How to Draw Outerplanar Minimum Weight Triangulations * (extended abstract)

William Lenhart[1] and Giuseppe Liotta[2]

[1] Department of Computer Science, Williams College, Williamstown, MA 01267.
lenhart@cs.williams.edu
[2] Department of Computer Science, Brown University, 115 Waterman Street, Providence, RI 02912.
gl@cs.brown.edu

Abstract. In this paper we consider the problem of characterizing those graphs that can be drawn as minimum weight triangulations and answer the question for maximal outerplanar graphs. We provide a complete characterization of minimum weight triangulations of regular polygons by studying the combinatorial properties of their dual trees. We exploit this characterization to devise a linear time (real RAM) algorithm that receives as input a maximal outerplanar graph G and produces as output a straight-line drawing of G that is a minimum weight triangulation of the set of points representing the vertices of G.

1 Introduction

A widely used graph drawing standard represents vertices as points on the plane and edges as straight-line segments between points. Drawings that follow such a standard are called *straight-line drawings* and the design of algorithms to produce such drawings is a field of growing interest. An extensive survey of results on straight-line drawings as well as on other graphic standards is provided by Di Battista et al. [3]. Recently, attention has been devoted to a special type of straight-line drawings, called *minimum weight drawings*, which have applications in areas including computational geometry and numerical analysis.

Let C be a class of graphs, let P be a set of points in the plane. Let G be a graph such that

1. G has vertex set P,
2. the edges of G are straight-line segments connecting pairs of points of P,

* Research supported in part by the National Science Foundation under grant CCR-9423847, by the U.S. Army Research Office under grant 34990–MA–MUR, by Progetto Finalizzato Sistemi Informatici e Calcolo Parallelo of the Italian National Research Council (CNR), and by N.A.T.O.- CNR Advanced Fellowships Programme.

3. $G \in \mathcal{C}$, and
4. the sum of the lengths of the edges of G is minimized over all graphs satisfying 1–3.

We call such a graph G a *minimum weight representative of* $\mathcal{C}$. Given a graph $G \in \mathcal{C}$, we say that G has a *minimum weight drawing* if there exists a set P of points in the plane such that G is a minimum weight representative of $\mathcal{C}$. For example, a *minimum spanning tree* of a set P of points is a connected, straight-line drawing that has P as vertex set and minimizes the total edge length. So, letting $\mathcal{C}$ be the class of all trees, a tree G has a minimum weight drawing if there exists a set P of points in the plane such that G is isomorphic to a minimum spanning tree of P. A *minimum weight triangulation* of a set P is a triangulation of P having minimum total edge length. Letting $\mathcal{C}$ be the class of all planar triangulations, a planar triangulation G has a minimum weight drawing if there exists a set P of points in the plane such that G is isomorphic to a minimum weight triangulation of P.

The problem of testing whether a tree admits a minimum weight drawing is essentially solved. Monma and Suri [20] proved that each tree with maximum vertex degree at most five can be drawn as a minimum spanning tree of some set of vertices by providing a linear time (real RAM) algorithm. In the same paper it is shown that no tree having at least one vertex with degree greater than six can be drawn as a minimum spanning tree. As for trees having maximum degree equal to six, Eades and Whitesides [5] showed that it is NP-hard to decide whether such trees can be drawn as minimum spanning trees.

Surprisingly, nothing seems to be known about the problem of constructing a minimum weight drawing of a planar triangulation. Moreover, it is still not known whether computing a minimum weight triangulation of a set of points in the plane is an NP-complete problem (see Garey and Johnson [6]). Several papers have been published on this last problem, either providing partial solutions, or giving efficient approximation heuristics. A limited list of results includes the work by Meijer and Rappaport [19], Lingas [14, 16], Kirkpatrick [10], Keil [9], Dickerson et al. [4], and Aichholzer et al. [1].

In this paper we examine the problem of characterizing those triangulations admitting a minimum weight drawing and answer the question for maximal outerplanar graphs. The contribution is twofold:

1. Minimum weight triangulations for points that are vertices of a regular polygons are characterized. The characterization is based on the combinatorial structure of the dual tree of the minimum weight triangulations of such point sets. A consequence of the characterization is an optimal time algorithm for computing a minimum weight triangulation of a regular n-gon for any given n. Interestingly, it is not necessary to supply the algorithm with the actual points, only the size (number of vertices) of the polygon is needed. When arbitrary convex polygons are considered, the fastest known algorithms require $O(n^3)$ time, where n is the number of vertices of the polygon (see Gilbert [7], Klincsek [11], and Heath and Pemmaraju [8]).

