

On the Edge-Length Ratio of Outerplanar Graphs

Sylvain Lazard¹, William Lenhart², and Giuseppe Liotta³(✉)

¹ Inria, CNRS, University of Lorraine, Nancy, France
sylvain.lazard@inria.fr

² Williams College, Williamstown, USA
wlenhart@williams.edu

³ University of Perugia, Perugia, Italy
giuseppe.liotta@unipg.it

Abstract. We show that any outerplanar graph admits a planar straight-line drawing such that the length ratio of the longest to the shortest edges is strictly less than 2. This result is tight in the sense that for any $\epsilon > 0$ there are outerplanar graphs that cannot be drawn with an edge-length ratio smaller than $2 - \epsilon$. We also show that every bipartite outerplanar graph has a planar straight-line drawing with edge-length ratio 1, and that, for any $k \geq 1$, there exists an outerplanar graph with a given combinatorial embedding such that any planar straight-line drawing has edge-length ratio greater than k .

1 Introduction

The problem of computing a planar straight-line drawing with prescribed edge lengths has been addressed by several authors, partly for its theoretical interest and partly for its application in different areas, including VLSI, wireless sensor networks, and computational geometry (see for example [6, 7, 9, 13]). Deciding whether a graph admits a straight-line planar drawing with prescribed edge lengths was shown to be NP-hard by Eades and Wormald for 3-connected planar graphs [8]. In the same paper, the authors show that it is NP-hard to determine whether a 2-connected planar graph has a *unit-length* planar straight-line drawing; that is, a drawing in which all edges have the same length. Cabello et al. extend this last result by showing that it is NP-hard to decide whether a 3-connected planar graph admits a unit-length planar straight-line drawing [3]. In addition, Bhatt and Cosmadakis prove that deciding whether a degree-4 tree has a planar drawing such that all edges have the same length and the vertices are at integer grid points is also NP-hard [2].

These hardness results have motivated the study of relaxations and variants of the problem of computing straight-line planar drawings with constraints on the edge lengths. For example, Aichholzer et al. [1] study the problem of computing straight-line planar drawings where, for each pair of edges of the input graph G , it is specified which edge must be longer. They characterize families of graphs

that are *length universal*, i.e. they admit a planar straight-line drawing for any given total order of their edge lengths.

Perhaps one of the most natural variants of the problem in the context of graph drawing is that where, instead of imposing constraints on the edge lengths, one aims at computing planar straight-line drawings where the variance of the lengths of the edges is minimized. See for example [5], where this optimization goal is listed among the most relevant aesthetics that impact the readability of a drawing of a graph. Computing straight-line drawings where the ratio of the longest to the shortest edge is close to 1 also arises in the approximation of unit disk graph representations, a problem of interest in the area of wireless communication networks (see, e.g. [4, 11]).

Discouragingly, Eades and Wormald observe in their seminal paper that the NP-hardness of computing 2-connected planar straight-line drawings with unit edge lengths persists even when a small tolerance (independent of the problem size) in the length of the edges is allowed. To our knowledge, little progress has been made on bounding the ratio between the longest and shortest edge lengths in planar straight-line drawings. We recall the work of Hoffmann et al. [10], who compare different drawing styles according to different quality measures including the edge-length variance.

In this paper we study planar straight-line drawings of outerplanar graphs that bound the ratio of the longest to the shortest edge lengths from above by a constant. We define the *planar edge-length ratio* of a planar graph G as the smallest ratio between the longest and the shortest edge lengths over all planar straight-line drawings of G . The main result of the paper is the following.

Theorem 1. *The planar edge-length ratio of an outerplanar graph is strictly less than 2. Also, for any given real positive number ϵ , there exists an outerplanar graph whose planar edge-length ratio is greater than $2 - \epsilon$.*

Informally, Theorem 1 establishes that 2 is a tight bound for the planar edge-length ratio of outerplanar graphs. The upper bound is proved by using a suitable decomposition of an outerplanar graph into subgraphs called *strips*, then drawing the graph strip by strip. The lower bound is proved by taking into account all possible planar embeddings of a maximal outerplanar graph whose maximum vertex degree is a function of ϵ . Theorem 1 naturally suggests some interesting questions that are discussed in Sect. 3.

