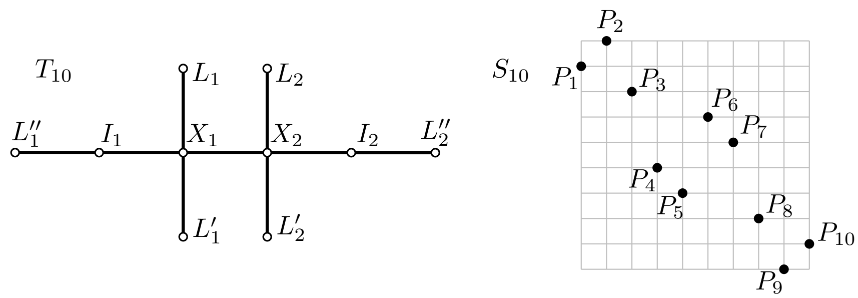


Figure 3: The ordered tree T_{10} (left) does not admit an L-shaped embedding in the point set S_{10} (right).

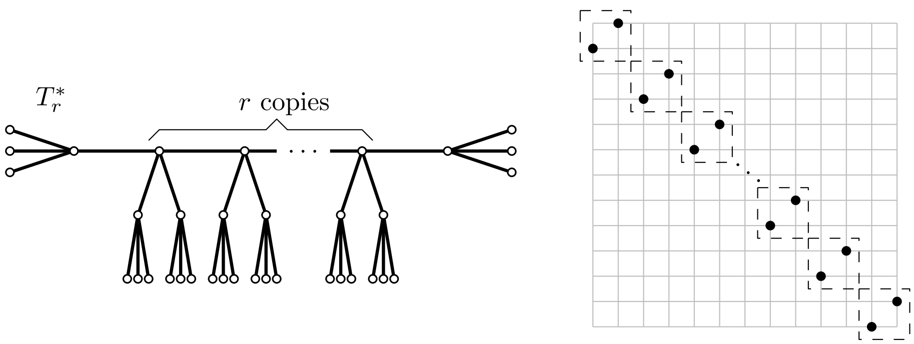


Figure 4: A family of ordered trees T_r^* (left) that does not admit an L-shaped embedding in the n-point $(2,\ldots,2)$-staircase (right) for $n = 9r + 8$ and even $r \geq 10$. The boxes of the point set are highlighted.

Theorem 4 *The ordered tree T_{10} in Figure 3 does not admit an L-shaped embedding in the point set S_{10} shown in the figure.*

Remarkably, the pair (T_{10}, S_{10}) is the only one on $n = 10$ vertices/points not admitting an L-shaped embedding.

Moreover, we construct an infinite family of ordered trees that do not admit an L-shaped embedding on certain point sets, answering a question raised by Biedl, Chan, Derka, Jain, and Lubiw in [3]. As it turns out, the point sets that appear to be difficult for embedding have a regular staircase shape as shown in Figure 4 (see also Figure 2).

Definition 5 (Staircase point set) *For any integer n with $n = a_1 + \cdots + a_k$ where $a_1, \ldots, a_k \in \mathbb{N}$, the $(a_1, \ldots, a_k)$-*staircase *is the point set consisting of a sequence of k disjoint boxes, ordered from top-left to bottom-right, and the ith box contains a sequence of a_i points with increasing x- and y-coordinate.*

Theorem 6 *For any even $r \geq 10$, the ordered tree T_r^* on $n = 9r + 8$ vertices in Figure 4 does not admit an L-shaped embedding in the n-point $(2, \ldots, 2)$-staircase.*

We conjecture that T_r^* does not admit an embedding in the same point set, even when considered as an unordered tree, i.e., in the original unrestricted setting.

1.2 Related work

Besides the problem of finding L-shaped embeddings of arbitrary trees in arbitrary point sets, various special classes of trees and point sets have also been studied. For instance, perfect binary and perfect ternary n-vertex trees can be embedded in any point set of size $O(n^{1.142})$ or $O(n^{1.465})$, respectively [3]. Moreover, trees with pathwidth k can be embedded in any set of $2^k n$ points [13, Chapter 3.3.2] (see also [1]).[2] Further, any n-vertex caterpillar with maximum degree 3 can be embedded in any point set of size n [5]. A *caterpillar* is a tree with the property that all leaves are in distance 1 of a central path. For maximum degree 4 caterpillars, the currently best known upper bound is $4n/3+O(1)$ many points [13, Chapter 5.2.1]. Biedl et al. [3] showed that any ordered caterpillar can be embedded in any point set of size $O(n \log n)$.

When point sets are chosen uniformly at random, i.e., the y-coordinates are a random permutation, it is known that $O(n \log n (\log \log n)^2)$ and $O(n^{1.332})$ points are sufficient to embed any tree with maximum degree 3 or 4, respectively, with probability at least 1/2 [13, Chapter 4] (see also [1]).

Another known setting are non-planar L-shaped point set embeddings, where L-shaped edges are allowed to cross properly, but edge-segments must not overlap. For this setting, it is known that n points are sufficient to embed any n-vertex tree with maximum degree 3 [5, 6] or any n-vertex caterpillar with maximum degree 4 [13, Theorem 21]. For n-vertex trees with maximum degree 4 the currently best upper bound on the required number of points is $7n/3 + O(1)$ [13, Theorem 7].

1.3 Outline of this paper

In Section 2 we present a key lemma that is used repeatedly in our constructions. In Sections 3 and 4 we present the proofs of Theorems 2 and 4, respectively. Section 5 is devoted to proving Theorem 6. We describe our computational approach to proving Theorems 1 and 3 by exhaustive search in Section 6. More non-embeddable small trees are presented in Section 7, together with our SAT model which is used to verify non-embeddability. We conclude in Section 8 with some challenging open problems.

2 Key lemma

The following key lemma is used several times in our arguments about non-embeddability of unordered trees. It asserts that in an L-shaped embedding, two tree vertices of degree 4 cannot both be mapped to the two points in a box

[2] For the definition of pathwidth, we refer the reader to [12].

of size 2 in a staircase point set. The size here refers to the number of points in the box, not to the width or height. Our examples in Theorem 2, Theorem 6, and the ones in Section 7 are all constructed by considering trees with many degree 4 vertices, and staircase point sets with many boxes of size 2, which creates many constraints.

Lemma 7 *Let T be an unordered tree with two vertices X_1 and X_2 of degree 4. Not both X_1 and X_2 can be mapped to the two points in a box of size 2 in a staircase point set.*

Proof: For the sake of contradiction assume that both X_1 and X_2 are mapped to the two points in a box of size 2 in the staircase point set. W.l.o.g. we may assume that X_2 is above and to the right of X_1. As X_1 and X_2 both have degree 4, each of them has edges incident to its left, right, bottom and top. Consider the two edges incident to the top and the right of X_1, and the two edges incident to the bottom and left of X_2. Among these edges, at most one can be an edge connecting X_1 and X_2 (provided they are adjacent in T). Consequently, one of these edges connects X_1 to another vertex outside this box, and one of these edges connects X_2 to another vertex outside this box. As all points outside this box are either above and to the left of it or below and to the right of it, these two edges must cross, a contradiction; see Figure 5. □

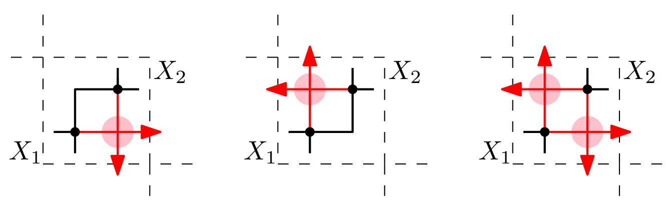


Figure 5: Illustration of the proof of Lemma 7. Crossing edges are highlighted.

3 Proof of Theorem 2

Consider the (unordered) tree T_{13} and the $(2, 2, 2, 1, 2, 2, 2)$-staircase point set S_{13} depicted in Figure 2. We label the degree-3 vertex of T_{13} by Y and the three degree-4 vertices of T_{13} by X_1, X_2, X_3. Moreover, we label the boxes in the staircase point set S_{13} from left to right by $B_{-3}, B_{-2}, \ldots, B_3$. Note the symmetry of T_{13}, as the vertex Y joins three isomorphic subtrees. Moreover, S_{13} has reflection symmetries along both diagonals of the grid.

For the sake of contradiction, we assume that an L-shaped embedding of T_{13} in S_{13} exists. We first derive three lemmas that capture which boxes the vertices X_1, X_2, X_3, Y can be mapped to in such an embedding, and we then complete the proof by distinguishing two main cases.

In the embedding, the L-shaped edge between any two neighboring vertices of the tree can have one of four possible orientations, and we refer to it as an ┐-, ┘-, └-, or ┌-edge.

