Convex Grid Drawings of Plane Graphs with Rectangular Contours

Akira Kamada[1], Kazuyuki Miura[2], and Takao Nishizeki[1]

[1] Graduate School of Information Sciences, Tohoku University,
Sendai 980-8579, Japan
[2] Faculty of Symbiotic Systems Science, Fukushima University,
Fukushima 960-1296, Japan
kamada@nishizeki.ecei.tohoku.ac.jp, miura@sss.fukushima-u.ac.jp,
nishi@ecei.tohoku.ac.jp

Abstract. In a convex drawing of a plane graph, all edges are drawn as straight-line segments without any edge-intersection and all facial cycles are drawn as convex polygons. In a convex grid drawing, all vertices are put on grid points. A plane graph G has a convex drawing if and only if G is internally triconnected, and an internally triconnected plane graph G has a convex grid drawing on an $n \times n$ grid if G is triconnected or the triconnected component decomposition tree $T(G)$ of G has two or three leaves, where n is the number of vertices in G. In this paper, we show that an internally triconnected plane graph G has a convex grid drawing on a $2n \times n^2$ grid if $T(G)$ has exactly four leaves. We also present an algorithm to find such a drawing in linear time. Our convex grid drawing has a rectangular contour, while most of the known algorithms produce grid drawings having triangular contours.

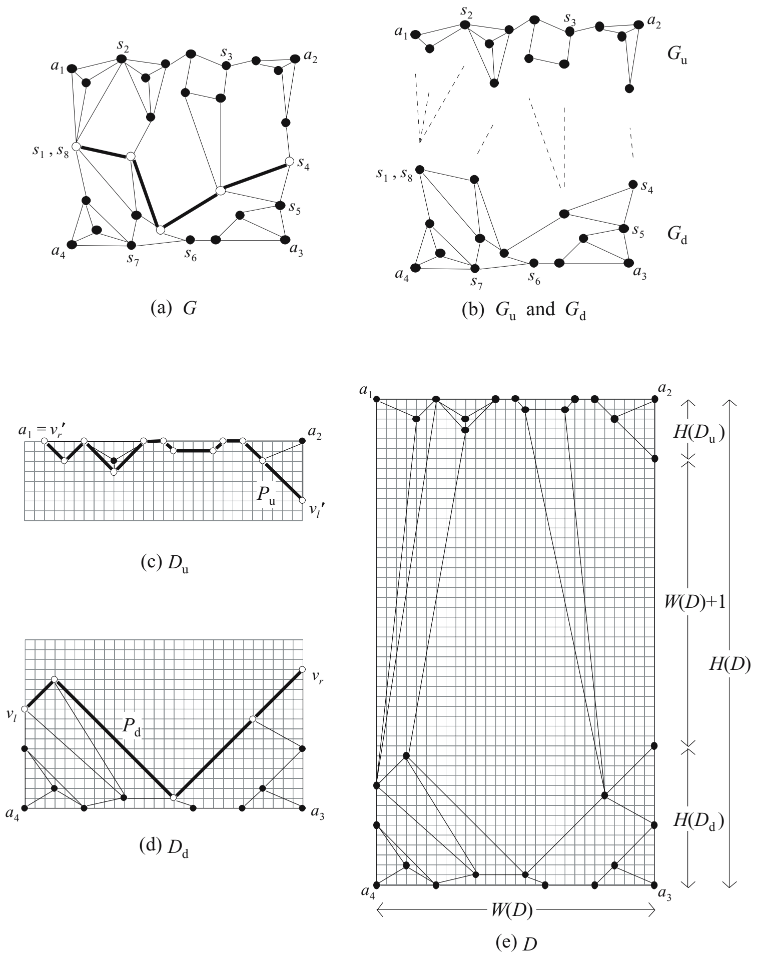
1 Introduction

Recently automatic aesthetic drawing of graphs has created intense interest due to their broad applications, and as a consequence, a number of drawing methods have come out [11]. The most typical drawing of a plane graph is a *straight line drawing* in which all edges are drawn as straight line segments without any edge-intersection. A straight line drawing is called a *convex drawing* if every facial cycle is drawn as a convex polygon. One can find a convex drawing of a plane graph G in linear time if G has one [3,4,11].