We shall assume familiarity with basic definitions of graph planarity and of graph drawing [5] and introduce only the terminology and notation that is strictly needed for our proofs.

2 Proof of Theorem 1

It suffices to establish the result for maximal outerplanar graphs. To show that the edge-length ratio of a maximal outerplanar graph is always less than 2, we imagine decomposing the dual G^* of G into a set of disjoint paths, which we call *chains*. Each chain corresponds to some sequence of pairwise-adjacent

triangles of G . The set of chains inherits a tree structure from G^* , and we use this structure to direct an algorithm that draws each of the chains proceeding from the root of this tree down to its leaves. We formally define a chain to be a sequence T_s, T_{s+1}, \dots, T_t of triangles of G where $s \leq 0 \leq t$, and such that (i) T_0 consists of an outer edge of G whose vertices are labeled with 0, along with a third vertex labeled 1, (ii) for each $i : 1 \leq i \leq t$, the vertices of T_i are labeled by $\{i-1, i, i+1\}$ so that T_i and T_{i-1} share the edge having vertices labeled i and $i-1$, and (iii) for each $i : -1 \geq i \geq s$, the vertices of T_i are labeled by $\{i, i+1, i+2\}$ so that T_i and T_{i+1} share the edge having vertices labeled $i+1$ and $i+2$. Note that this definition prohibits *fans* (consecutive triangles all sharing a common vertex) containing more than 3 triangles, except for the vertex labeled 1, which has four incident triangles on the chain.

The decomposition into chains is constructed by first selecting an edge e' on the outer face of some outerplanar topological embedding of G . The edge e' is incident with a unique triangle of G . Label each vertex of e' with 0, and label the third vertex of the triangle with 1. There is now a unique maximal chain $C_{e'}$ in G containing this labeled triangle. The edges of $C_{e'}$ can be partitioned into two sets: $S_{e'}$ and $L_{e'}$ where $S_{e'}$ consists of edges of $C_{e'}$ whose vertex labels differ by 1 and $L_{e'}$ consists of all edges of $C_{e'}$ whose vertex labels differ by 2, along with e' .

Removing the edges of $S_{e'}$ from G produces a set of 2-connected components in 1-1 correspondence with the edges of $L_{e'}$: Each component contains exactly one element of $L_{e'}$ which lies on its outer face. For each edge $e \in L_{e'}$, let G_e be the component of $G - S_{e'}$ containing e . We can then recursively decompose each G_e by choosing the (unique) maximal chain C_e in G_e containing the one triangle (if any) of G_e that is incident with e . We call the set of chains so constructed a *chain decomposition of G* . A chain decomposition produces a decomposition of the edges of G into sets L and S , where L is the union of the edges in each L_e and S is the union of all of the edges in each S_e . Note that there is a single chain for each edge in L , and that the collection of chains produced naturally form a tree: The root of the tree is the chain $C_{e'}$ and its children are the chains $\{C_e : e \in L_{e'}\}$; the chain decomposition is entirely determined by the choice of external edge e' .

The drawing algorithm proceeds by first drawing the root chain $C_{e'}$ of the chain decomposition tree of G and then recursively drawing the chain decomposition trees of each $G_e : e \in L_{e'}$. The algorithm depends on a specific method for drawing a single chain. To describe it, we need a few definitions. First, given a line segment s in the plane and a direction (unit vector) \mathbf{d} not parallel to s , denote by $S(s, \mathbf{d})$ the half-infinite strip bounded by s and the two infinite rays in direction \mathbf{d} that have their sources at the endpoints of s . Finally, given a chain C , the edges of $C \cap L$ are called *external edges* of C ; note that each external edge is incident to exactly one triangle of C .

Lemma 1. *Given a chain C with n vertices, an external edge e of C , a segment s of length 1 in the plane, and a direction \mathbf{d} such that the (smaller) angle between s and \mathbf{d} is $\theta < \theta_0 = \arccos(1/4) \approx 75.5^\circ$, there exists a planar straight-line drawing*

of C such that: (i) the drawing is completely contained within the strip $S(s, \mathbf{d})$; (ii) no external edge e' of C is parallel to \mathbf{d} , and the strips $S(e', \mathbf{d})$ are all empty; (iii) each external edge has length 1 and all other edges have lengths greater than $1/2$; (iv) each external edge forms an angle less than θ_0 with \mathbf{d} . Moreover, such a drawing can be computed in $O(n)$ -time in the real RAM model.