Lemma 8 *Neither of the four vertices X_1, X_2, X_3, Y is mapped to B_{-3} or to B_3.*

Proof: All points in B_{-3} and B_3 lie on the bounding box of the point set, so if one of the X_i is mapped to such a point, then one of the four edges incident with X_i would leave the bounding box, which is impossible. Moreover, Y cannot be mapped to one of these two boxes, as otherwise one of the X_i, which are the only neighbors of Y in T_{13}, would be mapped to the other point of that same box. □

Lemma 7 immediately gives the following result.

Lemma 9 *Each of the degree-4 vertices X_i is mapped to a distinct box.*

Lemma 10 *Not all three points X_1, X_2, X_3 lie on the same side (above, below, left, or right) of Y.*

Proof: It suffices to prove one of the statements, then the others follow by symmetry. Suppose for the sake of contradiction that X_1, X_2, X_3 all lie above Y. As one edge is incident to the right of Y, one of the X_i, say X_3, is mapped to the same box, and Y is below and to the left of X_3 in that box; see Figure 6. Moreover, YX_3 is an ┘-edge. As X_3 has degree 4, and each box contains at most two points, the edge incident to the top of Y that connects Y to X_1 or X_2 crosses the edge incident to the left of X_3, a contradiction. □

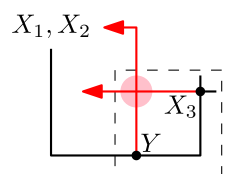
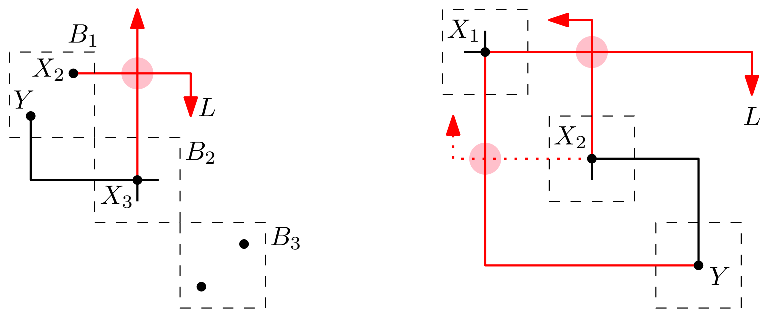


Figure 6: Illustration of the proof of Lemma 10.

By Lemma 8 and Lemma 10, Y is mapped to one of the boxes B_{-1}, B_0, or B_1. By Lemma 9 we may assume that X_1, X_2, X_3 appear in distinct boxes in exactly this order from left to right and also from top to bottom, and none of them is in B_{-3} or B_3. Moreover, from Lemma 10 we conclude that X_1 and X_3 are in other boxes than Y, so at most Y and X_2 are in the same box. We now distinguish two cases.

Case 1: Y and X_2 are mapped to the same box. By symmetry, we may assume that they are mapped to B_1 and that X_2 lies above and to the right of Y. Then the vertex X_3 must be mapped to the box B_2; see Figure 7(a). If

YX_3 were an ⌉-edge, then it would cross the edge incident to the bottom of X_2. It follows that YX_3 is an ⌊-edge. Note that the edge incident to the right of X_2 can only connect to a leaf L that is mapped to $B_2 \cup B_3$, and L must be mapped to the right of X_3, as otherwise the edges X_2L and YX_3 would cross. The edges that are incident to the bottom and right of X_3 can only connect to points from $B_2 \cup B_3$, so together with X_3 and L we already have four vertices that are mapped to $B_2 \cup B_3$. Consequently, the edge incident to the top of X_3 must connect to a point outside of $B_1 \cup B_2 \cup B_3$, and therefore this edge crosses the edge X_2L (see the marked crossing in the figure), a contradiction.



(a) Case 1 (b) Case 2

Figure 7: Illustration of the proof of Theorem 2.

Case 2: Y and X_2 are mapped to distinct boxes, so all four points X_1, X_2, X_3, Y are in different boxes. By symmetry, we assume that X_1 and X_2 both lie above and to the left of Y, and X_3 lies below and to the right of Y. Moreover, we assume that YX_1 is an ⌊-edge and that YX_2 is an ⌉-edge; see Figure 7(b). Note that X_2 cannot connect to any points right of Y, and X_1 can only connect to such points by the edge incident to the right of X_1. As Y is either mapped to B_0 or B_1, there are at most 7 points above and to the left of Y. Therefore, as X_1 and X_2 together with their leaves form a set of 8 points, Y must be mapped to B_1, and exactly one leaf L of X_1 is mapped to a point right of Y, connected to X_1 via an ⌉-edge. Note that X_2 cannot be mapped to B_0, as then the edge incident to the bottom of X_2 could not connect to any point without either crossing YX_1 or YX_2. Consequently, X_2 is mapped to B_{-1}. However, as B_{-1} and B_0 together contain only 3 points, and X_2 together with its leaves form a set of 4 vertices, at least one of the two edges incident to the left or top of X_2 must connect to a point above or left of X_1, and this edge will cross either the edge YX_1 or X_1L (creating one of the two marked crossings in the figure), again a contradiction.

In both cases we obtain a contradiction to the assumption that T_{13} admits an L-shaped embedding in the point set S_{13}. This completes the proof of Theorem 2.

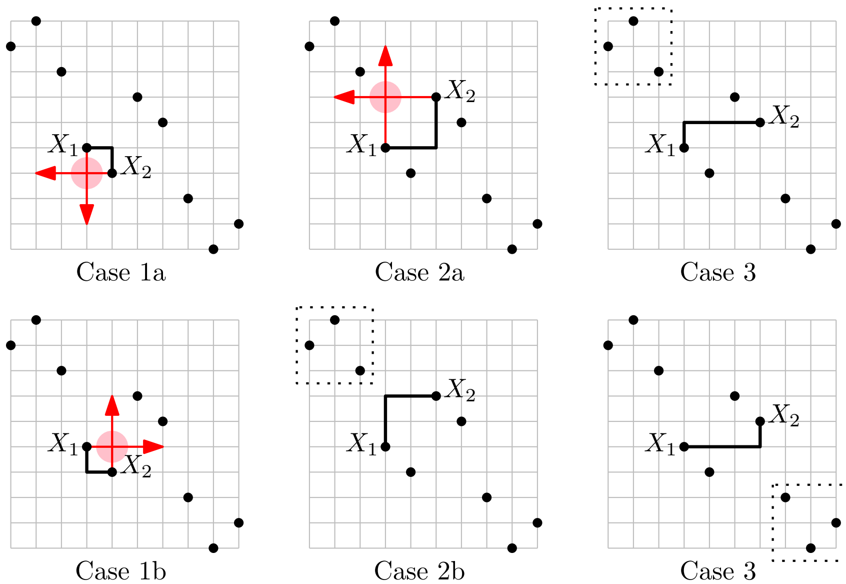


Figure 8: Illustration of the proof of Theorem 4.

4 Proof of Theorem 4

Consider the ordered tree T_{10} and the point set S_{10} depicted in Figure 3. We label the two degree-4 vertices of T_{10} by X_1, X_2 and the two degree-2 vertices incident to X_1 and X_2 by I_1 and I_2, respectively. Moreover, we label the leaves adjacent to X_1 and X_2 by L_1, L_1', L_2, L_2', and the leaves adjacent to I_1 and I_2 by L_1'' and L_2'', as shown in the figure. We label the points of the point set S_{10} from left to right by $P_1, \ldots, P_{10}$. Note the symmetry of T_{10}, and observe that S_{10} has reflection symmetries along both diagonals of the grid.

For the sake of contradiction, we assume that an order-preserving L-shaped embedding of T_{10} in S_{10} exists. Clearly, none of the degree-4 vertices X_1, X_2 can be mapped to any of the four points P_1, P_2, P_9, P_{10} which lie on the bounding box of the point set S_{10}. We also claim that X_1, X_2 cannot be mapped to P_3 or P_8. By symmetry, it suffices to exclude the case that X_1 is mapped to P_3. In this case, we may assume by symmetry that the edge X_1X_2 is incident to the right of X_1. Consequently, due to the cyclic order of the neighbors of X_1, I_1 must be mapped to P_1 and L_1 must be mapped to P_2. Then L_1'' cannot be mapped to any point, a contradiction.

It follows that X_1 and X_2 are mapped to the points P_4, P_5, P_6, P_7. By symmetry, we may assume that X_1 is mapped to P_4. We now distinguish six cases, illustrated in Figure 8:

- Case 1a: X_2 is mapped to P_5 and X_1X_2 is an ⌐-edge. In this case, the edge incident to the bottom of X_1 and the edge incident to the left of X_2 must cross, a contradiction.

