ELSEVIER

Information Processing Letters 70 (1999) 17–22

Information Processing Letters

www.elsevier.com/locate/ipl

# Maximum weight triangulation and graph drawing ☆

Cao An Wang [a,*], Francis Y. Chin [b,1], Bo Ting Yang [a]

[a] *Department of Computer Science, Memorial University of Newfoundland, St. John's, Newfoundland, Canada A1B 3X5*
[b] *Department of Computer Science, University of Hong Kong, Hong Kong*

Received 1 July 1998; received in revised form 1 February 1999
Communicated by S.G. Akl

**Abstract**

In this paper, we investigate the maximum weight triangulation of a convex polygon and its application to graph drawing. We can find the maximum weight triangulation of a special $n$-gon which inscribed on a circle in $O(n^2)$ time. The complexity of this algorithm can be reduced to $O(n)$ if the polygon is regular. The algorithm also produces a triangulation approximating the maximum weight triangulation of a convex $n$-gon with weight ratio 0.5. We further show that a tree always admits a maximum weight drawing if the internal nodes of the tree connect to at most 2 non-leaf nodes, and the drawing can be done in $O(n)$ time. Finally, we prove a property of maximum planar graphs which do not admit a maximum weight drawing on any convex point set. 

*Keywords:* Algorithms; Approximation; Graph drawing; Maximum weight triangulation

## 1. Introduction

Triangulation of a set of points is a fundamental structure in computational geometry. Among different triangulations, the *minimum weight triangulation*, MWT, of a set of points in the plane attracts special attention [3,4,7]. The construction of the MWT of a point set is still an outstanding open problem. When the given point set is the set of vertices of a convex polygon (so-called *convex point set*), then the corresponding MWT can be found in $O(n^3)$ time by dynamic programming [3,4].

According to the authors' best knowledge, there is not much research done on *maximum weight triangulation*, MaxWT. From the theoretical viewpoint, the maximum weight triangulation problem and the minimum weight triangulation problem attract equally interest, and one seems not to be easier than the other. The study of maximum weight triangulation will help us to understand the nature of optimal triangulations.

Straight-line drawing is a field of growing interest [1]. A special type of straight-line drawings is minimum weight drawings. Let $C$ be a class of graphs, and $S$ be a set of points in the plane. Let $G = (V, E)$ be a graph of $C$ such that $V(G) = S$, $E$ is a set of non-crossing straight-line segments connecting pairs of points of $S$, and the length sum of all the edges in $E$ is minimized over all straight-line graphs of class $C$ on $S$, $G$ is called a *minimum weight representative of* $C$ with respect to $S$. A graph $G \in C$ is said to admit a *minimum weight drawing* if $G$ is a minimum weight representative of $C$ with respect to some point set $S$. In particular, if $C$ is the class of trees, tree $G$ admits

☆ This work is supported by NSERC grant OPG0041629 and RGC grant HKU 541/96E.

* Corresponding author. Email: wang@garfield.cs.mun.ca.
[1] Email: chin@cs.hku.hk.

0020-0190/99/$ – see front matter 
PII: S0020-0190(99)00037-X

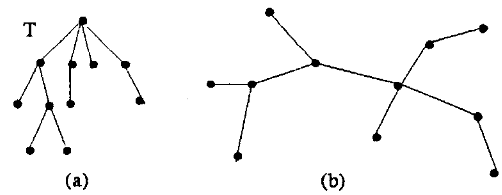


Fig. 1. An illustration of minimum weight drawing.

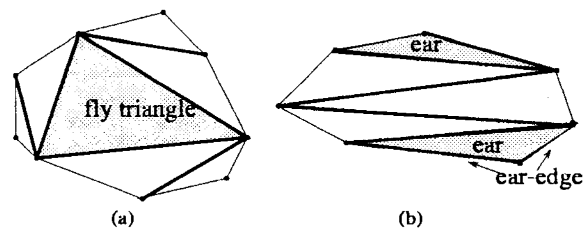


Fig. 2. An illustration of the definitions.

a minimum weight drawing if there exists a set $S$ of points in the plane such that $G$ is isomorphic to an Euclidean minimum weight spanning tree of $S$. For example, tree $T$ in Fig. 1(a) has a minimum weight drawing as $T$ is isomorphic to an Euclidean minimum weight spanning tree as given in Fig. 1(b).