A straight line drawing of a plane graph is called a *grid drawing* if all vertices are put on grid points of integer coordinates. This paper deals with a *convex grid drawing* of a plane graph. Throughout the paper we assume for simplicity that every vertex of a plane graph G has degree three or more, because the two edges incident to a vertex of degree two are often drawn on a straight line. Then G has a convex drawing if and only if G is "internally triconnected" [9]. One may thus assume without loss of generality that G is internally triconnected. If either G is triconnected [2] or the "triconnected component decomposition tree" $T(G)$ of G has two or three leaves [8], then G has a convex grid drawing on an $(n-1)\times(n-1)$

T. Asano (Ed.): ISAAC 2006, LNCS 4288, pp. 131–140, 2006.
© Springer-Verlag Berlin Heidelberg 2006

grid and such a drawing can be found in linear time, where n is the number of vertices of G. However, it has not been known whether G has a convex grid drawing of polynomial size if $T(G)$ has four or more leaves. Figure 1(a) depicts an internally triconnected plane graph G, Fig. 2(b) the triconnected components of G, and Fig. 2(c) the triconnected component decomposition tree $T(G)$ of G, which has four leaves l_1, l_2, l_3 and l_4.



(a) G (b) G_u and G_d

(c) D_u

(d) D_d

(e) D

Fig. 1. (a) A plane graph G, (b) subgraphs G_u and G_d, (c) a drawing D_u of G_u, (d) a drawing D_d of G_d, and (e) a convex grid drawing D of G

In this paper, we show that an internally triconnected plane graph G has a convex grid drawing on a $2n \times n^2$ grid if $T(G)$ has exactly four leaves, and present an algorithm to find such a drawing in linear time. The algorithm is outlined as follows: we first divide a plane graph G into an upper subgraph G_u

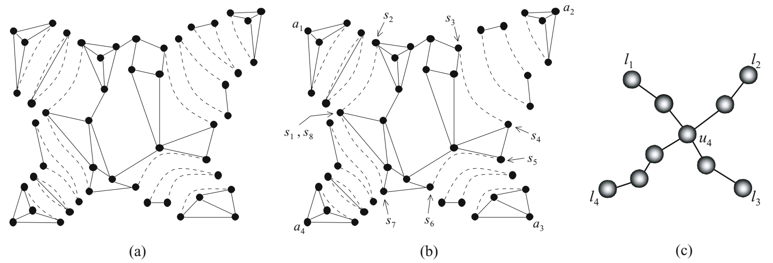


Fig. 2. (a) Split components of the graph G in Fig. 1(a), (b) triconnected components of G, and (c) a decomposition tree $T(G)$

and a lower subgraph G_{d} as illustrated in Fig. 1(b) for the graph in Fig. 1(a); we then construct "inner" convex grid drawings of G_{u} and G_{d} by a so-called shift method as illustrated in Figs. 1(c) and (d); we finally extend these two drawings to a convex grid drawing of G as illustrated in Fig. 1(e). This is the first algorithm that finds a convex grid drawing of such a plane graph G in a grid of polynomial size. Our convex grid drawing has a rectangular contour, while most of the previously known algorithms produce a grid drawing having a triangular contour [1,2,5,6,8,13].

2 Preliminaries

We denote by $W(D)$ the width of a minimum integer grid enclosing a grid drawing D of a graph, and by $H(D)$ the height of D. A plane graph G divides the plane into connected regions, called *faces*. The infinite face is called an *outer face*, and the others are called *inner faces*. The boundary of a face is called a *facial cycle*. We denote by $F_{\mathrm{o}}(G)$ the outer facial cycle of G. A vertex on $F_{\mathrm{o}}(G)$ is called an *outer vertex*, while a vertex not on $F_{\mathrm{o}}(G)$ is called an *inner vertex*. In a convex drawing of a plane graph G, all facial cycles must be drawn as convex polygons. The convex polygonal drawing of $F_{\mathrm{o}}(G)$ is called an *outer polygon*. We call a vertex of a polygon an *apex* in order to avoid the confusion with a vertex of a graph.

We call a pair $\{u, v\}$ of vertices in a biconnected graph G a *separation pair* if its removal from G results in a disconnected graph, that is, $G - \{u, v\}$ is not connected. A biconnected graph G is *triconnected* if G has no separation pair. A biconnected plane graph G is *internally triconnected* if, for any separation pair $\{u, v\}$ of G, both u and v are outer vertices and each connected component of $G - \{u, v\}$ contains an outer vertex. In other words, G is internally triconnected if and only if it can be extended to a triconnected graph by adding a vertex in the outer face and joining it to all outer vertices.