- Case 1b: X_2 is mapped to P_5 and X_1X_2 is an └-edge. In this case, the edge incident to the right of X_1 and the edge incident to the top of X_2 must cross, a contradiction.
- Case 2a: X_2 is mapped to P_6 and X_1X_2 is an ┘-edge. In this case, the edge incident to the top of X_1 and the edge incident to the left of X_2 must cross, a contradiction.
- Case 2b: X_2 is mapped to P_6 and X_1X_2 is an ┌-edge. Clearly, none of the four vertices L_1', L_2', I_1, and I_2 can be mapped to P_1, P_2, or P_3. We claim that L_1'' and L_2'' cannot be mapped to any of these points either. By symmetry, it suffices to show the argument for L_1'': Indeed, X_1I_1 is an └-edge, and I_1 can only be mapped to one of P_5, P_8, P_9, or P_{10}. If L_1'' is mapped to one of P_1, P_2, or P_3, then I_1 and L_1'' must be joined via an ┐-edge. Consequently, if I_1 is mapped to P_5, then the edge I_1L_1'' intersects the edge X_1X_2. On the other hand, if I_1 is mapped to P_8, P_9 or P_{10}, then together the two edges X_1I_1 and I_1L_1'' prevent at least one of the two points P_9, P_{10} from being reachable from X_2 via one or two L-shaped edges. Indeed, given the two edges X_1I_1 and I_1L_1'', then neither P_9 nor P_{10} can be reached from X_2 via a single edge, and the only way to reach one of these points via two edges from X_2 is to first take a ┐-edge incident to the top of X_2, but the edge incident to the top of X_2 must lead to the leaf L_2. This completes the argument that L_1'' and L_2'' cannot be mapped to P_1, P_2, or P_3. Consequently, only two vertices, namely L_1 and L_2 can be mapped to the three points P_1, P_2, P_3, a contradiction.
- Case 3: X_2 is mapped to P_7. The subcases where X_1X_2 is an ┌-edge or an ┘-edge are symmetric, so it suffices to consider the first one. In this case we can argue as in Case 2b that only L_1 and L_2 can be mapped to the three points P_1, P_2, P_3, a contradiction.

In each case we obtain a contradiction, so the proof of Theorem 4 is complete.

5 Proof of Theorem 6

Throughout this section, we assume that $r \geq 10$ is even and $n = 9r + 8$. We label the degree-4 vertices of the ordered n-vertex tree T_r^* along the central path by $X_0, \ldots, X_{r+1}$, and for any vertex X_i, $1 \leq i \leq r$, we label its two neighbors of degree 4 not on the central path by X_i' and X_i'', as shown in Figure 9. For our later arguments it will be convenient to orient the edges of T_r^* which are not on the central path. Edges incident to a leaf are oriented away from the leaf and edges $X_i'X_i$ and $X_i''X_i$ are oriented towards X_i. In an embedding of the tree, any L-shaped oriented edge appears in one of eight possible orientations, and four of them are important for our proofs; we refer to them as an ⬑-, ↰-, ↳-, or ↴-edge, respectively, where the arrow marks the tip of the oriented edge.

Lemma 7 immediately gives the following result.

Lemma 11 *Each of the degree-4 vertices X_i for $0 \leq i \leq r+1$, and X_i', X_i'' for $1 \leq i \leq r$, is mapped to a distinct box of the n-point $(2, \ldots, 2)$-staircase.*

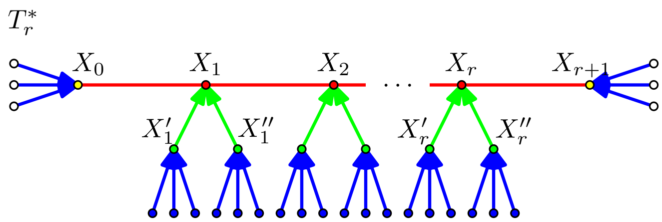


Figure 9: Labeling of vertices of the ordered tree T_r^* for the proof of Theorem 6.

We refer to the sequence of ⌝- or ⌞-edges connecting the central path vertices $X_0, \ldots, X_{r+1}$ as the *spine*. By symmetry, we may assume w.l.o.g. that X_0 is mapped to a box on the left of X_1. In the following we distinguish two main cases, depending on whether X_0X_1 is an ⌝-edge or an ⌞-edge.

5.1 Case 1: X_0X_1 is an ⌝-edge

Throughout this section, we assume that X_0X_1 is an ⌝-edge. Lemma 11 and the cyclic order of neighbors around each of the vertices X_i, $i = 0, \ldots, r+1$, now enforce a particular shape of all tree edges that connect two degree-4 vertices, as captured by the following lemma; see Figure 10.

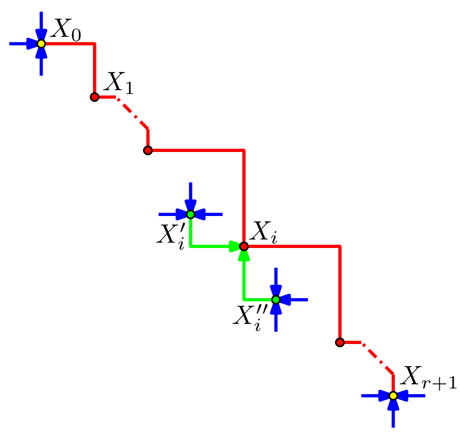


Figure 10: Illustration of Lemma 12.

Lemma 12 *The vertices $X_0, \ldots, X_{r+1}$ appear exactly in this order from left to right, and any two consecutive such vertices are connected by an ⌝-edge. Moreover, for $i = 1, \ldots, r$,*

- *the vertices X_i' and X_i are connected by an ↳-edge;*
- *the vertices X_i'' and X_i are connected by an ⮤-edge;*
- *the end segments of the three edges directed from the leaves towards the vertices X_i', X_i'', X_0, and X_{r+1} form a ⊥, ⊢, ⊣, and ⊤, respectively.*

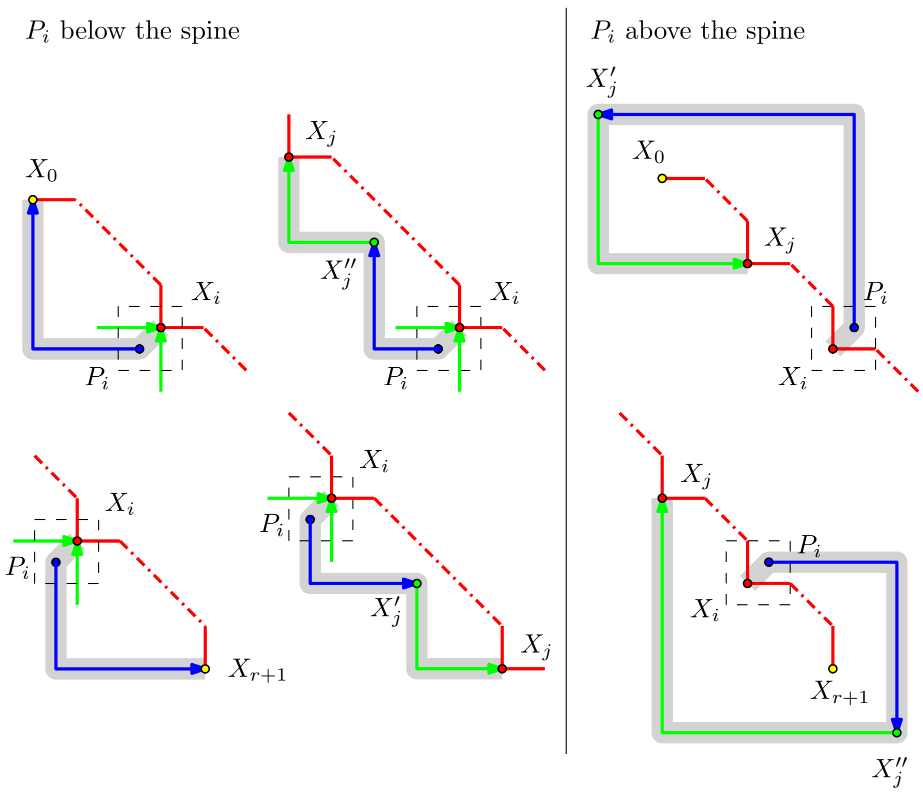


Figure 11: Illustration of the six different cases in Lemma 13. The corresponding blockers are highlighted with bold lines.

By Lemma 11, each box containing one of the X_i, $1 \le i \le r$, contains a second point to which a leaf is mapped. We denote this point by P_i. Combining Lemmas 11 and 12 yields the following lemma, which is illustrated in Figure 11.