Proof. Let T_0 be the triangle of C containing e . T_0 is either adjacent to zero, one, or two triangles of C . We handle these three cases in turn. If T_0 is the only triangle in C , then we simply draw T_0 in $S(s, \mathbf{d})$ as an isosceles triangle with e drawn as s and with its third vertex drawn so that its two edges have length l , where $\frac{1}{2} < l < 1$.

Assume now that T_0 is adjacent to a triangle T_1 of C . Denote the vertices of C as follows: $T_0 = \{v_0^-, v_0^+ v_1\}$, where $e = \{v_0^-, v_0^+\}$ and v_0^- is not incident with T_1 . The vertex of T_1 not in T_0 is denoted by v_2 , and, subsequently, the vertex of each T_i not in T_{i-1} is denoted by v_{i+1} . We draw T_0 as previously, but with more careful positioning of v_1 . To determine where to position v_1 , we draw edge $\{v_0^+, v_2\}$ of T_1 as a unit-length segment in direction \mathbf{d} . As long as v_1 is positioned within $S(s, \mathbf{d})$ but outside of the disks of radius $\frac{1}{2}$ centered at v_0^-, v_0^+ , and v_2 , the edges from each of these vertices to v_1 will have length greater than $\frac{1}{2}$ (see top half of Fig. 1). By placing v_1 close to e , the edges $\{v_1, v_0^-\}$ and $\{v_1, v_0^+\}$ will have lengths less than 1. Also, since $\angle v_2 v_0^+ v_1 < \theta_0$, edge $\{v_1, v_2\}$ will have length less than 1.

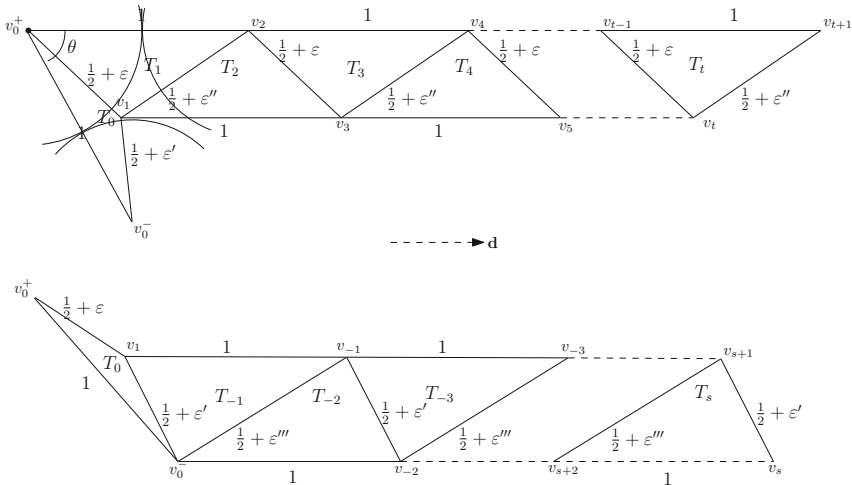


Fig. 1. Drawing a chain

Assuming that T_0, \dots, T_{i-1} have been drawn for some $i > 1$, T_i is drawn by positioning v_{i+1} one unit distant from v_{i-1} in direction \mathbf{d} . The result is that each T_i is congruent to T_1 and so the edge-length ratio of C is less than 2.

At this point, all of the unit-length segments, except for e , lie on the two rays in direction \mathbf{d} emanating from v_0^+ and v_1 . By rotating these rays a very small amount towards one another, we can preserve the lengths of the unit-length segments while ensuring that all of the remaining segments have lengths in the range $(\frac{1}{2}, 1)$. See the top of Fig. 1.

Finally, suppose that T_0 has two adjacent triangles. Starting with T_0 , label the other triangles in C so that the labels of adjacent triangles differ by 1. Thus, for example, T_0 is adjacent to T_1 and T_{-1} . The vertices in the $T_i, i \geq 0$ will be labeled as in the previous case: the unique vertex in T_i not in T_{i-1} is labeled v_{i+1} . The vertices in the $T_i, i < 0$ will be similarly labeled: the unique vertex in T_i not in T_{i+1} is labeled v_i .