The results of minimum weight drawing of graphs can be found in [1,2,6]. The *maximum weight drawing* of graphs, MaxWD, can be defined similarly.

## 2. Preliminaries

Let $S$ be a set of points in the plane. A *triangulation* of $S$, denoted by $T(S)$, is a maximal set of non-crossing line segments with their endpoints in $S$. It follows that the interior of the convex hull of $S$ is partitioned into non-overlapping triangles. The weight of a triangulation $T(S)$ is given by

$$\omega\big(T(S)\big) = \sum_{\overline{s_i s_j} \in T(S)} \omega(\overline{s_i s_j}),$$

where $\omega(\overline{s_i s_j})$ is the Euclidean length of line segment $\overline{s_i s_j}$.

A *maximum weight triangulation* of $S$ (MaxWT($S$)) is defined, for all possible $T(S)$, as

$$\omega\big(\mathrm{MaxWT}(S)\big) = \max\big\{\omega(T(S))\big\}.$$

Let $P$ be a convex polygon (whose vertices are a *convex point set*) and $T(P)$ be its triangulation. A *fly triangle* of $T(P)$ is one that consists of three diagonals (Fig. 2(a)). An *ear* of $T(P)$ is a triangle containing two consecutive boundary edges of $P$, which are called *ear edges*. An *inner-spanning tree* of the vertices of $P$ is a subgraph of $T(P)$ whose nodes are those vertices of $P$ and whose edges are the internal edges of $T(P)$ plus two ear edges, one per ear (Fig. 2(b)).

For simplicity, in the proofs of the lemmas, we use $\leqslant, =, <, +, -, *$, etc. to denote the comparison and arithmetic operations of the lengths of arcs or line segments, i.e., $\overline{ab} < \overline{cd}$ means $\omega(\overline{ab}) < \omega(\overline{cd})$, and $\widehat{ab} < \widehat{cd}$ means the length of arc $\widehat{ab}$ is less than that of arc $\widehat{cd}$.

## 3. The MaxWT of some convex polygons

The following lemma shows an important property of the maximum weight triangulation of convex polygon.

**Lemma 1.** *If $P$ is a convex polygon, then each interior angle of any fly triangle of the* MaxWT($P$) *must be no less than $\pi/4$.*

**Proof.** By contradiction. Without loss of generality, assume $\triangle abd$ is a fly triangle in the MaxWT($P$) with $\angle a < \pi/4$ as the smallest angle and $\overline{ai}$ is the line segment perpendicular to $\overline{bd}$ from $a$ (Fig. 3). As $\angle a = (\alpha+\beta) < \pi/4$, we have $\overline{bd} = \overline{ai} * (\tan\alpha + \tan\beta) < \overline{ai} * \tan(\alpha+\beta) < \overline{ai}$. Replacing $\overline{bd}$ by $\overline{ac}$ (as $\overline{ac} > \overline{ai} > \overline{bd}$) would arrive at another triangulation whose weight is larger than the weight of MaxWT($P$). This leads to a contradiction. □

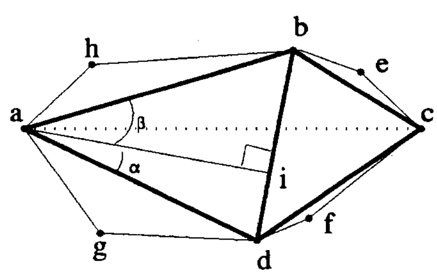


Fig. 3. Fly triangles in the MaxWT($P$).