Lemma 13 *For every point P_i below the spine exactly one of the following four conditions holds:*

- *P_i is connected to X_0 by an ⮤-edge;*
- *P_i is connected to X_{r+1} by an ↳-edge;*
- *there is an index j, $1 \le j < i$, such that P_i, X_j'', X_j are joined by two consecutive ⮤-edges;*
- *there is an index j, $i < j \le r$, such that P_i, X_j', X_j are joined by two consecutive ↳-edges.*

For every point P_i above the spine exactly one of the following two conditions holds:

- *there is an index j, $1 \le j \le r$, such that P_i, X_j' and X_j', X_j are joined by an ↰-edge and an ↳-edge, respectively, wrapping around the top left end of the* *spine;*

- *there is an index j, $1 \leq j \leq r$, such that P_i, X''_j and X''_j, X_j are joined by an* ⬎*-edge and an* ↲*-edge, respectively, wrapping around the bottom right end of the spine.*

Consider any pair of points P_i, X_0 as in Lemma 13 connected by an ↲-edge. We refer to this edge together with the short diagonal line joining the points X_i and P_i in the same box (this line is not part of the tree embedding), as a ↲-*blocker starting at X_i and ending at X_0*; see Figure 11. Similarly, given any triple of points P_i, X''_j, X_j as in Lemma 13 joined by two consecutive ↲-edges, we refer to these two edges together with the line joining X_i and P_i, as a ⬑↲-*blocker starting at X_i and ending at X_j*. Moreover, given any triple of points P_i, X'_j, X_j as in Lemma 13 joined by an ↰-edge followed by an ↳-edge wrapping around the top left end of the spine, we refer to these two edges together with the line joining X_i and P_i, as a ⮣-*blocker starting at X_i and ending at X_j*. The terms ↳-*blocker*, ⬎-*blocker* and ⮠-*blocker* are defined analogously; see the bottom part of Figure 11. The right hand side of Figure 11 shows that for a ⮣-blocker or a ⮠-blocker, there is no constraint on j for a given i (other than $1 \leq j \leq r$). Observe also that no tree edge can cross a blocker.

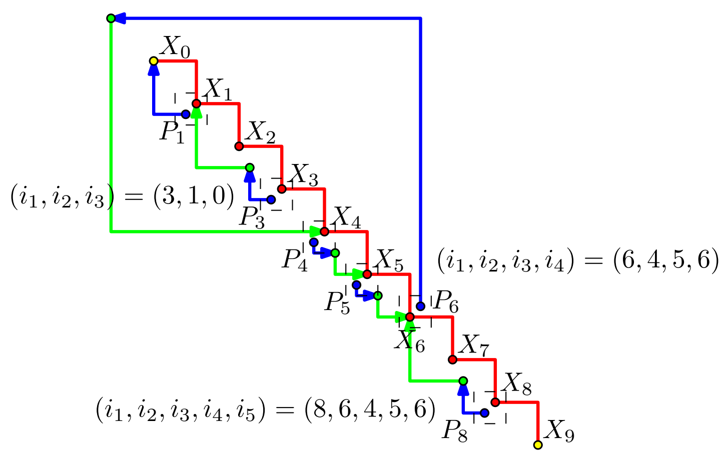


Figure 12: Illustration of the definition of blocker sequences. The figure shows three blocker sequences, one starting at X_3 and ending at X_0 with corresponding index sequence $(i_1, i_2, i_3) = (3, 1, 0)$, one starting and ending at X_6 with corresponding index sequence $(i_1, i_2, i_3, i_4) = (6, 4, 5, 6)$, and one starting at X_8 and ending at X_6 with corresponding index sequence $(i_1, i_2, i_3, i_4, i_5) = (8, 6, 4, 5, 6)$.

For every index i_1, $1 \leq i_1 \leq r$, we define a finite sequence of blockers as follows; see Figure 12: For $j = 1, 2, \ldots$ we consider the point X_{i_j} and the blocker starting at X_{i_j}. The endpoint $X_{i_{j+1}}$ of this blocker defines the next index i_{j+1}. If $i_{j+1} \notin \{i_1, \ldots, i_j\} \cup \{0, r+1\}$, we repeat this process, otherwise we stop. This yields a finite sequence of indices $i_1, i_2, \ldots, i_\ell$, such that any two consecutive points X_{i_j} and $X_{i_{j+1}}$ are joined by a blocker starting at X_{i_j} and

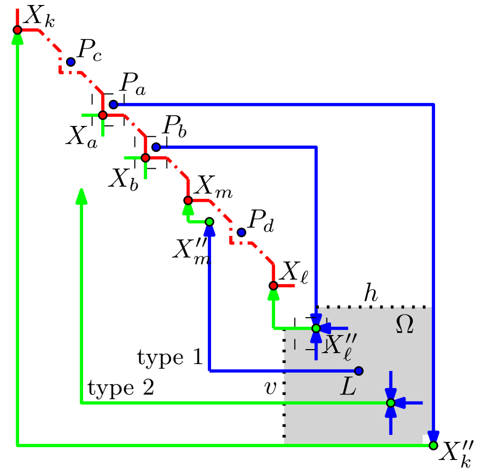


Figure 13: Illustration of Lemma 14.

ending at $X_{i_{j+1}}$. Clearly, $i_2, \ldots, i_\ell$ all depend on the choice of i_1. Moreover, by the termination condition above we either have $i_\ell \in \{i_1, \ldots, i_{\ell-1}\}$ if the blockers close cyclically, or $i_\ell \in \{0, r+1\}$ if the last blocker ends at X_0 or X_{r+1} (the terminal index is included in the sequence). These two cases are illustrated in Figure 12. We refer to the sequence of blockers generated in this fashion as the *blocker sequence starting at* X_{i_1}.

Statement and proof of the following key lemma are illustrated in Figure 13.

Lemma 14 *Let* $1 \leq a < b \leq r$ *be such that* P_a *and* P_b *are two consecutive points above the spine each contained in a* ↲*-blocker, and let* X_k *and* X_ℓ *be the blocker endpoints, respectively. Then there are indices* c, d *with* $k < c < d \leq \ell$ *such that* P_c *and* P_d *are above the spine.*
Symmetrically, if P_a *and* P_b*,* $1 \leq a < b \leq r$*, are two consecutive points above the spine each contained in a* ↱*-blocker, then there are indices* c, d *with* $k \leq c < d < \ell$ *such that* P_c *and* P_d *are above the spine.*

Observe that this lemma does not make any assertions about the relative positions of the points in $\{X_a, X_b\}$ and $\{X_k, X_\ell\}$. In particular, it does not make any assertions about the disjointness of the sets $\{P_a, P_b\}$ and $\{P_c, P_d\}$.

Proof: It suffices to prove the first part of the lemma where P_a and P_b are both contained in a ↲-blocker. The second part follows by symmetry. Let h denote the horizontal line segment slightly above the box containing X''_ℓ between the two vertical segments of the ⌝-edges leaving P_a and P_b. Let v denote the vertical line segment slightly left of the box containing X''_ℓ between the two horizontal segments of the ⌞-edges leaving X''_k and X''_ℓ. Let Ω denote the region enclosed by the two ↲-blockers starting at X_a and X_b and between the segments h and v, without the point X''_k. Note that Ω contains X''_ℓ and also the second point in its box, but neither X''_k nor the second point in its box, so Ω contains an even number of points from the $(2,\dots,2)$-staircase. Observe also that no edge crosses the segment h, as P_a and P_b are consecutive points above the spine. Consider an edge crossing the segment v. By Lemma 12, this can only be an ↰-edge starting at a leaf L in Ω and ending at a vertex X''_m for some m, $k < m < \ell$ (type 1), or an ↰-edge starting at some X''_m in Ω, $k < m < \ell$, and ending at X_m (type 2). Figure 13 gives an illustration of both types of edges. In the case of a type 2 edge, all three leaves adjacent to X''_m must also be in Ω. Therefore, every type 1 edge contributes 1 to the number of vertices in Ω, and every type 2 edge contributes 4 to the number of vertices in Ω. Note that the other two leaves adjacent to X''_ℓ apart from P_b must also be in Ω, so X''_ℓ together with these two leaves contributes 3 to the number of vertices in Ω. As the number of points from the $(2,\dots,2)$-staircase in Ω is even, there must be at least one type 1 edge starting at a leaf L in Ω and ending at a vertex X''_m, $k < m < \ell$.