The triangulation produced by our algorithm turns out to be that which would result from the application of the greedy algorithm described by Levcopoulos and Lingas [13]. Lloyd [17] has shown that the greedy triangulation of a convex polygon is not necessarily of minimum weight; lower bounds for the nonoptimality of the greedy triangulation are given by Manacher and Zobrist [18] and by Levcopoulos [12]. Lingas [15] shows that on average the greedy triangulation approximates the optimum by an $O(\log n)$ factor.

2. We show that every maximal outerplanar graph G has a minimum weight drawing. This is done by exhibiting an algorithm that computes a minimum weight drawing of G in time proportional to the number of vertices of G, within the real RAM model of computation. The drawing algorithm exploits the combinatorial properties of the minimum weight triangulations of regular polygons and is based on a decomposition rule of minimum weight triangulations.

2 Preliminaries

We assume familiarity with the basic terminology of graph theory and computational geometry (see also Bondy and Murty [2], and Preparata and Shamos [21]). A graph G is *outerplanar* if it has a planar embedding such that all vertices lie on a single face. G is *maximal outerplanar* if G is outerplanar, but the addition of any new edge results in a non-outerplanar graph. In geometric applications graphs often arise as the result of selecting a set S of points in the plane and then connecting certain pairs to be joined by straight line segments which form the edge set of the graph. In this paper, we will be interested in *triangulations* of a point set S: planar graphs obtained from S by taking as edge set a maximal number of mutually non-crossing straight line segments connecting pairs of points in S. In particular, a *triangulation of a regular n-gon P* is a triangulation obtained by adding $n-3$ mutually non-intersecting diagonals connecting pairs of vertices of P. Every triangulation G of a regular n-gon gives a straight-line drawing of a maximal outerplanar graph on n vertices, such that the outer face forms a regular n-gon; similarly, every maximal outerplanar graph on n vertices gives rise to a unique triangulation of a regular n-gon. We will refer to any graph as a triangulation if the graph is isomorphic to a triangulation of some point set S.

Given an embedded planar graph G, the *extended dual tree (or e-dual)* of G is a planar graph G' defined as follows. G' has a vertex for each internal face of G and a vertex for each of the edges on the external face of G. Two vertices u, v of G' are adjacent either if they correspond to two internal faces of G that share an edge, or if u corresponds to an edge e of G and v to a face of G containing e.

Note that if G is a triangulation of an n-gon, then G' is a tree with $2n-2$ vertices, with the property that every non-leaf vertex has degree 3. In the rest of the paper we assume that the e-dual of the triangulation of an n-gon is rooted at a non leaf-vertex r.

Let G be a triangulation of an n-gon, and let T be the extended dual of of G having root r. Observe that there is a natural way to associate the edges of G to all but vertices of T other than r: Each leaf of T corresponds to an external edge of the triangulation; each non-leaf $v \neq r$ of T will correspond to the third edge of the triangular face of G formed by v and its two children. Only r has no corresponding edge in G.

A (non minimum weight) triangulation of a regular 15-gon and its e-dual are shown in Figure 1. The vertex of the dual labeled r is the root of the tree; vertices labeled x, y and z will be of use in the rest of the paper.

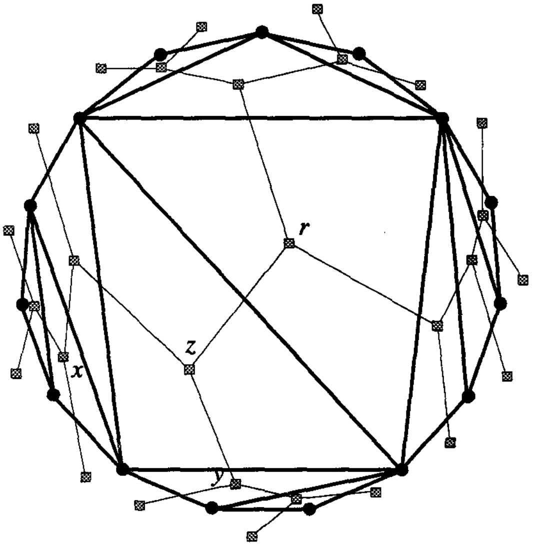

Fig. 1. A triangulation and its e-dual.