Draw each $T_i, i > 0$ as in the previous case. Now draw the $T_i, i < 0$ in a similar fashion: Place vertex v_i one unit distant from v_{i+2} in direction \mathbf{d} . Then, as above, all of the unit length edges of the triangles $T_i, i < 0$ will lie on the two rays in direction \mathbf{d} emanating from v_1 and v_0^- , and these two rays can be rotated slightly towards each other while maintaining the length of the unit-length edges and ensuring that the other edges still have lengths in the range $(\frac{1}{2}, 1)$. See the bottom of Fig. 1. This can clearly be done so that all external edges form angles less than θ_0 with \mathbf{d} .

However, we need to ensure that v_1 can be placed so that both triangle T_1 and triangle T_{-1} can simultaneously satisfy the required edge-length conditions: Namely, that edges $\{v_0^-, v_0^+\}, \{v_1, v_{-1}\}$, and $\{v_0^+, v_2\}$ are all unit-length, while edges $\{v_1, v_2\}, \{v_0^+, v_1\}, \{v_1, v_0^-\}$, and $\{v_0^-, v_{-1}\}$ all have lengths in the range $(\frac{1}{2}, 1)$. However, it is relatively simple to show that v_1 can always be successfully placed if the (smaller) angle between s and \mathbf{d} is less than $\theta_0 = \arccos(1/4)$. [The angle θ_0 is the angle opposite an edge of length 2 in an isosceles triangle having side lengths 2, 2, and 1.] Finally, the computation of the locations of the vertices can each be computed in constant time in the real RAM model, giving a run-time linear in the size of the chain. \square

We are now ready to prove the following lemma. For a planar straight-line drawing Γ , we denote with $\rho(\Gamma)$ the ratio of a longest to a shortest edge in Γ .

Lemma 2. *A maximal outerplanar graph G with n vertices admits a planar straight-line drawing Γ with $\rho(\Gamma) < 2$ that can be computed in $O(n)$ time assuming the real RAM model of computation.*

Proof. We call the drawing computed as in Lemma 1 a *U-strip drawing* of C and adopt the same notation as in Lemma 1. Recall that in a chain decomposition of a graph, the external edges of the chains are exactly the edges of L . A drawing of G is computed as follows.

1. Compute a chain decomposition tree for G ; let $C_{e'}$ be the root of the tree.
2. Select a line segment s of length 1 in the plane and an initial direction \mathbf{d} not parallel to s such that $\angle s\mathbf{d} < \theta_0$.
3. Apply Lemma 1 to compute a U-strip drawing of $C_{e'}$.

4. Each edge $e \in L_{e'}$ is drawn as a segment s_e of length 1, not parallel to \mathbf{d} , that forms an angle with \mathbf{d} that is less than θ_0 , so draw the subtree of $C_{e'}$ rooted at C_e in the empty strip $S(s_e, \mathbf{d})$.

The result is an outerplanar straight-line drawing in which all edges of L (*long edges*) have length 1 while all edges in S (*short edges*) have length strictly greater than $\frac{1}{2}$. If we assume that the input is provided to the algorithm in the form of a doubly-connected edge list [12], then a chain decomposition tree for G can be computed in linear time. Also, since by Lemma 1 each chain can be drawn in time proportional to its length, the algorithm runs in $O(n)$ time in the real RAM model. \square

The following lemma can be proved by means of a packing argument and elementary geometry.

Lemma 3. *For any $\epsilon > 0$ there exists a maximal outerplanar graph whose planar edge length ratio is greater than $2 - \epsilon$.*

We conclude the section by observing that Lemmas 2 and 3 imply Theorem 1.

3 Additional Remarks and Open Problems

The upper and the lower bound of Theorem 1 suggest some questions that we find worth investigating. One question is whether better bounds on the planar edge-length ratio can be established for subfamilies of outerplanar graphs (for example, it is easy to show that trees have unit-length drawings). A second question is whether an edge length variance bounded by a constant can be guaranteed for drawings of outerplanar graphs where not all vertices lie in a common face. By a variant of the approach used to prove Lemma 2 and by using some simple geometric observations, the following results can be proved.