Let $G = (V, E)$ be a biconnected graph, and let $\{u, v\}$ be a separation pair of G. Then, G has two subgraphs $G'_1 = (V_1, E'_1)$ and $G'_2 = (V_2, E'_2)$ satisfying the following two conditions.

(a) $V = V_1 \bigcup V_2$, $V_1 \bigcap V_2 = \{u, v\}$; and
(b) $E = E'_1 \bigcup E'_2$, $E'_1 \bigcap E'_2 = \emptyset$, $|E'_1| \geq 2$, $|E'_2| \geq 2$.

For a separation pair $\{u, v\}$ of G, $G_1 = (V_1, E'_1 + (u, v))$ and $G_2 = (V_2, E'_2 + (u, v))$ are called the *split graphs* of G with respect to $\{u, v\}$. The new edges (u, v) added to G_1 and G_2 are called the *virtual edges.* Even if G has no multiple edges, G_1 and G_2 may have. Dividing a graph G into two split graphs G_1 and G_2 is called *splitting.* Reassembling the two split graphs G_1 and G_2 into G is called *merging.* Merging is the inverse of splitting. Suppose that a graph G is split, the split graphs are split, and so on, until no more splits are possible, as illustrated in Fig. 2(a) for the graph in Fig. 1(a) where virtual edges are drawn by dotted lines. The graphs constructed in this way are called the *split components* of G. The split components are of three types: triconnected graphs; triple bonds (i.e. a set of three multiple edges); and triangles (i.e. a cycle of length three). The *triconnected components* of G are obtained from the split components of G by merging triple bonds into a bond and triangles into a ring, as far as possible, where a *bond* is a set of multiple edges and a *ring* is a cycle. Thus the triconnected components of G are of three types: (a) triconnected graphs; (b) bonds; and (c) rings. Two triangles in Fig. 2(a) are merged into a single ring, and hence the graph in Fig. 1(a) has ten triconnected components as illustrated in Fig. 2(b).

Let $T(G)$ be a tree such that each node corresponds to a triconnected component H_i of G and there is an edge (H_i, H_j), $i \neq j$, in $T(G)$ if and only if H_i and H_j are triconnected components with respect to the same separation pair, as illustrated in Fig. 2(c). We call $T(G)$ a *triconnected component decomposition tree* or simply a *decomposition tree* of G [7]. We denote by $\ell(G)$ the number of leaves of $T(G)$. Then $\ell(G) = 4$ for the graph G in Fig. 1(a). (See Fig. 2(c).) If G is triconnected, then $T(G)$ consists of a single isolated node and hence $\ell(G) = 1$.

The following two lemmas are known.

Lemma 1. *[9] Let G be a biconnected plane graph in which every vertex has degree three or more. Then the following three statements are equivalent to each other:*

(a) G has a convex drawing;
(b) G is internally triconnected; and
(c) both vertices of every separation pair are outer vertices, and a node of the decomposition tree $T(G)$ of G has degree two if it is a bond.

Lemma 2. *[9] If a plane graph G has a convex drawing D, then the number of apices of the outer polygon of D is no less than* $\max\{3, \ell(G)\}$*, and there is a convex drawing of G whose outer polygon has exactly* $\max\{3, \ell(G)\}$ *apices.*

Since G is an internally triconnected simple graph and every vertex of G has degree three or more, by Lemma 1 every leaf of $T(G)$ is a triconnected graph.

Lemmas 1 and 2 imply that if $T(G)$ has exactly four leaves then the outer polygon must have four or more apices. Our algorithm obtains a convex grid drawing of G whose outer polygon has exactly four apices and is a rectangle in particular, as illustrated in Fig. 1(e).