We will frequently be transforming rooted trees on the same number of vertices. It will be convenient to think of a given set V of vertices which has different trees defined on it. Each of these trees will be used to define several functions both on V and on subsets of V.

A rooted tree T defined on a set V is a *feasible* tree if every non-leaf has degree 3 and each subtree of the root has at most $|V|/2$ vertices. Clearly any e-dual of a triangulation of an n-gon is a feasible tree on $2n-2$ vertices. Conversely, any feasible tree on $2n-2$ vertices is an e-dual of some triangulation of a n-gon.

Let T be a rooted tree defined on a set V. Two vertices x and y are *incomparable* if neither x nor y is an ancestor of the other.

If T is a rooted tree defined on V, the subtree of T with root v is denoted by $T(v)$; the number of leaves in $T(v)$ is denoted by $l_T(v)$. If $x \in V$ and z is an ancestor of x in T, then the set of interior vertices on the path in T from x to z (i.e., all vertices except x and z) is denoted by $\pi(x,z)$. Note that if z is the parent of x then $\pi(x,z) = \emptyset$.

Let x and y be two vertices of a regular n-gon of radius 1. Let l be the number of edges on a path between x and y. Then the length of the diagonal from x to y is given by $2sin(\frac{\pi}{n}l)$. Note that this formula holds regardless of whether we used the short or the long path between x and y.

Let T be a rooted tree defined on V. For any vertex $v \in V$, the *weight* of v, $w_T(v)$, is defined to be $2sin(\frac{\pi}{n}l_T(v))$. Observe that if T is the (rooted) e-dual tree of a triangulation G of an n-gon, then $w_T(v)$ denotes the length of the edge of G corresponding to v. In particular, $w_T(r) = 2sin(\frac{\pi}{n}n) = 0$, which conveniently agrees with the earlier observation that r does not correspond to any edge of G. Thus the weight of the triangulation G is given by the sum of the weights of the vertices of T. Moreover, for any $X \subseteq V$, we define the *weight* of X, $W_T(X)$, to be $\sum_{v \in X} w_T(v)$. Thus, for example, $W_T(V)$ is the weight of the triangulation. We will also denote this value by $W(T)$ and refer to it as the *weight of* T. Finally, we will also need to consider the related functions $\overline{w}_T(v) = 2cos(\frac{\pi}{n}l_T(v))$ and $\overline{W}_T(X) = \sum_{v \in X} \overline{w}_T(v)$.

The problem of finding a minimum weight triangulation of a regular n-gon can be reformulated as follows: Given a set of $2n-2$ vertices, minimize $W(T)$ over all feasible trees T.

In the next section, we solve the minimum weight triangulation problem just mentioned. In Section 4, we exploit our results to prove that every maximal outerplanar graph G admits a *minimum weight drawing*; that is, a drawing of G such that the edges of the drawing are those of some minimum weight triangulation of the vertices of the drawing. The last section summarizes our conclusions and mentions some open problems.

3 Minimum Weight Triangulations of Regular Polygons

We begin by defining an operation on a feasible tree T defined on V, which transforms T to another feasible tree T'' on V. Let x and y be incomparable vertices of T and let z be the lowest common ancestor of x and y. A *swap* consists of two steps:

1. The subtrees $T(x)$ and $T(y)$ are exchanged, resulting in a tree denoted by T',
2. If T' is feasible, then $T'' = T'$, otherwise T'' is constructed by choosing a new root for T' to make it feasible.

Notice that $W(T') = W(T'')$. A swap is an *improvement* (or *improving swap*) if $W(T') < W(T)$.

Observe that the only vertices of T' whose weights change as a result of Step 1 are those vertices in $\pi(x,z) \cup \pi(y,z)$, since these are the only vertices v

such that $l_T(v) \neq l_{T'}(v)$. Thus, $W(T')-W(T) = (W_{T'}(\pi(x,z))-W_T(\pi(x,z)))+(W_{T'}(\pi(y,z)) - W_T(\pi(y,z)))$.