**Corollary 2.** *If $P$ is a convex polygon, then no interior angle of any fly triangle of* MaxWT($P$) *is larger than $\pi/2$.*

**Lemma 3.** *If $P$ is an inscribed polygon. Then* MaxWT($P$) *cannot contain any fly triangle.*

**Proof.** By contradiction. With respect to Fig. 4, let $\triangle abc$ be a fly triangle of MaxWT($P$). Then, $\triangle abc$ has three neighboring triangles, say $\triangle aeb$, $\triangle bfc$, and $\triangle cda$. Let the intersection points of diagonals $\overline{af}$, $\overline{bd}$, and $\overline{ce}$ be $a'$, $b'$, and $c'$, respectively. There are two distinct cases, depending on whether center $o$ of the circumcircle lies inside $\triangle a'b'c'$ or not.

(1) Let $o$ lie outside the triangle $\triangle a'b'c'$ (Fig. 4(a)). Then, $o$ must lie inside one of the areas bounded by ($\overline{cc'}$, $\overline{c'a}$, $\widehat{adc}$) or ($\overline{aa'}$, $\overline{a'b}$, $\widehat{bea}$) or ($\overline{bb'}$, $\overline{b'c}$, $\widehat{cfb}$). Without loss of generality, let $o$ lie inside the area bounded by $\overline{cc'}$, $\overline{c'a}$, and $\widehat{adc}$. In quadrilateral $\square aebc$ of MaxWT($P$), given $\overline{ce} \leqslant \overline{ab}$, then $\widehat{cfbe} \leqslant \widehat{bea}$. Similarly, in $\square abfc$ of MaxWT($P$), as $\overline{af} \leqslant \overline{bc}$, then $\widehat{fbea} \leqslant \widehat{cfb}$. Thus, we have $\widehat{cfbe} + \widehat{fbea} \leqslant \widehat{bea} + \widehat{cfb}$, or $\widehat{fbe} \leqslant 0$, a contradiction.

(2) Let $o$ lie inside $\triangle a'b'c'$ (Fig. 4(b)). In $\square aebc$ of MaxWT($P$), $\overline{ce} \leqslant \overline{ab}$, then we have $\widehat{cfbe} \leqslant \widehat{bea}$. Similarly, we have $\widehat{adcf} \leqslant \widehat{cfb}$ as $\overline{af} \leqslant \overline{bc}$, and $\widehat{bead} \leqslant \widehat{adc}$ as $\overline{db} \leqslant \overline{ac}$. Adding them accordingly, we have $(\widehat{bead} + \widehat{adcf} + \widehat{cfbe}) \leqslant (\widehat{adc} + \widehat{cfb} + \widehat{bea})$. Then, $(\widehat{ad} + \widehat{cf} + \widehat{be}) \leqslant 0$, a contradiction.

Both of the above cases assume center $o$ lies inside fly triangle $\triangle abc$. If center $o$ lies outside $\triangle abc$, one of the angles of $\triangle abc$ must be larger than $\pi/2$. By Lemma 3, $\triangle abc$ cannot be a fly triangle. □

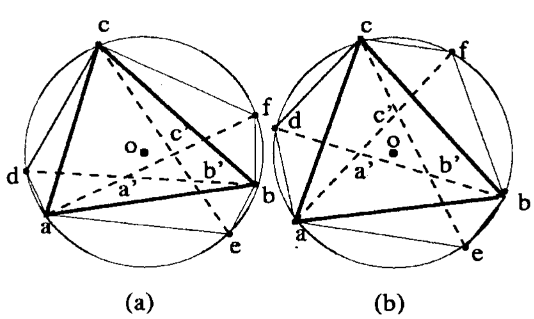


Fig. 4. For the proof of Lemma 3.