In Section 3, we will present an algorithm to draw the upper subgraph G_{u} and the lower subgraph G_{d}. The algorithm uses the following "canonical decomposition." Let $G = (V, E)$ be an internally triconnected plane graph, and let $V = \{v_1, v_2, \cdots, v_n\}$. Let v_1, v_2 and v_n be three arbitrary outer vertices appearing counterclockwise on $F_{\mathrm{o}}(G)$ in this order. We may assume that v_1 and v_2 are consecutive on $F_{\mathrm{o}}(G)$; otherwise, add a virtual edge (v_1, v_2) to the original graph, and let G be the resulting graph. Let $\Pi = (U_1, U_2, \cdots, U_m)$ be an ordered partition of V into nonempty subsets $U_1, U_2, \cdots, U_m$. We denote by G_k, $1 \leq k \leq m$, the subgraph of G induced by $U_1 \bigcup U_2 \bigcup \cdots \bigcup U_k$, and denote by $\overline{G_k}$, $0 \leq k \leq m-1$, the subgraph of G induced by $U_{k+1} \bigcup U_{k+2} \bigcup \cdots \bigcup U_m$. We say that Π is a *canonical decomposition* of G (with respect to vertices v_1, v_2 and v_n) if the following three conditions (cd1)–(cd3) hold:

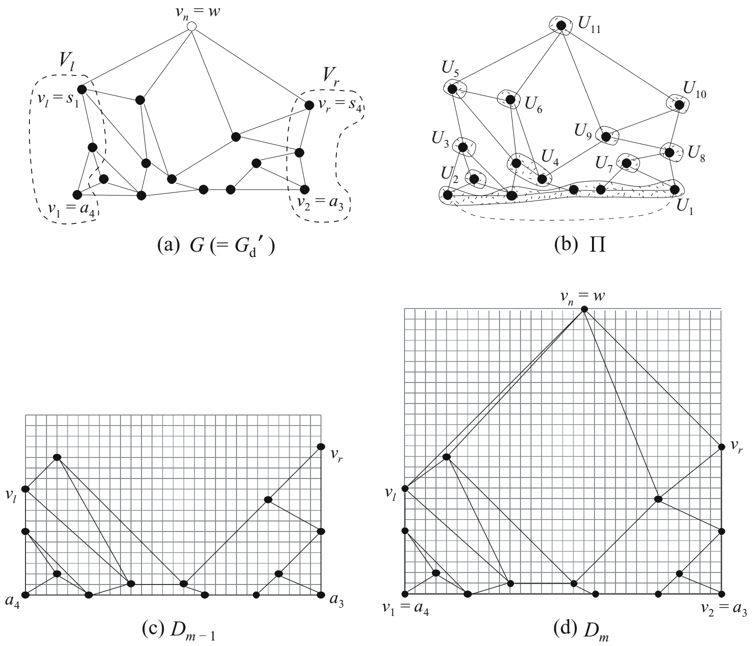
- (cd1) $U_m = \{v_n\}$, and U_1 consists of all the vertices on the inner facial cycle containing edge (v_1, v_2).
- (cd2) For each index k, $1 \leq k \leq m$, G_k is internally triconnected.
- (cd3) For each index k, $2 \leq k \leq m$, all the vertices in U_k are outer vertices of G_k, and
  - (a) if $|U_k| = 1$, then the vertex in U_k has two or more neighbors in G_{k-1} and has one or more neighbors in $\overline{G_k}$ when $k < m$; and
  - (b) if $|U_k| \geq 2$, then each vertex in U_k has exactly two neighbors in G_k, and has one or more neighbors in $\overline{G_k}$.

A canonical decomposition $\Pi = (U_1, U_2, \cdots, U_{11})$ with respect to vertices v_1, v_2 and v_n of the graph in Fig. 3(a) is illustrated in Fig. 3(b).

3 Pentagonal Drawing

Let G be a plane graph having a canonical decomposition $\Pi = (U_1, U_2, \cdots, U_m)$ with respect to vertices v_1, v_2 and v_n, as illustrated in Figs. 3(a) and (b). In this section, we present a linear-time algorithm, called the *pentagonal drawing algorithm*, to find a convex grid drawing of G with a pentagonal outer polygon, as illustrated in Fig. 3(d). The algorithm is based on the so-called shift methods given by Chrobak and Kant [2] and de Fraysseix *et al.* [5], and will be used by our convex grid drawing algorithm in Section 4 to draw G_{u} and G_{d}.

Let v_l be an arbitrary vertex on the path going from v_1 to v_n clockwise on $F_{\mathrm{o}}(G)$, and let $v_r (\neq v_l)$ be an arbitrary vertex on the path going from v_2 to v_n counterclockwise on $F_{\mathrm{o}}(G)$, as illustrated in Fig. 3(a). Let V_l be the set of all vertices on the path going from v_1 to v_l clockwise on $F_{\mathrm{o}}(G)$, and let V_r be the set of all vertices on the path going from v_2 to v_r counterclockwise on $F_{\mathrm{o}}(G)$. Our



(a) $G (= G_d')$ (b) Π

(c) D_{m-1} (d) D_m

Fig. 3. (a) An internally triconnected plane graph $G(= G'_{\rm d})$, (b) a canonical decomposition Π of G, (c) a drawing D_{m-1} of G_{m-1}, and (d) a pentagonal drawing D_m of G

pentagonal drawing algorithm obtains a convex grid drawing of G whose outer polygon is a pentagon with apices v_1, v_2, v_r, v_n and v_l, as illustrated in Fig. 3(d).