By Lemma 12, X''_m is connected to X_m by another ↰-edge. Now consider the blocker sequence starting at X_m. We prove that it must contain a ⮣-or ⮠-blocker. For the sake of contradiction suppose not. Then it can only have ⮤-blockers, but no ↰-, ↳-, or ⮡-blockers: Indeed, an ↰-blocker would lead to X_0, which is impossible because of the ↰-edge between X''_k and X_k that shields this blocker sequence from the left. Moreover, an ↳- or ⮡-blocker would force one of the points X_i, $1 \le i \le r+1$, to lie inside Ω, which is impossible. However, if the blocker sequence consists only of ⮤-blockers, then it must end at X_0, which is again impossible. This proves our claim that the blocker sequence starting at X_m contains a ⮣- or ⮠-blocker, and the first such blocker in the sequence will contain the desired point P_c, $k < c \le m$ (if the very first blocker is of this type then $c = m$).

An analogous argument applies to the blocker sequence starting at X_ℓ. As the ↰-edge between L and X''_m shields this blocker sequence from the left, the first ⮣- or ⮠-blocker in this sequence contains the desired point P_d, $m < d \le \ell$. This completes the proof of the lemma. □

We will later use the following corollary of Lemma 14.

Corollary 15 *Suppose there are in total $\alpha \ge 2$ points $P_{i_1},\dots,P_{i_\alpha}$, $i_1 < \dots < i_\ell$, above the spine each contained in a ⮠-blocker, and let X_k be the endpoint of the blocker starting at X_{i_1}. Then we have $k < i_1$, and there are at least $2(\alpha-1)$ many points P_i with $i > k$ above the spine.*
Symmetrically, suppose there are in total $\alpha \ge 2$ points $P_{i_1},\dots,P_{i_\alpha}$, $i_1 < \dots < i_\ell$, above the spine each contained in a ⮣-blocker, and let X_k be the endpoint of the blocker starting at X_{i_α}. Then we have $k > i_\alpha$, and there are at least $2(\alpha-1)$ many points P_i with $i < k$ above the spine.

Proof: By symmetry, it suffices to prove the statement for ⮠-blockers. For $j = 1,\dots,\alpha$, let X_{k_j} be the endpoint of the blocker starting at X_{i_j}. As these blockers do not intersect, we have $k_1 < k_2 < \dots < k_\alpha$ and consequently the intervals $]k_j, k_{j+1}]$, $j = 1,\dots,\alpha-1$, are pairwise disjoint. Applying Lemma 14 to the pair of consecutive points $P_{i_j}, P_{i_{j+1}}$, $j = 1,\dots,\alpha-1$, shows that there are at least two points P_i with $i \in]k_j, k_{j+1}]$ above the spine. Overall, this gives

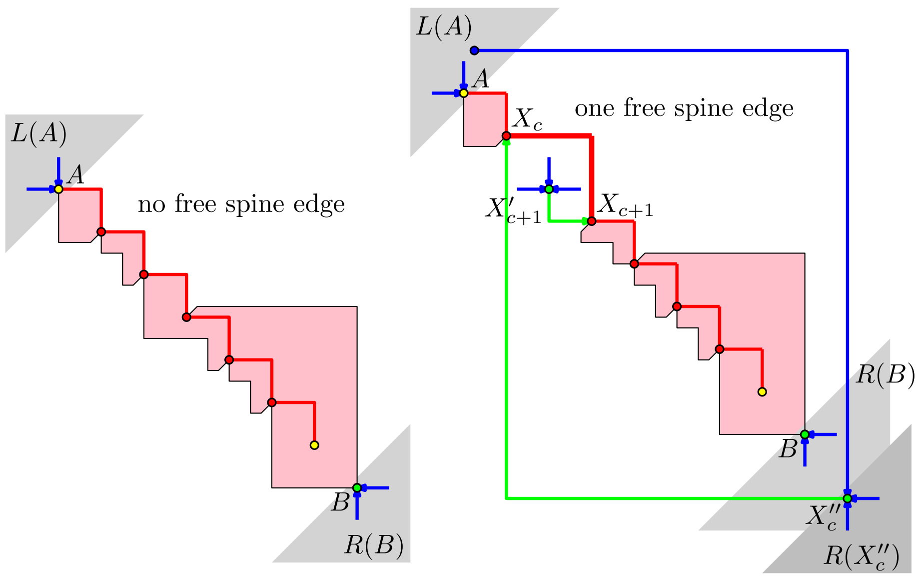


Figure 14: Illustration of Lemma 16. Regions enclosed by blockers and the spine are shaded.

$2(\alpha - 1)$ points P_i with $i > k_1 = k$ above the spine. As $2(\alpha - 1) \geq \alpha$, we must have $k < i_1$. □

Consider the collection of all blocker sequences starting at any of the points X_i, $1 \leq i \leq r$. Any blocker in one of these sequences encloses a region together with the spine, and if this region touches a spine edge from the bottom left, then we say that this spine edge is *enclosed*. Any spine edge that is not enclosed is called *free*. In Figure 14, enclosed regions are shaded. For any point A of the staircase point set, consider the second point A' in the same box of the staircase, and let $L(A)$ denote the halfplane containing those two points, such that the points lie slightly to the left of the boundary of the halfplane. We define the halfplane $R(A)$ analogously, by changing left and right in the previous definition.

Lemma 16 *There is no valid embedding of T_r^* with zero or one free spine edges.*

Proof: We choose two particular degree-4 vertices A and B of T_r^* as follows: If there are no ↱-blockers, then $A := X_0$, and otherwise A is defined as the middle vertex of the outermost ↱-blocker. Similarly, if there are no ↲-blockers, then $B := X_{r+1}$, and otherwise B is defined as the middle vertex of the outermost ↲-blocker; see Figure 14. Note that if $A = X_0$ and the spine edge X_0X_1 is enclosed, then the edge incident to the bottom of X_0 is part of a ↥-blocker. Similarly, if $B = X_{r+1}$ and the spine edge X_rX_{r+1} is enclosed, then the edge incident to the left of X_{r+1} is part of a ↳-blocker.

We first assume that there is no free spine edge. Consider the regions $L(A)$ and $R(B)$. Note that A and exactly two of the leaves adjacent to it lie in $L(A)$, and that B and exactly two of the leaves adjacent to it lie in $R(B)$. On the other hand, both regions contain an even number of points from the $(2,\dots,2)$-staircase. This immediately yields a contradiction, as none of the vertices X'_i, X''_i, $1 \le i \le r$, or any of the leaves adjacent to them can reach into $L(A)$ or $R(B)$; see the left hand side of Figure 14.

It remains to consider the case that there is one free spine edge X_cX_{c+1}, $0 \le c \le r$. In the following we only consider the subcase $1 \le c \le r-1$; see the right hand side of Figure 14. The remaining subcases $c = 0$ and $c = r$ are symmetric, and can be handled analogously. We again consider the regions $L(A)$ and $R(B)$. As X_cX_{c+1} is the only free spine edge, at least one of the vertices X''_c, X'_{c+1} or one of the leaves adjacent to one of them must be inside $L(A)$, and one of them must be inside $R(B)$. By symmetry, we may assume that the ⬑-edge from X''_c to X_c or the ⬑-edge entering X''_c has its starting point in $R(B)$. This prevents the ↳-edge from X'_{c+1} to X_{c+1} and the ↳-edge entering X'_{c+1} from reaching into $L(A)$. In this situation the starting point of the ↴-edge entering X''_c is the only one that can reach into $L(A)$, wrapping around the entire spine, which forces X''_c to be in $R(B)$. This, however, leads to a contradiction, as only 3 vertices would be mapped to points in $R(X''_c)$. □

With Corollary 15 and Lemma 16 in hand, we are ready to complete the proof of Theorem 6 in the case that X_0X_1 is an ↴-edge. We let α_L and α_R denote the number of points P_i above the spine that are contained in a ↪- or ↩-blocker, respectively, and we define $\alpha := \alpha_L + \alpha_R$. Observe that by the second part of Lemma 13, for every point P_i above the spine, the corresponding point X_i in the same box is the starting point of a ↪- or ↩-blocker, so α is the total number of points P_i above the spine. Moreover, when considering the points P_i above the spine from left to right, then we first encounter all those that are contained in a ↪-blocker, and then all those that are contained in a ↩-blocker. By symmetry we may assume that $\alpha_L \le \alpha_R$. In the following we distinguish the five cases $\alpha \in \{0,1,2,3,4\}$ and the case $\alpha > 4$, and we show that none of them can occur.

Case $\alpha = 0$: We claim that in this case, exactly one of the spine edges X_cX_{c+1}, $0 \le c \le r$, is free. Applying Lemma 16 will therefore conclude the proof.