Lemma 1. *Let T be a feasible tree and let x and y be incomparable vertices of T such that $l_T(x) \neq l_T(y)$ and having lowest common ancestor z. The swap of $T(x)$ and $T(y)$ is an improvement if either $\overline{W}_T(\pi(x,z)) - \overline{W}_T(\pi(y,z))) = 0$, or $(l_T(x) - l_T(y)) \times (\overline{W}_T(\pi(x,z)) - \overline{W}_T(\pi(y,z))) > 0$.*

Proof. Let $x_i \in \pi(x,z)$, $y_j \in \pi(y,z)$ and $\Delta l = l_T(y) - l_T(x)$. Observe that $l_{T'}(x_i) = l_T(x_i) + \Delta l$, and $l_{T'}(y_j) = l_T(y_j) - \Delta l$.

Now, $W(T') - W(T) = \sum_{i=1}^{k}(w_{T'}(x_i) - w_T(x_i)) + \sum_{j=1}^{m}(w_{T'}(y_j) - w_T(y_j))$. Using the definition of $w_T()$ and properties of the sine function, we have,

$$\begin{aligned} w_{T'}(x_i) &= 2sin(\frac{\pi}{n}l_{T'}(x_i)) \\ &= 2sin(\frac{\pi}{n}(l_T(x_i) + \Delta l)) \\ &= 2sin(\frac{\pi}{n}l_T(x_i))cos(\frac{\pi}{n}\Delta l) + 2cos(\frac{\pi}{n}l_T(x_i))sin(\frac{\pi}{n}\Delta l) \\ &= w_T(x_i)cos(\frac{\pi}{n}\Delta l) + \overline{w}_T(x_i)sin(\frac{\pi}{n}\Delta l). \end{aligned}$$

Similarly, $w_{T'}(y_j) = w_T(y_j)cos(\frac{\pi}{n}\Delta l) - \overline{w}_T(y_j)sin(\frac{\pi}{n}\Delta l)$.
Thus,

$$\begin{aligned} W(T') - W(T) &= \sum_{i=1}^{k}((cos(\frac{\pi}{n}\Delta l) - 1)w_T(x_i) + sin(\frac{\pi}{n}\Delta l)\overline{w}_T(x_i)) + \\ &\quad \sum_{j=1}^{m}((cos(\frac{\pi}{n}\Delta l) - 1)w_T(y_j) - sin(\frac{\pi}{n}\Delta l)\overline{w}_T(y_j)) \\ &= (cos(\frac{\pi}{n}\Delta l) - 1)(W_T(\pi(x,z)) + W_T(\pi(y,z))) + \\ &\quad sin(\frac{\pi}{n}\Delta l)(\overline{W}_T(\pi(x,z)) - \overline{W}_T(\pi(y,z))) \end{aligned}$$

The first term is negative since $cos(\frac{\pi}{n}\Delta l) - 1 < 0$. The second term is non-positive since $sin(\frac{\pi}{n}\Delta l)$ has the same sign as Δl, and since, by assumption, either $\overline{W}_T(\pi(x,z)) = \overline{W}_T(\pi(y,z))$ or Δl has the opposite sign of $\overline{W}_T(\pi(x,z)) - \overline{W}_T(\pi(y,z))$. Thus $W(T') - W(T) < 0$ and the swap is improving. □

The preceding result is used to establish the following two corollaries.

Corollary 2. *Let T be a feasible tree and let x and y be incomparable non-leaf vertices of T having children x', x'' and y', y'' respectively. If $l_T(x') > l_T(y')$ and $l_T(x'') < l_T(y'')$, then T admits an improving swap.*

Proof. Let z be the lowest common ancestor of x and y. The hypotheses of the corollary imply that $l_T(x') - l_T(y')$ and $l_T(x'') - l_T(y'')$ have opposite signs. Therefore one of these two differences has the same sign as $\overline{W}_T(\pi(x',z)) -$

$\overline{W}_T(\pi(y', z))$. Since $\pi(x', z) = \pi(x'', z)$ and $\pi(y', z) = \pi(y'', z)$, we can apply Lemma 1, to see that either swapping Tx' with Ty' or swapping Tx'' with Ty'' must decrease the weight of the tree. Thus, we have made an improving swap. □

As an example, vertices x and y of Figure 1 are two incomparable non-leaf vertices for which there is an improving swap.