We first obtain a drawing D_1 of the subgraph G_1 of G induced by all vertices of U_1. Let $F_{\rm o}(G_1) = w_1, w_2, \cdots, w_t$, $w_1 = v_1$, and $w_t = v_2$. We draw G_1 as illustrated in Fig. 4, depending on whether (v_1, v_2) is a real edge or not, $w_2 \in V_l$ or not, and $w_{t-1} \in V_r$ or not.

We then extend a drawing D_{k-1} of G_{k-1} to a drawing D_k of G_k for each index k, $2 \leq k \leq m$. Let $F_{\rm o}(G_{k-1}) = w_1, w_2, \cdots, w_t$, $w_1 = v_1$, $w_t = v_2$, and $U_k = \{u_1, u_2, \cdots, u_r\}$. Let w_f be the vertex with the maximum index f among all the vertices w_i, $1 \leq i \leq t$, on $F_{\rm o}(G_{k-1})$ that are contained in V_l. Let w_g be the vertex with the minimum index g among all the vertices w_i that are contained in V_r. Of course, $1 \leq f < g \leq t$. We denote by $\angle w_i$ the interior angle of apex w_i of the outer polygon of D_{k-1}. We call w_i a *convex apex* of the

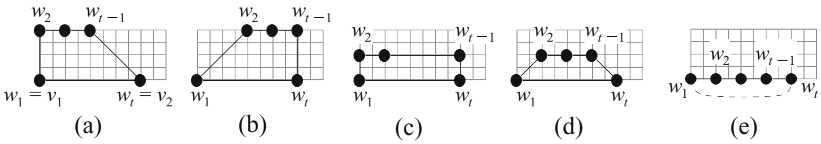


Fig. 4. Drawings D_1 of G_1

polygon if $\angle w_i < \pi$. Assume that a drawing D_{k-1} of G_{k-1} satisfies the following six conditions (sh1)–(sh6). Indeed D_1 satisfies them.

(sh1) w_1 is on the grid point $(0, 0)$, and w_t is on the grid point $(2|V(G_{k-1})| - 2, 0)$.
(sh2) $x(w_1) = x(w_2) = \cdots = x(w_f)$, $x(w_f) < x(w_{f+1}) < \cdots < x(w_g)$, $x(w_g) = x(w_{g+1}) = \cdots = x(w_t)$, where $x(w_i)$ is the x-coordinate of w_i.
(sh3) Every edge (w_i, w_{i+1}), $f \le i \le g - 1$, has slope $-1, 0$, or 1.
(sh4) The Manhattan distance between any two grid points w_i and w_j, $f \le i < j \le g$, is an even number.
(sh5) Every inner face of G_{k-1} is drawn as a convex polygon.
(sh6) Vertex w_i, $f + 1 \le i \le g - 1$, has one or more neighbors in $\overline{G_{k-1}}$ if w_i is a convex apex.

We extend D_{k-1} to D_k, $2 \le k \le m$, so that D_k satisfies conditions (sh1)–(sh6). Let w_p be the leftmost neighbor of u_1, that is, w_p is the neighbor of u_1 in G_k having the smallest index p, and let w_q be the rightmost neighbor of u_r. Before installing U_k to D_{k-1}, we first shift $w_1, w_2, \cdots, w_p$ of G_{k-1} and some inner vertices of G_k to the left by $|U_k|$, and then shift $w_q, w_{q+1}, \cdots, w_t$ of G_{k-1} and some inner vertices of G_k to the right by $|U_k|$. After the operation, we shift all vertices of G_{k-1} to the right by $|U_k|$ so that w_1 is on the grid point $(0, 0)$.

Clearly $W(D_1) = 2|V(G_1)| - 2$ and $H(D_1) \le 4$. One can observe that $W(D_k) = 2|V(G_k)| - 2$ and $H(D_k) \le H(D_{k-1}) + W(D_k)$ for each k, $2 \le k \le m$. We thus have the following lemma.