Consider the blocker sequences starting at X_i for all $i = 1,\dots,r$. Each such blocker sequence contains only ⮤-, ⮧, ⬑-, and ↳-blockers, but no ↪- or ↩-blockers, and hence it either ends at X_0 or X_{r+1}. If all blocker sequences end at X_{r+1}, then the blocker sequence starting at X_1 only consists of ⮧- and ↳-blockers, and it encloses all spine edges X_iX_{i+1}, $1 \le i \le r$, i.e., X_0X_1 is the only free spine edge and the claim is proved. Symmetrically, if all blocker sequences end at X_0, then the blocker sequence starting at X_r only consists of ⮤- and ⬑-blockers, and it encloses all spine edges X_iX_{i+1}, $0 \le i < r$, i.e., X_rX_{r+1} is the only free spine edge and the claim is proved. It remains to consider the case that at least one blocker sequence ends at X_0 and at least one ends at X_{r+1}. We let c be the largest index $1 \leq i \leq r$ for which the blocker sequence starting at X_i ends at X_0. By this definition, the blocker sequence starting at X_c contains no points X_i for $i > c$, otherwise the blocker sequence starting at such a point X_i would also end at X_0. Consequently, the blocker sequence starting at X_c only consists of ↰- and ⬑-blockers, and it encloses all spine edges X_iX_{i+1}, $0 \leq i < c$, which entails that all blocker sequences starting at X_i for any $1 \leq i \leq c$ end at X_0 as well. Consequently, the blocker sequence starting at X_{c+1}, which ends at X_{r+1} by definition, only consists of ⤷- and ↳-blockers, and it encloses all spine edge X_iX_{i+1}, $c+1 \leq i \leq r$, which entails that all blocker sequences starting at X_i for any $c+1 \leq i \leq r$ end at X_{r+1} as well. We conclude that X_cX_{c+1} is the only free spine edge.

Case $\alpha = 1$**:** We only need to consider the case $(\alpha_L, \alpha_R) = (0, 1)$. Let P_a be the unique point above the spine, i.e., P_a is contained in a ⮤-blocker. Consider the blocker sequence starting at X_a. If it ends at X_0, then it encloses all spine edges X_iX_{i+1}, $0 \leq i \leq r$, i.e., there are no free spine edges, and so we are done with the help of Lemma 16. Otherwise this blocker sequence ends at X_a, enclosing the spine edges X_iX_{i+1}, $a \leq i \leq r$. If $a = 1$, then $X_{a-1}X_a = X_0X_1$ is the only free spine edge, and we are done using Lemma 16. Otherwise consider the blocker sequences starting at X_i for all $1 \leq i < a$, which must end either at X_0 or X_a. We let c be the largest index $1 \leq i < a$ for which the blocker sequence starting at X_i ends at X_0. Similarly to the case $\alpha = 0$, we obtain that the blocker sequence starting at X_c encloses all spine edges X_iX_{i+1}, $0 \leq i < c$, that the blocker sequence starting at X_{c+1} encloses all spine edges X_iX_{i+1}, $c+1 \leq i < a$, and that X_cX_{c+1} is the only free spine edge. Consequently, we are done with the help of Lemma 16.

Case $\alpha = 2$**:** We only need to consider the cases $(\alpha_L, \alpha_R) = (1, 1)$ and $(\alpha_L, \alpha_R) = (0, 2)$. Let P_a, P_b, $a < b$, be the two points above the spine.

We first consider the case $(\alpha_L, \alpha_R) = (1, 1)$, i.e., P_a is contained in a ⮡-blocker and P_b is contained in a ⮤-blocker. Let S_a and S_b be the blocker sequences starting at X_a and X_b, respectively. Observe that either S_a and S_b both end at X_a, or both end at X_b, or S_a ends at X_a and S_b ends at X_b. In the first two cases, there are no free spine edges, so applying Lemma 16 concludes the proof. If $b - a = 1$, then $X_aX_{a+1} = X_{b-1}X_b$ is the only free spine edge, and we are done with Lemma 16. Otherwise the blocker sequences starting at X_i for all $a < i < b$ either end at X_a or X_b. We let c be the largest index $a < i < b$ for which the blocker sequence starting at X_i ends at X_a. Similarly to the case $\alpha = 0$, we obtain that the blocker sequence starting at X_c encloses all spine edges X_iX_{i+1}, $a \leq i < c$, that the blocker sequences starting at X_{c+1} encloses all the spine edges X_iX_{i+1}, $c+1 \leq i < b$, and that X_cX_{c+1} is the only free spine edge, so applying Lemma 16 concludes the proof.

We now consider the case $(\alpha_L, \alpha_R) = (0, 2)$, i.e., P_a and P_b are both contained in a ⮤-blocker. Let X_k be the endpoint of the blocker starting at X_a. From Corollary 15, we obtain that $k < a$, i.e., the blocker sequence starting at X_a must end at X_0, enclosing all spine edges X_iX_{i+1}, $0 \leq i \leq r$. Applying Lemma 16 again completes the proof.

Case $\alpha = 3$: We only need to consider the cases $(\alpha_L, \alpha_R) = (1, 2)$ and $(\alpha_L, \alpha_R) = (0, 3)$. Let P_a, P_b, P_c, $a < b < c$, be the three points above the spine.

We first consider the case $(\alpha_L, \alpha_R) = (1, 2)$, i.e., P_b, P_c are both contained in a ↵-blocker. Let X_k be the endpoint of the blocker starting at X_b. From Corollary 15, we obtain that $k < b$, i.e., the blocker sequence starting at X_b must end at X_a, together with the blocker sequence starting at X_a, and both enclose all spine edges X_iX_{i+1}, $0 \leq i \leq r$. Consequently, we are done with the help of Lemma 16.

We now consider the case $(\alpha_L, \alpha_R) = (0, 3)$, i.e., all three points P_a, P_b, P_c are contained in a ↵-blocker. Corollary 15 implies that there are at least $2(\alpha_R - 1) = 4$ points P_i above the spine, a contradiction.

Case $\alpha = 4$: We only need to consider the cases $(\alpha_L, \alpha_R) = (2, 2)$, $(\alpha_L, \alpha_R) = (1, 3)$, and $(\alpha_L, \alpha_R) = (0, 4)$. Let P_a, P_b, P_c, P_d, $a < b < c < d$, be the four points above the spine.

If $(\alpha_L, \alpha_R) = (2, 2)$, then Corollary 15 shows that the blocker sequences starting at X_b and X_c cannot coexist: Specifically, the blocker sequence starting at X_b with a ↳-blocker ends at X_i with $b < i \leq r + 1$, and the blocker sequence starting at X_c with a ↵-blocker ends at X_j with $0 \leq j < c$. This is a contradiction. Similarly, if $(\alpha_L, \alpha_R) = (1, 3)$, then Corollary 15 shows that the blocker sequences starting at X_a and X_b cannot coexist. If $(\alpha_L, \alpha_R) = (0, 4)$, then Corollary 15 implies that there are at least $2(\alpha_R - 1) = 6$ points P_i above the spine, a contradiction.

Case $\alpha > 4$: Corollary 15 shows that there are at least $2(\alpha_L - 1) + 2(\alpha_R - 1) = 2\alpha - 4$ points P_i above the spine, which is a contradiction, as $2\alpha - 4 > \alpha$ for $\alpha > 4$.

5.2 Case 2: X_0X_1 is an ∟-edge

Throughout this section, we assume that X_0X_1 is an ∟-edge. Lemma 11 and the cyclic order of neighbors around each of the vertices X_i, $i = 0, \ldots, r + 1$, now enforce a particular shape of all tree edges that connect two degree-4 vertices, as captured by the following lemma; see Figure 15.

Lemma 17 *For $i = 0, \ldots, r$, the vertex X_i is left of X_{i+1} and both are connected by an ∟-edge if i is even, and the vertex X_i is right of X_{i+1} and both are connected by an ⌝-edge if i is odd. Moreover, for $i = 1, \ldots, r$,*

- *the vertices X_i' and X_i are connected by an ⌝↓-edge if i is even, and they are connected by an ↑∟-edge if i is odd;*

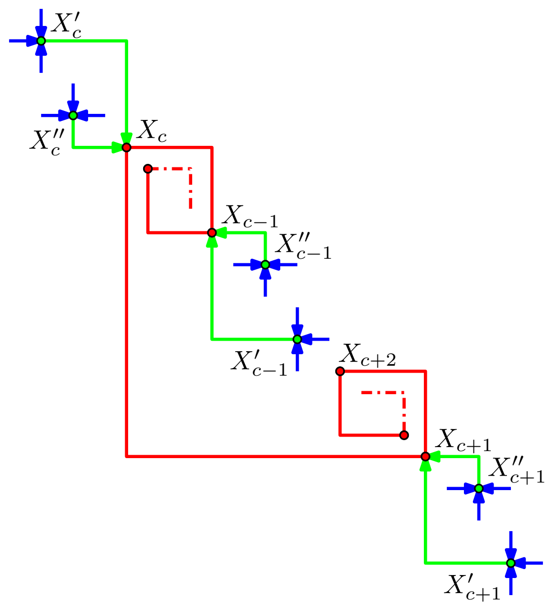


Figure 15: Illustration of Lemma 17.