Corollary 3. *Let T be a feasible tree and let z be a vertex having children x and y such that y has children y', y''. If $l_T(x) < l_T(y')$, then T admits an improving swap.*

Proof. Since the path from x to z is empty, $\overline{W}_T(\pi(x, z)) = 0$; also $\overline{W}_T(\pi(y', z)) = \overline{W}_T(\pi(y'', z)) = \overline{w}_T(y) \geq 0$, since the subtree rooted at y has at most half of the leaves in the tree. Thus $\overline{W}_T(\pi(x, z)) - \overline{W}_T(\pi(y', z)) \leq 0$. Therefore, if $l_T(x) - l_T(y') < 0$, there is an improving swap by Lemma 1. □

We now have two operations which can be applied to feasible trees in order to decrease their weight. It turns out that these two operations suffice to transform any feasible tree to a minimum weight tree.

A *weight-balanced tree* is a feasible tree which admits no improving swaps.

Clearly every minimum-weight tree is weight-balanced. The rest of this section consists in showing that every weight-balanced tree is minimum-weight.

To accomplish this, we show that every weight-balanced tree can be put into a standard form by repeatedly swapping left and right subtrees of vertices of the tree. Clearly swapping the left and right subtrees of a given vertex does not change the weight of the tree, since it does not change the weight of the vertex. Since every weight-balanced tree can be put into this form, all weight-balanced trees on V must have the same weight. This weight must be minimum since all minimum weight trees are themselves weight-balanced.

A weight-balanced tree T is *sorted* if, in a breadth-first, left-to-right traversal of T, the function $l_T()$ is not increasing.

Lemma 4. *There is only one sorted weight-balanced tree T having $2n - 2$ vertices.*

Proof. The vertices on each level are sorted by decreasing $l_T()$-value; if a level contains a leaf v, all vertices to the right of v on that level are also leaves. Since all vertices on lower levels have $l_T()$-values no greater than $l_T(v)$, all vertices on the next level down are also leaves. Because all feasible trees on $2n - 2$ vertices have exactly n leaves, this uniquely determines the structure of the tree. □

Let T be a weight-balanced tree. It is clear that we can arrange T so that each vertex has its children sorted from left to right by decreasing $l_T()$-value. It turns out that, once this is accomplished, the resulting tree is sorted. We show this in two steps.

Lemma 5. *Let T be a weight-balanced tree, and let x and y be incomparable vertices of T having children x_L, x_R and y_L, y_R respectively, such that $l_T(x_L) \geq l_T(x_R)$ and $l_T(y_L) \geq l_T(y_R)$. If $l_T(x) \geq l_T(y)$, then $l_T(x_R) \geq l_T(y_L)$.*

Proof. By assumption, $l_T(x_L) + l_T(x_R) = l_T(x) \geq l_T(y) = l_T(y_L) + l_T(y_R)$. So, if $l_T(x_R) < l_T(y_L)$, then it must be that $l_T(x_L) > l_T(y_R)$. Therefore, by Corollary 2, T admits an improving swap, contradicting the assumption that T is weight-balanced. □

Lemma 6. *If T is a weight-balanced tree, such that the children of each vertex are sorted left-to-right by decreasing $l_T()$-value, then T is sorted.*

Proof. The proof consists of two parts.

1. We first show that vertices at same depth have $l_T()$-values decreasing from left to right.
2. We then show that the right-most vertex at depth k has $l_T()$-values at least as large as that of the left-most vertex at depth $k+1$.

We prove the first claim by induction on the depth of vertices in the tree. The base case is simple, the vertices at depth one are simply the children of the root, and so are sorted by decreasing $l_T()$-value. Suppose now, that all vertices at depth $k-1$ are sorted by decreasing $l_T()$-value, and consider now the vertices at depth k. Let x and y be two consecutive vertices, in left-to-right order, at depth $k-1$. By induction, $l_T(x) \geq l_T(y)$; by hypothesis, the children of x are in decreasing $l_T()$-value order, as are the children of y. Thus, by Lemma 5, the children of x and y together are in decreasing $l_T()$-value order. Since this holds for all pairs of consecutive vertices at depth $k-1$, all vertices at depth k must also be sorted.