Lemma 3. *For a plane graph G having a canonical decomposition $\Pi = (U_1, U_2, \cdots, U_m)$ with respect to v_1, v_2 and v_n, the pentagonal drawing algorithm obtains a convex grid drawing of G on a $W \times H$ grid with $W = 2n - 2$ and $H \le n^2 - n - 2$ in linear time. Furthermore, $W(D_{m-1}) = 2(|V(G_{m-1})|) - 2$ and $H(D_{m-1}) \le |V(G_{m-1})|^2 - |V(G_{m-1})| - 2$.*

4 Convex Grid Drawing Algorithm

In this section we present a linear algorithm to find a convex grid drawing D of an internally triconnected plane graph G whose decomposition tree $T(G)$ has exactly four leaves. Such a graph G does not have a canonical decomposition, and hence none of the algorithms in [1], [2], [6], [8] and Section 3 can find a convex grid drawing of G.

Division. We first explain how to divide G into G_{u} and G_{d}. (See Figs. 1(a) and (b).) One may assume that the four leaves l_1, l_2, l_3 and l_4 of $T(G)$ appear clockwise in $T(G)$ in this order. Clearly, either exactly one node u_4 of $T(G)$ has degree four and each of the other non-leaf nodes has degree two as illustrated in Fig. 2(c), or two nodes have degree three and each of the other non-leaf nodes has degree two. In this extended abstract, we consider only the former case. Since each vertex of G is assumed to have degree three or more, all the four leaves of $T(G)$ are triconnected graphs. Moreover, according to Lemma 1, every bond has degree two in $T(G)$. Therefore, node u_4 is either a triconnected graph or a ring. We assume in this extended abstract that u_4 is a triconnected graph as in Fig. 2.

As the four apices of the rectangular contour of G, we choose four outer vertices a_i, $1 \leq i \leq 4$, of G; let a_i be an arbitrary outer vertex in the component l_i that is not a vertex of the separation pair of the component. The four vertices a_1, a_2, a_3 and a_4 appear clockwise on $F_{\mathrm{o}}(G)$ in this order as illustrated in Fig. 1(a).

We then choose eight vertices $s_1, s_2, \cdots, s_8$ from the outer vertices of the component u_4. Among these outer vertices, let s_1 be the vertex that one encounters first when one traverses $F_{\mathrm{o}}(G)$ counterclockwise from the vertex a_1, and let s_2 be the vertex that one encounters first when one traverses $F_{\mathrm{o}}(G)$ clockwise from a_1, as illustrated in Fig. 1(a). Similarly, we choose s_3 and s_4 for a_2, s_5 and s_6 for a_3, and s_7 and s_8 for a_4.

We then show how to divide G into G_{u} and G_{d}. Split G for separation pairs $\{s_1, s_2\}$ and $\{s_3, s_4\}$ as far as possible, and let G' be the resulting split graph containing vertices a_3 and a_4. Then, G' is internally triconnected, and $T(G')$ has exactly two leaves. Consider all the inner faces of G' that contain one or more vertices on the path going from s_2 to s_3 clockwise on $F_{\mathrm{o}}(G')$. Let G'' be the subgraph of G' induced by the vertices on these faces. Then $F_{\mathrm{o}}(G'')$ is a simple cycle. Clearly, $F_{\mathrm{o}}(G'')$ contains vertices s_1 and s_4. Let P be the path going from s_1 to s_4 counterclockwise on $F_{\mathrm{o}}(G'')$. (P is drawn by thick lines in Fig. 1(a).)

Let G_{d} be the subgraph of G induced by all the vertices on or below P, and let G_{u} be the subgraph of G obtained by deleting all vertices in G_{d} as illustrated in Fig. 1(b). Let n_{d} be the number of vertices of G_{d}, and let n_{u} be the number of vertices of G_{u}. Then $n_{\mathrm{d}} + n_{\mathrm{u}} = n$.

Drawing G_{d}. We now explain how to draw G_{d}. Let G'_{d} be a graph obtained from G by contracting all the vertices of G_{u} to a single vertex w, as illustrated in Fig. 3(a) for the graph G in Fig. 1(a)D One can prove that the plane graph G'_{d} is internally triconnected.