- *the vertices X''_i and X_i are connected by an ↳-edge if i is even, and they are connected by an ↰-edge if i is odd;*
- *the end segments of the three edges directed from the leaves towards the vertex X_0 form a →↓←, and the end segments of the three edges directed from the leaves towards X'_i and X''_i form a →↓↑ or →↓←, respectively, if i is even, and a ↓←↑ or →←↑ if i is odd.*

We define the *length* of a spine edge X_iX_{i+1}, $0 \leq i \leq r$, in the embedding as the number of boxes in the $(2,\ldots,2)$-staircase between its endpoints plus 1. For instance, if it connects two neighboring boxes, then its length is 1. By Lemma 17, there is a unique longest spine edge X_cX_{c+1}, $0 \leq c \leq r$, and the two length sequences of the edges X_iX_{i+1} for $i = c, c+1, \ldots, r$ and $X_{i+1}X_i$ for $i = c, c-1, \ldots, 0$ are strictly decreasing, i.e., each of the two corresponding parts of the spine spirals into itself in counterclockwise or clockwise direction, respectively, as shown in Figure 15. By the requirement that $r \geq 10$, the longer of these two sequences consists of at least 6 spine edges, and by symmetry we may assume that it is the latter one, i.e., the initial part of the spine looks as shown in Figure 16.

For $i \in \{1,2\}$, we let h_i denote a horizontal line segment between the vertical segments of the spine edges X_iX_{i+1} and $X_{i+2}X_{i+3}$, passing above the box containing X_1 and P_1 if $i = 1$ and below the box containing X_2 and P_2 if $i = 2$, and let Ω_i denote the region enclosed by the spine and this segment; see the figure. One of the leaves of the tree T_r^* must be mapped to the point P_1, which lies in the same box as X_1. This can only be the leaf adjacent to X''_2 via an

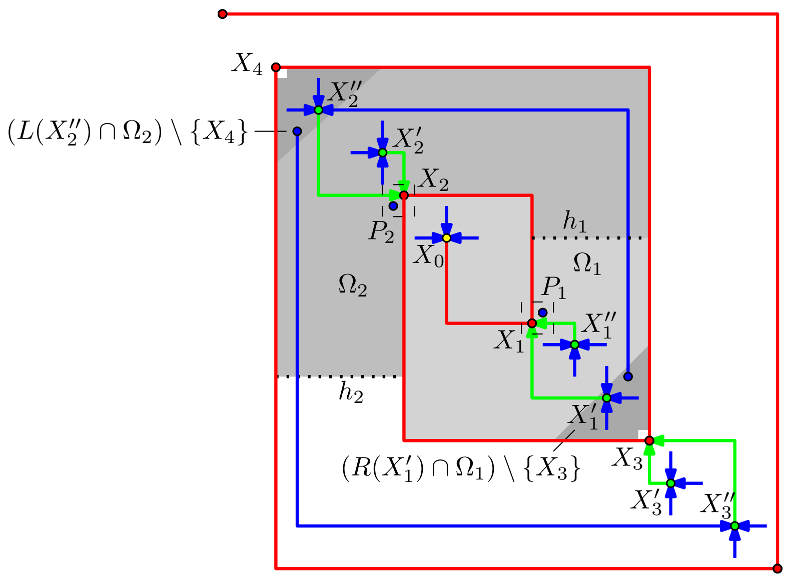


Figure 16: Illustration of the proof of Theorem 6 in Case 2.

↰-edge, or the leaf adjacent to X_1' via an ↴-edge (if P_1 is above and to the right of X_1), or the leaf adjacent to X_1'' via an ↳-edge (if P_1 is below and to the left of X_1). In the last two cases, the leaf adjacent to X_2'' via an ↴-edge must lie in the region $(R(X_1') \cap \Omega_1) \setminus \{X_3\}$ or in the region $(R(X_1'') \cap \Omega_1) \setminus \{X_3\}$, respectively. This is because this region contains an even number of points, and therefore an even number of tree vertices must be mapped to them. In any case, the edge incident to the right of X_2'' must reach into Ω_1. Consequently, the leaf adjacent to X_3'' via an ↳-edge must lie in the region $(L(X_2'') \cap \Omega_2) \setminus \{X_4\}$, in order to map an even number of tree vertices to this region. However, as none of the leaves adjacent to X_2' can connect to P_2, which lies in the same box as X_2, no vertex is mapped to P_2, a contradiction.

This completes the proof of Theorem 6.

6 Computer-based proofs of Theorems 1 and 3

We implemented a C++ program to test if a given (unordered or ordered) tree admits an L-shaped embedding in a given set of points. Our algorithm recursively embeds vertices and edges in all possible ways until either a crossing occurs or a valid drawing is obtained.

Each point set is represented by a permutation, which captures the y-coordinates of the points from left to right. Those permutations are generated in lexicographic order using the C++ standard library function `next_permutation`. When embedding unordered trees, we only need to test point sets that are non-isomorphic up to rotation and mirroring, as an unordered tree is embeddable in a point set if and only if it is embeddable on the rotated or mirrored point set. This filtering of point sets is achieved by considering only the lexicographically smallest permutation under these two operations. Similarly, when embedding ordered trees, we may omit testing point sets that are isomorphic up to rotation (but not mirroring).

The list of all non-isomorphic unordered and ordered trees was generated with SageMath [14], using the integrated nauty graph generator [9], and then loaded by the C++ program.

When testing ordered trees, we only need to test trees that admit more than one way to cyclically order the neighbors of all vertices, as otherwise the tree is equivalent to the corresponding unordered tree. Here we consider two ordered trees the same if they differ only in changing the orientation of all cyclic orders from clockwise to counterclockwise or vice versa, which corresponds to mirroring the embedding.

As pairs of trees and point sets can be tested independently, we parallelized our computations; see Table 1. The source code of all those programs is available as supplementary material to this paper or on the websites [10].

Table 1: Number of non-isomorphic point sets and unordered/ordered trees with maximum degree 4 up to $n \leq 12$, and the computation times of our C++ program. The times marked with * are the sum of parallelized computations on 16 cores.

n	point sets OEIS/A903	unordered trees OEIS/A602	CPU time	point sets OEIS/A263685	ordered trees	CPU time
4	7	2		9	2	
5	23	3		33	3	
6	115	5		192	5	
7	694	9		1.272	10	
8	5.282	18	< 1 sec	10.182	21	< 1 sec
9	46.066	35	9 sec	90.822	48	21 sec
10	456.454	75	7 min	908.160	120	21 min
11	4.999.004	159	12 hours	9.980.160	312	64 hours*
12	59.916.028	355	84 days*	119.761.980	864	—

7 Further non-embeddable examples

In this section, we present further pairs of (unordered) n-vertex trees and sets of n points for $n = 13, 14, 16, 17, 18, 19, 20$, which do not admit an L-shaped embedding. The trees T_n are obtained as subtrees of the tree shown in Figure 17, by taking the subgraph induced by all unlabeled vertices and the vertices with labels $\leq n$. The corresponding point sets are encoded below in staircase notation. Note that all those staircase point sets have rotation and reflection symmetry and boxes of size at most 3. The fact that those instances do not allow an L-shaped embedding was established with computer help via a SAT solver, as described below.

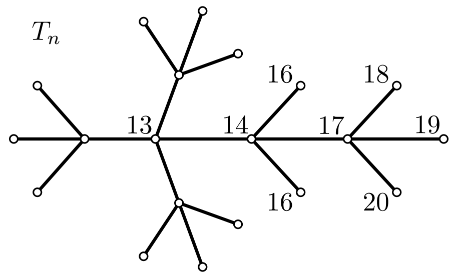


Figure 17: The 20-vertex tree T_{20}.