Theorem 2. *The planar edge-length ratio of a bipartite outerplanar graph is 1.*

The *plane edge-length ratio* of a planar embedding \mathcal{G} of a graph G is the minimum edge-length ratio taken over all embedding-preserving planar straight-line drawings of \mathcal{G} .

Theorem 3. *For any given $k \geq 1$, there exists an embedded outerplanar graph whose plane edge-length ratio is at least k .*

We conclude this paper by listing some open questions that we find interesting to study: (i) Study the edge-length ratio of triangle-free outerplanar graphs. For example, it is not hard to see that if all faces of an outerplanar graph have five vertices, a unit edge length drawing may not exist; however, the planar edge length ratio for this family of graphs could be smaller than the one established in Theorem 1. (ii) Extend the result of Theorem 1 to families of non-outerplanar graphs. For example it would be interesting to study whether the planar edge-length ratio of 2-trees is bounded by a constant. (iii) Study the complexity of deciding whether an outerplanar graph admits a straight-line drawing where the ratio of the longest to the shortest edge is within a given constant. This problem is interesting also in the special case that we want all edges to be unit length.

References

1. Aichholzer, O., Hoffmann, M., van Kreveld, M.J., Rote, G.: Graph drawings with relative edge length specifications. In: 2014 Proceedings of the 26th Canadian Conference on Computational Geometry, CCCG 2014, Halifax, Nova Scotia, Canada (2014)
2. Bhatt, S.N., Cosmadakis, S.S.: The complexity of minimizing wire lengths in VLSI layouts. *Inf. Process. Lett.* **25**(4), 263–267 (1987)
3. Cabello, S., Demaine, E.D., Rote, G.: Planar embeddings of graphs with specified edge lengths. *J. Graph Algorithms Appl.* **11**(1), 259–276 (2007)
4. Chen, J., Jiang, A.A., Kanj, I.A., Xia, G., Zhang, F.: Separability and topology control of quasi unit disk graphs. *Wirel. Netw.* **17**(1), 53–67 (2011)
5. Di Battista, G., Eades, P., Tamassia, R., Tollis, I.G.: *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice-Hall, Upper Saddle River (1999)
6. Di Battista, G., Vismara, L.: Angles of planar triangular graphs. *SIAM J. Discrete Math.* **9**(3), 349–359 (1996)
7. Doherty, L., Pister, K.S.J., El Ghaoui, L.: Convex optimization methods for sensor node position estimation. In: *Proceedings IEEE INFOCOM 2001, The Conference on Computer Communications, Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies, Twenty years into the communications odyssey, Anchorage, Alaska, USA, 22–26 April 2001*, pp. 1655–1663. IEEE (2001)
8. Eades, P., Wormald, N.C.: Fixed edge-length graph drawing is NP-hard. *Discrete Appl. Math.* **28**(2), 111–134 (1990)
9. Held, S., Korte, B., Rautenbach, D., Vygen, J.: Combinatorial optimization in VLSI design. In: Chvátal, V. (ed.) *Combinatorial Optimization - Methods and Applications*, NATO Science for Peace and Security Series - D: Information and Communication Security, vol. 31, pp. 33–96. IOS Press (2011)
10. Hoffmann, M., van Kreveld, M.J., Kusters, V., Rote, G.: Quality ratios of measures for graph drawing styles. In: 2014 Proceedings of the 26th Canadian Conference on Computational Geometry, CCCG 2014, Halifax, Nova Scotia, Canada (2014)
11. Kuhn, F., Moscibroda, T., Wattenhofer, R.: Unit disk graph approximation. In: Basagni, S., Phillips, C.A. (eds.) *Proceedings of the DIALM-POMC Joint Workshop on Foundations of Mobile Computing*, Philadelphia, PA, USA, 1 October 2004, pp. 17–23. ACM (2004)
12. Muller, D.E., Preparata, F.P.: Finding the intersection of two convex polyhedra. *Theoret. Comput. Sci.* **7**(2), 217–236 (1978)
13. Savarese, C., Rabaey, J.M., Langendoen, K.: Robust positioning algorithms for distributed ad-hoc wireless sensor networks. In: *Ellis, C.S. (ed.) Proceedings of the General Track: 2002 USENIX Annual Technical Conference, Monterey, California, USA, 10–15 June 2002*, pp. 317–327. USENIX (2002)