We also prove the second claim by induction on the depth of vertices in the tree. The base case is easy to establish (by contradiction using Corollary 3), so we show only the induction step. Let x_i and y_i denote the left-most and right-most vertices at depth i respectively. Assume that, for all $i < k$, y_i has $l_T()$-value at least as great as that of x_{i+1}. Thus $\overline{w}_T(x_{i+1}) \geq \overline{w}_T(y_i)$. Let $y = y_k$ be the right-most vertex at depth k, let $x = x_{k+1}$ be the left-most vertex at depth $k+1$, and let r be the root of T. Then $\overline{W}_T(\pi(x,r)) > \overline{W}_T(\pi(y,r))$, since $\overline{w}_T(x_1) > 0$ and $\overline{w}_T(x_{i+1}) \geq \overline{w}_T(y_i)$, for each $i < k$. Since T admits no improving swaps, by Lemma 1, it must be that $l_T(x) \leq l_T(y)$. □

Lemmas 4–6 immediately yield the following result.

Theorem 7. *A triangulation G of a regular n-gon is minimum weight if and only if its e-dual can be rooted so that it is weight-balanced.*

Figure 2 shows a minimum weight triangulation of a regular 15-gon; r designates the root of the e-dual.

We can use our understanding of the structure of weight-balanced trees to design an optimal algorithm for computing a minimum weight triangulation of

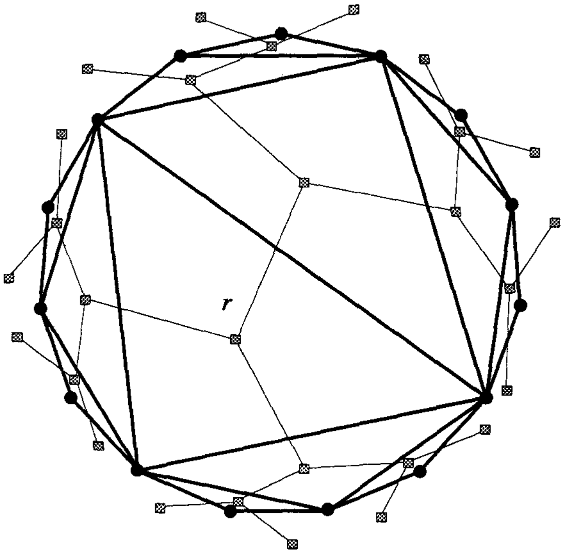

Fig. 2. A minimum weight triangulation of a regular 15-gon.

a regular n-gon. Let T be the sorted, weight-balanced tree on $2n-2$ vertices. T induces a triangulation of the regular n-gon which, by Theorem 7 is a minimum weight triangulation. Observe that the triangulation is the same as the one which would be obtained by an application of the greedy algorithm (always selecting the smallest segment which does not cross any of the segments selected so far). Moreover, this triangulation can be computed in time proportional to the size of T. We summarize this in the following:

Theorem 8. *A minimum weight triangulation of a regular n-gon can be computed in time proportional to the size of the triangulation produced. Furthermore, the triangulation obtained is the same as that obtained by the greedy algorithm.*

4 Minimum Weight Drawings of Maximal Outerplanar Graphs

Before proving the main result of this section, we note a feature of minimum weight triangulations.

Lemma 9. *Let G be a minimum weight triangulation of a set P of points in the plane. Let Δabc be an interior face of G such that $\overline{ab}$ and $\overline{ac}$ are on the outer face of G. Then $E(G) - \{\overline{ab}, \overline{ac}\}$ is a minimum weight triangulation of $P - \{a\}$, where $E(G)$ denotes the set of edges of G.*

Using the e-dual of a embedded maximal outerplanar graph allows us to compute a minimum weight drawing of the graph.

Theorem 10. *Let G be a maximal outerplanar graph on n vertices. A minimum weight drawing of G can be computed in $O(n)$ time in the real RAM model.*

Proof. We begin by considering an embedded maximal outerplanar graph G with all vertices the outer face, and then computing the e-dual T of G. Let T' be any feasible weight-balanced tree containing T as a sub-tree, and let N be the number of leaves of T'. T' gives rise to a minimum-weight drawing Γ of a triangulated regular N-gon. A minimum weight drawing of G can now be obtained by repeated application of Lemma 9 to Γ. Figure 3(b) shows an example of such construction, when the input is the graph of Figure 3(a). Observe that the drawing is a sub-triangulation of the minimum weight triangulation of a regular 15-gon. Dotted lines describe the parts of the triangulation (and of its e-dual) that are not part of the drawing.