The decomposition tree $T(G'_{\mathrm{d}})$ of G'_{d} has exactly two leaves, and a_3 and a_4 are contained in the triconnected graphs corresponding to the leaves and are not vertices of the separation pairs. Every vertex of G'_{d} other than w has degree three or more, and w has degree two or more in G'_{d}. Therefore, G'_{d} has a canonical decomposition $\Pi = (U_1, U_2, \cdots, U_m)$ with respect to a_4, a_3 and w, as illustrated in Fig. 3(b), where $U_m = \{w\}$. Let v_l be the vertex preceding w clockwise on the outer face $F_{\mathrm{o}}(G'_{\mathrm{d}})$, and let v_r be the vertex succeeding w, as illustrated in Fig. 3(a). We obtain a pentagonal drawing D_m of G'_{d} by the algorithm in Section 3, as illustrated in Fig. 3(d). The drawing D_{m-1} of G_{m-1} induced by $U_1 \bigcup U_2 \bigcup \cdots \bigcup U_{m-1}$ is our drawing D_{d} of $G_{\mathrm{d}}(= G_{m-1})$. (See Figs. 1(d) and 3(c).) By Lemma 3, we have $W(D_{\mathrm{d}}) = 2n_{\mathrm{d}} - 2$ and $H(D_{\mathrm{d}}) \leq n_{\mathrm{d}}^2 - n_{\mathrm{d}} - 2$.

Drawing G_{u}. We now explain how to draw G_{u}. Let G'_{u} be a graph obtained from G by contracting all the vertices of G_{d} to a single vertex w'. Similarly to

G'_{d}, G'_{u} has a canonical decomposition $\Pi = (U_1, U_2, \cdots, U_m)$ with respect to a_2, a_1 and w'. Let v'_r be the vertex succeeding w' clockwise on the outer face $F_{\mathrm{o}}(G'_{\mathrm{u}})$, and let v'_l be the vertex preceding w'. We then obtain a drawing D_{m-1} of $G_{\mathrm{u}}(= G_{m-1})$ by the algorithm in Section 3, as illustrated in Fig. 1(c). By Lemma 3, we have $W(D_{\mathrm{u}}) = 2n_{\mathrm{u}} - 2$ and $H(D_{\mathrm{u}}) \leq n_{\mathrm{u}}^2 - n_{\mathrm{u}} - 2$.

Drawing of G. If $W(D_{\mathrm{d}}) \neq W(D_{\mathrm{u}})$, then we widen the narrow one of D_{d} and D_{u} by the shift method in Section 3. We may thus assume that $W(D_{\mathrm{d}}) = W(D_{\mathrm{u}}) = \max\{2n_{\mathrm{d}} - 2, 2n_{\mathrm{u}} - 2\}$. Since we combine the two drawings D_{d} and D_{u} of the same width to a drawing D of G, we have

$$W(D) = \max\{2n_{\mathrm{d}} - 2, 2n_{\mathrm{u}} - 2\} < 2n.$$

We arrange D_{d} and D_{u} so that $y(a_3) = y(a_4) = 0$ and $y(a_1) = y(a_2) = H(D_{\mathrm{d}}) + H(D_{\mathrm{u}}) + W(D) + 1$, as illustrated in Fig. 1(e).

Noting that $n_{\mathrm{d}} + n_{\mathrm{u}} = n$ and $n_{\mathrm{d}}, n_{\mathrm{u}} \geq 5$, we have

$$\begin{aligned} H(D) &= H(D_{\mathrm{d}}) + H(D_{\mathrm{u}}) + W(D) + 1 \\ &< (n_{\mathrm{d}}^2 - n_{\mathrm{d}} - 2) + (n_{\mathrm{u}}^2 - n_{\mathrm{u}} - 2) + 2n + 1 \\ &< n^2. \end{aligned}$$

We finally draw, by straight line segments, all the edges of G that are contained in neither G_{u} nor G_{d}. This completes the grid drawing D of G. (see Fig. 1(e).)

Validity of drawing algorithm. In this section, we show that the drawing D obtained above is a convex grid drawing of G. Since both D_{d} and D_{u} satisfy condition (sh5), every inner facial cycle of G_{d} and G_{u} is drawn as a convex polygon in D. Therefore, it suffices to show that the straight line drawings of the edges not contained in G_{u} and G_{d} do not introduce any edge-intersection and that all the faces newly created by these edges are convex polygons.