**Theorem 4.** *The* MaxWT($P$) *for an inscribed $n$-gon $P$ can be found in* $O(n^2)$ *time.*

**Proof.** Assume $P = (0, 1, \ldots, n-1)$ and all the vertex indices are modulo $n$. Let $W_{i,j}$ with $0 \leqslant i, j \leqslant n-1$ denote the weight of MaxWT($P_{i,j}$), where $P_{i,j} = (i, i+1, \ldots, j)$ is the convex subpolygon of $P$. By Lemma 3, MaxWT($P$) does not contain any fly triangle. Thus, for each internal edge $\overline{ij}$ in MaxWT($P$), the triangle in $P_{i,j}$ associated with edge $\overline{ij}$ must involve with either boundary edge $\overline{i(i+1)}$ and diagonal $\overline{(i+1)j}$ or boundary edge $\overline{(j-1)j}$ and diagonal $\overline{i(j-1)}$.

Thus, we have the following recurrence formula for $W_{i,j}$, where all indices are modulo $n$.

$$W_{i,j} = \begin{cases} 0 \quad \text{if } j = (i+1) \bmod (n); \\ \max\{W_{i,j-1}, W_{i+1,j}\} + \omega(\overline{ij}) \\ \qquad \text{otherwise.} \end{cases}$$

Since the recurrence indices $i$ and $j$ range from 0 to $n-1$ and each evaluation of $W_{i,j}$ takes constant time, all $W_{i,j}$ for $0 \leqslant i, j \leqslant n-1$ can be evaluated in $O(n^2)$ time. Finally,

$$\omega(\text{MaxWT}(P)) = \max\{W_{i,i+1} \mid 0 \leqslant i \leqslant n-1\}$$

which takes another $O(n)$ time. □

Theorem 4 gives an $O(n^2)$ algorithm for finding the maximum weight triangulation of a general convex polygon $P$ such that its internal edges form a tree. We let Ap($P$) be the triangulation produced by this algorithm.

**Corollary 5.** *The maximum weight inner-spanning tree of the vertices of an inscribed polygon $P$ can be found in* $O(n^2)$ *time.*

**Proof.** As the inner-spanning tree of a convex point set cannot contain any fly triangle, the algorithm for finding the maximum weight inner-spanning tree of the vertices of an inscribed polygon $P$ is similar to the algorithm for finding MaxWT($P$). Let $W'_{i,j}$ be the weight of the maximum weight inner-spanning tree of $P_{i,j}$, thus the recurrence formula for $W'_{i,j}$, with all indices are modulo $n$, is

$$W'_{i,j} = \begin{cases} \omega(\overline{ij}) \quad \text{if } j = (i+1) \bmod (n); \\ \max\{W'_{i,j-1}, W'_{i+1,j}\} + \omega(\overline{ij}) \\ \qquad \text{otherwise.} \end{cases}$$

□

When $P$ is a regular polygon, the following theorem shows that any triangulation of $P$ without fly triangles is a maximum weight triangulation.

**Theorem 6.** *Any inner-spanning tree of a regular n-gon $P$ is maximum and together with the boundary edges of $P$ it forms a* MaxWT($P$).

**Proof.** Corollary 2 implies that the MaxWT($P$) does not contain any cycle formed by diagonals. As $P$ is regular and its boundary edges are shorter than its diagonals, all the internal edges of MaxWT($P$) and two edges of $P$ form a maximum weight spanning tree. We say that a diagonal *bridges* $k$ boundary edges if the diagonal and the $k$ boundary edges form a cycle of length $k+1$. For every inner-spanning tree of $P$, it must consist of two boundary edges, a diagonal bridging two boundary edges, a diagonal bridging three boundary edges, ..., and a diagonal bridging $(n-2)$ boundary edges. As $P$ is regular, all diagonals bridging the same number of boundary edges must be of the same length. Thus, all the inner-spanning tree must be maximum and of the same weight. □

## 4. MaxWD of caterpillar graphs

A *caterpillar* is a tree such that all internal nodes connect to at most 2 non-leaf nodes. Fig. 5 gives an example of caterpillar.