$n = 13$:

```
(1,1,2,2,1,2,2,1,1)
(1,1,3,1,1,1,3,1,1)
(2,2,2,1,2,2,2)
(2,3,1,1,1,3,2)
```

$n = 14$:

```
(1,1,2,1,2,2,1,2,1,1)
(2,2,1,2,2,1,2,2)
```

$n = 16$:

```
(1,3,1,1,1,2,1,1,1,3,1)
(1,3,2,1,2,1,2,3,1)
```

$n = 17$:

```
(1,1,3,1,1,3,1,1,3,1,1)
```

$n = 18$:

```
(1,1,2,1,1,1,2,2,1,1,1,2,1,1)
```

$n = 19$:

```
(1,1,3,1,1,1,3,1,1,1,3,1,1)
(1,1,3,1,2,3,2,1,3,1,1)
(1,1,3,2,1,3,1,2,3,1,1)
(2,3,1,1,1,3,1,1,1,3,2)
(2,3,1,2,3,2,1,3,2)
(2,3,2,1,3,1,2,3,2)
```

$n = 20$**:**

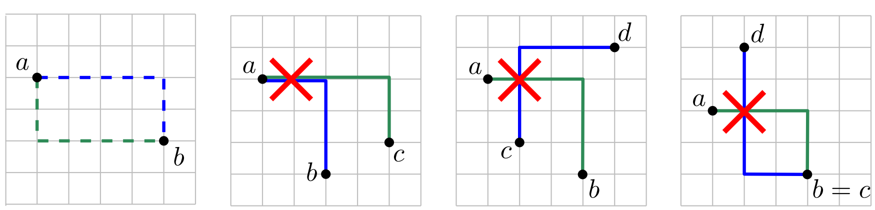
```
(1,1,2,1,1,1,2,2,2,1,1,1,2,1,1)
(1,1,2,1,2,2,2,2,2,1,2,1,1)
(1,1,2,2,1,2,2,2,1,2,2,1,1)
(2,2,1,1,1,2,2,2,1,1,1,2,2)
(2,2,1,2,2,2,2,2,1,2,2)
(2,2,2,1,2,2,2,1,2,2,2)
```

7.1 The SAT model

To test if a given tree with vertex set $\{1, \ldots, n\}$ admits an L-shaped embedding in a given point set $\{P_1, \ldots, P_n\}$, we formulated a Boolean satisfiability problem that has a solution if and only if the tree admits an embedding in the point set.

Our SAT model has variables $x_{i,j}$ to indicate whether the vertex i is mapped to the point P_j, and for every edge ab in the tree a variable $y_{a,b}$ to indicate whether the edge is connected horizontally to a (otherwise it is connected vertically to a). The constraints listed in the following are necessary and sufficient to guarantee the existence of an L-shaped embedding:

- **Injective mapping from vertices to points:** Each vertex is mapped to a point, and no two vertices are mapped to the same point.
- **L-shaped edges:** For each edge ab of the tree, the vertex a is either connected horizontally or vertically to b. Figure 18(a) gives an illustration.
- **No overlapping edge segments:** For each pair of incident edges ab and ac, if b and c are mapped to the right of a, then a cannot be connected horizontally to both b and c. An analogous statement holds if b and c are both mapped to the left, above, or below a. Figure 18(b) gives an illustration.
- **No crossing edge segments:** For each pair of edges ab and cd, the vertices a, b, c, d must not be mapped so that segments cross. More specifically, for each four points p, q, r, s (to which a, b, c, d may map), there are at most four cases that have to be forbidden in the mapping, depending on the relative position of p, q, r, s. Figures 18(c) and 18(d) give an illustration.



(a) (b) (c) (d)

Figure 18: Illustration of the constraints of the SAT model.

The resulting CNF formula thus has $\Theta(n^2)$ variables and $\Theta(n^4)$ clauses. Our Python program that creates a SAT instance for a given pair of tree and staircase point set is available as supplementary material to this paper or on the websites [10]. We used the SAT solver PicoSAT [4], which allows enumeration of all solutions. We also made use of pycosat, which provides Python bindings to PicoSAT.

8 Open problems

We currently do not know of any infinite family of (unordered) trees which do not always admit an L-shaped embedding. However, we conjecture that the instance in Figure 4 is such a family when considering the tree as unordered. Moreover, since all non-embeddable examples that we know are trees with path-width 2 (lobsters), it would be interesting to know whether trees with path-width 1 (caterpillars) always admit an L-shaped embedding. So far all known non-embeddable trees have maximum degree 4, so the question for trees with maximum degree 3 remains open [5, 6, 7].

A more general class of embeddings are *orthogeodesic* embeddings, where the edges are drawn with minimal ℓ_1-length and consist of segments along the grid induced by the point set [2, 5, 8, 13]. The best known bounds are due to Bárány et al. [2] who showed that every n-vertex tree with maximum degree 4 admits an orthogeodesic embedding in every point set of size $\lfloor 11n/8 \rfloor$. Unfortunately, the tree T_{13}, which we proved not to admit an L-shaped embedding on the point S_{13}, does admit an orthogeodesic embedding on S_{13} (see Figure 19), so the question whether n points are always sufficient to guarantee an orthogeodesic embedding of any n-vertex tree [2, 5] also remains open.

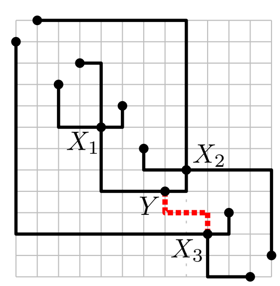


Figure 19: An orthogeodesic embedding of the tree T_{13} in the point set S_{13}. The only edge with two turns (not L-shaped) is drawn dotted.

Acknowledgements

We gratefully acknowledge the computing time granted by TBK Automatisierung und Messtechnik GmbH. We also thank the anonymous referees for several thoughtful comments that helped improving the presentation of this paper.

References

[1] O. Aichholzer, T. Hackl, and M. Scheucher. Planar L-shaped point set embeddings of trees. In *Proc. 32nd European Workshop on Computational Geometry (EuroCG 2016)*, pages 51–54, 2016. URL: http://www.eurocg2016.usi.ch/sites/default/files/paper_26.pdf.

[2] I. Bárány, K. Buchin, M. Hoffmann, and A. Liebenau. An improved bound for orthogeodesic point set embeddings of trees. In *Proc. 31nd European Workshop on Computational Geometry (EuroCG 2015)*, pages 47–50, 2016. URL: http://www.eurocg2016.usi.ch/sites/default/files/paper_44.pdf.

[3] T. Biedl, T. M. Chan, M. Derka, K. Jain, and A. Lubiw. Improved bounds for drawing trees on fixed points with L-shaped edges. In *Graph drawing and network visualization*, volume 10692 of *Lecture Notes in Comput. Sci.*, pages 305–317. Springer, Cham, 2018. doi:10.1007/978-3-319-73915-1_24.

[4] A. Biere. PicoSAT essentials. *Journal on Satisfiability, Boolean Modeling and Computation (JSAT)*, 4:75–97, 2008. URL: http://satassociation.org/jsat/index.php/jsat/article/view/45.

[5] E. Di Giacomo, F. Frati, R. Fulek, L. Grilli, and M. Krug. Orthogeodesic point-set embedding of trees. *Comput. Geom.*, 46(8):929–944, 2013. doi:10.1016/j.comgeo.2013.04.003.

[6] M. Fink, J.-H. Haunert, T. Mchedlidze, J. Spoerhase, and A. Wolff. Drawing graphs with vertices at specified positions and crossings at large angles. In *WALCOM: algorithms and computation*, volume 7157 of *Lecture Notes in Comput. Sci.*, pages 186–197. Springer, Heidelberg, 2012. doi:10.1007/978-3-642-28076-4_19.

[7] M. Kano and K. Suzuki. Geometric graphs in the plane lattice. In *Computational geometry*, volume 7579 of *Lecture Notes in Comput. Sci.*, pages 274–281. Springer, Cham, 2011. doi:10.1007/978-3-642-34191-5_26.

[8] B. Katz, M. Krug, I. Rutter, and A. Wolff. Manhattan-geodesic embedding of planar graphs. In *Graph drawing*, volume 5849 of *Lecture Notes in Comput. Sci.*, pages 207–218. Springer, Berlin, 2010. doi:10.1007/978-3-642-11805-0_21.

[9] B. D. McKay and A. Piperno. Practical graph isomorphism, II. *J. Symbolic Comput.*, 60:94–112, 2014. doi:10.1016/j.jsc.2013.09.003.

[10] T. Mütze and M. Scheucher. Personal websites with source code of our programs. http://tmuetze.de and http://page.math.tu-berlin.de/~scheucher/suppl/lshapes/.

[11] T. Mütze and M. Scheucher. On L-shaped point set embeddings of trees: First non-embeddable examples. In *Graph Drawing and Network Visualization - 26th International Symposium, GD 2018, Barcelona, Spain, September 26-28, 2018, Proceedings*, pages 354–360, 2018. doi:10.1007/978-3-030-04414-5_25.

[12] N. Robertson and P. D. Seymour. Graph minors. I. Excluding a forest. *J. Combin. Theory Ser. B*, 35(1):39–61, 1983. doi:10.1016/0095-8956(83)90079-5.

[13] M. Scheucher. Orthogeodesic point set embeddings of outerplanar graphs. Master's thesis, Graz University of Technology, 2015. URL: http://page.math.tu-berlin.de/~scheucher/publ/masters_thesis_2015.pdf.

[14] W. A. Stein et al. *Sage Mathematics Software (Version 8.1)*. The Sage Development Team, 2018. http://www.sagemath.org.