Since D_{d} satisfies condition (sh3), the absolute value of the slope of every edge on the path P_{d} going from v_l to v_r clockwise on $F_{\mathrm{o}}(G_{\mathrm{d}})$ is at most 1. The path P_{d} is drawn by thick lines in Fig. 1(d). Similarly, the absolute value of the slope of every edge on the path P_{u} going from v'_r to v'_l counterclockwise on $F_{\mathrm{o}}(G_{\mathrm{u}})$ is at most 1. Since $H(D) = H(D_{\mathrm{d}}) + H(D_{\mathrm{u}}) + W(D) + 1$, the absolute value of the slope of every straight line segment that connects a vertex in G_{u} and a vertex in G_{d} is larger than 1. Therefore, all the outer vertices of G_{d} on P_{d} are visible from all the outer vertices of G_{u} on P_{u}. Furthermore, G is a plane graph. Thus the addition of all the edges not contained in G_{u} and G_{d} does not introduce any edge-intersection.

Since D_{d} satisfies condition (sh6), every convex apex of the outer polygon of G_{d} on P_{d} has one or more neighbors in G_{u}. Similarly, every convex apex of the outer polygon of G_{u} on P_{u} has one or more neighbors in G_{d}. Therefore, every interior angle of a newly formed face is smaller than 180°. Thus all the inner faces of G not contained in G_{u} and G_{d} are convex polygons in D.

Thus, D is a convex grid drawing of G. Clearly the algorithm takes linear time. We thus have the following main theorem.

Theorem 1. *Assume that G is an internally triconnected plane graph, every vertex of G has degree three or more, and the triconnected component decomposition tree $T(G)$ has exactly four leaves. Then our algorithm finds a convex grid drawing of G on a $2n \times n^2$ grid in linear time.*

We finally remark that the grid size is improved to $2n \times 4n$ for the case where either the node u_4 of degree four in $T(G)$ is a ring or $T(G)$ has two nodes of degree three.

References

1. N. Bonichon, S. Felsner and M. Mosbah, *Convex drawings of 3-connected plane graphs -Extended Abstract-*, Proc. of GD 2004, LNCS 3383, pp. 60–70, 2005.
2. M. Chrobak and G. Kant, *Convex grid drawings of 3-connected planar graphs*, International Journal of Computational Geometry and Applications, 7, pp. 211–223, 1997.
3. N. Chiba, K. Onoguchi and T. Nishizeki, *Drawing planar graphs nicely*, Acta Inform., 22, pp. 187–201, 1985.
4. N. Chiba, T. Yamanouchi and T. Nishizeki, *Linear algorithms for convex drawings of planar graphs*, in Progress in Graph Theory, J. A. Bondy and U. S. R. Murty (Eds.), Academic Press, pp. 153–173, 1984.
5. H. de Fraysseix, J. Pach and R. Pollack, *How to draw a planar graph on a grid*, Combinatorica, 10, pp. 41–51, 1990.
6. S. Felsner, *Convex drawings of plane graphs and the order of dimension of 3-polytopes*, Order, 18, pp. 19–37, 2001.
7. J. E. Hopcroft and R. E. Tarjan, *Dividing a graph into triconnected components*, SIAM J. Compt.2, 3, pp. 135–138, 1973.
8. K. Miura, M. Azuma and T. Nishizeki, *Canonical decomposition, realizer, Schnyder labeling and orderly spanning trees of plane graphs*, International Journal of Foundations of Computer Science, 16, 1, pp. 117–141, 2005.
9. K. Miura, M. Azuma and T. Nishizeki, *Convex drawings of plane graphs of minimum outer apices*, Proc. of GD 2005, LNCS 3843, pp. 297–308, 2005, also to appear in International Journal of Foundations of Computer Science.
10. K. Miura, S. Nakano and T. Nishizeki, *Convex grid drawings of four-connected plane graphs*, Proc. of ISAAC 2000, LNCS 1969, pp. 254–265, 2000, also to appear in International Journal of Foundations of Computer Science.
11. T. Nishizeki and M. S. Rahman, *Planar Graph Drawing*, World Scientific, Singapore, 2004.
12. S. Nakano, M. S. Rahman and T. Nishizeki, *A linear time algorithm for four partitioning four-connected planar graphs*, Information Processing Letters, 62, pp. 315–322, 1997.
13. W. Schnyder and W. Trotter, *Convex drawings of planar graphs*, Abstracts of the AMS 13, 5, 92T-05-135, 1992.