Let $C$ be the class of caterpillars. We say caterpillar $G_c$ has a maximum weight drawing if there exists a convex point set $P$ in the plane such that $G_c$ is isomorphic to an Euclidean maximum weight spanning tree of $P$.

In this section, we present a linear-time algorithm for the MaxWD of caterpillars through the inner-spanning trees on the vertex set of a regular polygons. Given a caterpillar of $n$ nodes $G_c$, we construct a *regular point set*, i.e., the vertex set of a regular $n$-gon, $(0, 1, \ldots, n-1)$. The drawing starts from a *head* of the caterpillar, i.e., an internal node with exactly one internal node as its neighbor. For example, nodes $a$ and $k$ are heads in the caterpillar given in Fig. 5(a). The next step is to select a vertex, say $(n-1)$, in the regular $n$-gon to represent the head, and to act as the center of a fan to vertices $0, 1, \ldots$ to represent edges adjacent to the head (Fig. 5(b)). The drawing of the spanning

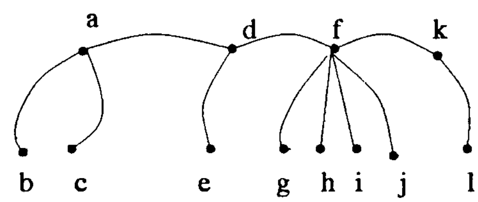


Fig. 5. An illustration of the definition of caterpillar.

tree will continue with the head's neighboring internal node to be represented by the last vertex in the fan (vertex 2 in Fig. 5(b)). The drawing will proceed along the chain of internal nodes of $G_c$ and the detailed algorithm is given below.

**Algorithm** MaxWDRAW
**Input:** Caterpillar graph $G_c$
**Output:** Maximum weight spanning tree isomorphic to $G_c$
**Method:**
1. $n \leftarrow |V(G_c)|$;
   Draw a regular point set $(0, 1, \ldots, n-1)$.
2. Let $V_I$ be the chain of internal nodes starting from a head of $G_c$.
   $s \leftarrow n-1$; $t \leftarrow 0$; Draw line $\overline{st}$
3. **While** $V_I \neq \emptyset$ **do**
   $v_I \leftarrow Extract(V_I)$; $k \leftarrow degree(v_I)$;
   Draw line $\overline{sj}$ for $j = t+1, t+2, \ldots, t+k-1$;
   $t \leftarrow t+k-1$;
   **if** $V_I \neq \emptyset$ **then**
   $v_I \leftarrow Extract(V_I)$; $k \leftarrow degree(v_I)$;
   Draw line $\overline{tj}$ for $j = s-1, s-2, \ldots, s-k+1$;
   $s \leftarrow s-k+1$;
**EndDo**

**Theorem 7.** *Any caterpillar graph has a straight-line maximum weight drawing and which can be drawn in linear time.*

**Proof.** Apply algorithm MaxWDRAW to $G_c$. The output is a spanning tree over a regular point set, which is isomorphic to $G_c$. By Theorem 6, this spanning tree gives the maximum weight triangulation of the regular polygon formed by the point set. The proof of Corollary 5 implies that this spanning tree also is of the maximum weight. Finally, it is easy to see that MaxWDRAW takes $O(n)$ time. □

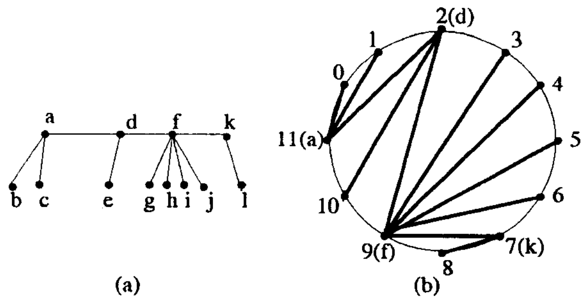


Fig. 6. (a) Caterpillar graph; (b) the corresponding maximum weight drawing.