This establishes that any maximal outerplanar graph G admits a minimum-weight drawing. We now provide a linear-time algorithm for constructing a minimum weight drawing Γ of G. We use the approach of the previous paragraph to choose a particular feasible tree T', namely, a complete feasible tree T' which contains T as a subtree. Let k be the height of T'. Thus T' is the e-dual of a regular $3 \cdot 2^{k-1}$-gon. Clearly, the construction of the regular $3 \cdot 2^{k-1}$-gon corresponding to T' must be avoided if the algorithm is to be linear time. Observe that the vertices of the regular $3 \cdot 2^{k-1}$-gon are evenly distributed around the circumference of a disk. thus the vertices of Γ are also distributed around the disk. We need only give a method for computing the location of these vertices. Recall that each edge of G corresponds to a non-root vertex of its e-dual T. Now, the length in Γ of an external edge e of G having e-dual vertex x is given by $w_{T'}(x')$, where x' is the vertex of T' corresponding to x. Note that x' is not necessarily a leaf of T', even though x is a leaf of T. Observe also that the length of e is completely determined by $l_{T'}(x')$, and that $l_{T'}(x') = 2^{k-d}$, where d is the depth of x (or x'). Thus for each leaf x of T, the edge corresponding to x is a chord of length $2sin(\frac{\pi}{n}2^{k-d})$ connecting 2 vertices on the disk.

So, to construct Γ, do an inorder traversal of the leaves of T, drawing chords of the corresponding lengths on the disk. This gives a drawing of the outer face of G, which completely determines the location of all vertices; now draw all remaining edges as straight line segments. □

Note that if G happens to have an e-dual which is weight-balanced, then this e-dual can be used to directly produce a minimum weight drawing of G.

5 Conclusions and Open Problems

In this paper we have proved that every maximal outerplanar graph admits a straight-line drawing that is a minimum weight triangulation of the set of points representing the vertices in the drawing. We have also provided a complete

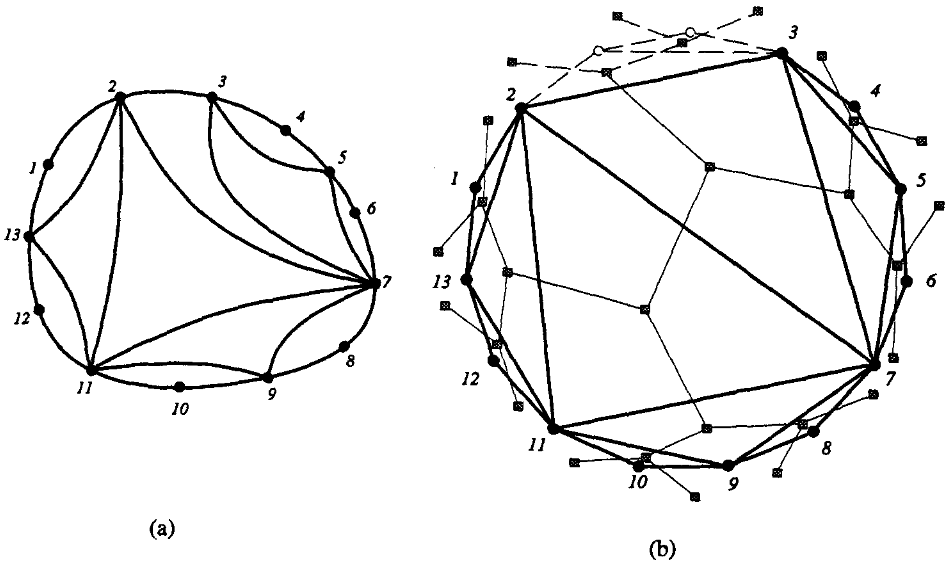

Fig. 3. (a) A maximal outerplanar graph and (b) its minimum weight drawing.

characterization of the minimum weight triangulations of regular polygons. The general problem of determining which triangulations are drawable as minimum weight triangulations is still far from solved. As an intermediate step toward answering the question, we think it might be worth investigating the minimum weight drawability of special classes of graphs, like the 4-connected planar triangulated graphs or the maximal k-outerplanar graphs (a graph is k-outerplanar when it has an embedding such that all vertices are on disjoint cycles properly nested at most k deep).