## 5. Forbidden graphs for MaxWD on convex point set

A graph $G$ is *outerplanar* if it has a planar embedding such that all its nodes lie on a single face; an outerplanar graph is *maximal* if no edge can be added to the planar embedding without crossing. In this section, we shall prove that some maximal outerplanar graphs do not admit an MaxWD, these graphs are called *forbidden graphs*.

**Lemma 8.** *If $P$ is a convex point set, then there cannot exist two fly triangles sharing an edge in the* MaxWT($P$).

**Proof.** By contradiction. Without loss of generality, assume the two fly triangles are $\triangle abd$ and $\triangle bcd$ as shown in Fig. 3. We have

$$\begin{aligned}\overline{ac}^2 &= \overline{ab}^2 + \overline{bc}^2 - 2\overline{ab} * \overline{bc} * \cos(\angle abc)\\ &= \overline{ad}^2 + \overline{dc}^2 - 2\overline{ad} * \overline{dc} * \cos(\angle cda).\end{aligned}$$

$$\begin{aligned}\overline{bd}^2 &= \overline{ab}^2 + \overline{ad}^2 - 2\overline{ab} * \overline{ad} * \cos(\angle dab)\\ &= \overline{bc}^2 + \overline{dc}^2 - 2\overline{bc} * \overline{dc} * \cos(\angle bcd).\end{aligned}$$

From Lemma 1, since all angles of the fly triangles are larger than $\pi/4$, $\angle abc$ and $\angle cda$ are larger than $\pi/2$, i.e., $\cos(\angle abc)$ and $\cos(\angle cda)$ are negative. Thus, we have $2\overline{ac}^2 - 2\overline{bd}^2 > 0$ or $\overline{ac} > \overline{bd}$. This contradicts that $\overline{bd}$ is an edge in MaxWT($P$). □

Let $C$ be the class of all maximal outerplanar graphs. A maximal outerplanar graph $G$ has a maximum weight drawing if there exists a convex point set $P$ in the plane such that $G$ is isomorphic to an Euclidean maximum weight triangulation of $P$.

Based on Lemma 8, the following theorem shows that some maximal outerplanar graphs do not have maximum weight drawings. Fig. 3 illustrates such an example.

**Theorem 9.** *If $G(V, E)$ is a maximal outerplanar graph containing a simple cycle $C$ with four nonconsecutive nodes which form two triangles sharing a common edge, then $G$ cannot have a maximum weight drawing.*

**Proof.** Fig. 3 shows a maximal outerplanar graph which does not have a maximum weight drawing, cycle $C = ahbecfdga$ and the four nonconsecutive nodes $a, b, c, d$, as specified in the theorem. As long as nodes $a, b, c, d$ are nonconsecutive, any ear edges in the triangulation can be replaced by chains of nodes (note that many edges are needed to connect these nodes to make the graph maximal). The proof follows directly from Lemma 8 as any triangulation of a convex polygon isomorphic to a maximal outerplanar graph having the property specified in the theorem would imply the existence of two fly triangles sharing a common edge. □

## 6. Approximating the MaxWT of a convex polygon

It is a well-known open problem of whether or not one can find the MWT($P$) of convex $n$-gon $P$ in $o(n^3)$ time, similarly for the MaxWT($P$) problem.

In this section, we present an $O(n^2)$ time algorithm to approximate the maximum weight triangulation of a convex $n$-gon. The worst ratio of $\omega$(ApT($P$)) and $\omega$(MaxWT($P$)) is at least 0.5, where $\omega$(ApT($P$)) is the weight of the triangulation produced by the algorithm described in the proof of Theorem 4.

Let $\triangle abc$ be a fly triangle of MaxWT($P$). The removal of a fly triangle $\triangle abc$ will divide $P$ into three components, each associates with an edge of $\triangle abc$. $\triangle abc$ is called an *ear-fly triangle* if at most one of its three components contains other fly triangles.

**Lemma 10.** *Ratio*

$$\frac{\omega(T(P))}{\omega(\mathrm{MaxWT}(P))} \geqslant \frac{1}{2}.$$

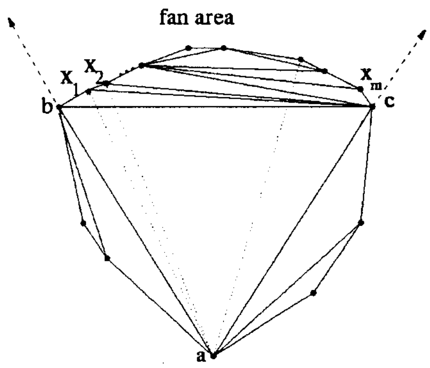


Fig. 7. For the proof of Lemma 10.

**Proof.** (Refer to Fig. 7.) The main idea of the proof is to locate an ear-fly triangle, say $\triangle abc$ and then to make a sequence of edge-flips [5] to transform MaxWT($P$) to $T(P)$. For example, in Fig. 7, these components associated with $\overline{ab}$ and $\overline{ac}$ do not contain any fly triangles, and we flip $\overline{bc}$ with $\overline{ax_1}$, then $\overline{cx_1}$ with $\overline{ax_2}$, etc. until all the nodes $x_1, x_2, \ldots, x_m$ are connected to $a$ directly. Since $\triangle abc$ is a fly triangle, $\angle bac$ ranges from 45° to 90° (Lemma 1 and Corollary 2). Thus,

$$\frac{\omega(\overline{ax_1})}{\omega(\overline{bc})} \geqslant \frac{1}{2}, \frac{\omega(\overline{ax_2})}{\omega(\overline{cx_1})} \geqslant \frac{1}{2}, \ldots,$$

and we have

$$\frac{\omega(T(P))}{\omega(\mathrm{MaxWT}(P))} \geqslant \frac{1}{2}. \qquad \square$$

**Theorem 11.** *There exists an* $O(n^2)$ *time approximation algorithm which guarantees that*

$$\frac{\omega(\mathrm{ApT}(P))}{\omega(\mathrm{MaxWT}(P))} \geqslant \frac{1}{2}.$$

**Proof.** As $T(P)$ is a triangulation which contains no fly triangles and Ap($P$) is the maximum weight triangulation which contains no fly triangles, so $\omega(\mathrm{ApT}(P)) \geqslant \omega(T(P))$. Thus,

$$\frac{\omega(\mathrm{ApT}(P))}{\omega(\mathrm{MaxWT}(P))} \geqslant \frac{\omega(T(P))}{\omega(\mathrm{MaxWT}(P))} \geqslant \frac{1}{2}.$$

Note that Theorem 4 gives an $O(n^2)$ algorithm to find Ap($P$). $\square$

**Remark.** We believe that the worst ratio should be $(4+\sqrt{3})/6$, occurring on a hexagon formed by having an extra node on each side of a regular triangle.

## References


[1] G. Di Battista, P. Eades, R. Tamassia, I. Tollos, Algorithms for automatic graph drawing: A annotated bibliography, Comput. Geom. Theory Appl. 4 (1994) 235–282.

[2] P. Eades, S. Whitesides, The realization problem for Euclidean minimum spanning tree is NP-hard, Algorithmica 16 (1996) 60–82.

[3] P. Gilbert, New results on planar triangulations, Technical Report ACT-15, Coord. Sci. Lab., University of Illinois at Urbana, 1979.

[4] G. Klincsek, Minimal triangulations of polygonal domains, Annual Discrete Math. 9 (1980) 121–123.

[5] C. Levcopoulos, A. Lingas, On approximation behavior of the greedy triangulation for convex polygon, Algorithmica 2 (1987) 175–193.

[6] W. Lenhart, G. Liotta, Drawing outerplanar minimum weight triangulations, Inform. Process. Lett. 57 (1996) 253–260.

[7] F. Preparata, M. Shamos, Computational Geometry, Springer, Berlin, 1985.