Acknowledgement

The authors wish to thank Andrzej Lingas for the information he provided on the status of the minimum weight triangulation problem and Roberto Tamassia for his encouragement during the preparation of this paper.

References

1. O. Aichholzer, F. Aurenhammer, G. Rote, M. Taschwer. Triangulations Intersect Nicely. *Proceedings 11th ACM Symposium on Computational Geometry,* Vancouver, Canada, 1995, pp. 220-229.

2. J.A. Bondy and U.S.R. Murty. *Graph Theory with Applications.* Elsevier Science, New York, 1976.
3. G. Di Battista, P. Eades, R. Tamassia and I.G. Tollis. Algorithms for Automatic Graph Drawing: An Annotated Bibliography. *Computational Geometry: Theory and Applications,* **4**, 1994, pp. 235-282.
4. M.T. Dickerson, S.A. McElfresh, M. Montague. New Algorithms and Empirical Findings on Minimum Weight Triangulations Heuristics. *Proceedings 11th ACM Symposium on Computational Geometry,* Vancouver, Canada, 1995, pp. 238-247.
5. P. Eades and S. Whitesides. The Realization Problem for Euclidean Minimum Spanning Tree is NP-hard. *Proceedings 10th ACM Symposium on Computational Geometry,* Stony Brook NY, USA, 1994, pp. 49-56.
6. M.R. Garey and D.S. Johnson. *Computers and Intractability-A Guide to the Theory of NP-Completeness.* Freeman, New York, 1979.
7. P.N. Gilbert. New Results in Planar Triangulations. M.Sc. Thesis, Coordinated Science Laboratory, University of Illinois, Urbana, IL, 1979.
8. L.S. Heath and S.V. Pemmaraju. New Results for the Minimum Weight Triangulation Problem. *Algorithmica,* **12**, 1994 pp. 533-552.
9. J.M. Keil. Computing a Subgraph of the Minimum Weight Triangulation. *Computational Geometry: Theory and Applications,* **4**, 1994, pp. 13-26.
10. D. G. Kirkpatrick. A Note on Delaunay and Optimal Triangulations. *information Processing Letters,* **10**, 1980, pp. 127-128.
11. G.T. Klincsek. Minimal triangulations of Polygonal Domains. *Ann. Discrete Math.,* **9**, 1980, pp. 121-123.
12. C. Levcopoulos. An $\Omega(\sqrt{n})$ Lower Bound for the Nonoptimality of the Greedy Triangulation. *Information Processing Letters,* **25**, 1987, pp. 247-251.
13. C. Levcopoulos and A. Lingas. On Approximation Behavior of the Greedy Triangulation for Convex Polygons. *Algorithmica,* **2**, 1987, pp. 175-193.
14. A. Lingas. A Linear-Time Heuristic for Minimum Weight Triangulation of Convex Polygons. *Proc. 23rd Allerton Conference on Computing, Communication, and Control,* Urbana, 1985, pp. 480 -485.
15. A. Lingas. The Greedy and Delaunay Triangulations are not Bad in the Average Case. *Information Processing Letters,* **22**, 1986, pp. 25-31.
16. A.Lingas. A New Heuristic for Minimum Weight Triangulation. *SIAM J. Algebraic Discrete Methods,* **8**, 1987, pp. 646-658.
17. E.L. Lloyd. On Triangulations of a Set of Points in the Plane. *Proceedings of the 18th Conference on the Foundations of Computer Science,* Providence, RI, 1977, pp. 228-240.
18. K. Manacher, L. Zobrist. Neither the Greedy nor the Delaunay Triangulation of a Planar Point Set Approximates the Optimal Triangulation. *Information Processing Letters,* **9**, 1979, pp. 31-34.
19. H. Meijer and D. Rappaport. Computing the Minimum Weight Triangulation of a Set of Linearly Ordered Points. *Information Processing Letters,* **42**, 1992, pp. 35-38.
20. C. Monma and S. Suri. Transitions in Geometric Minimum Spanning Trees. *Proc. 7th ACM Symposium on Computational Geometry,* 1991, pp. 239-249.
21. F. P. Preparata and M. I. Shamos, *Computational Geometry – an Introduction.* Springer-Verlag, New York, 1985.