

# An Application of Well-Orderly Trees in Graph Drawing

Huaming Zhang and Xin He\*

Department of Computer Science and Engineering  
SUNY at Buffalo  
Buffalo, NY, 14260, USA

**Abstract.** *Well-orderly trees* seem to have the potential of becoming a powerful technique capable of deriving new results in graph encoding, graph enumeration and graph generation [3, 4]. Our application of well-orderly trees in this paper provides new evidence to their power. We give more compact visibility representation of plane graphs using the properties of well orderly trees.

## 1 Introduction

Graph drawing has emerged as an exciting and fast growing area of research in the computer science community in recent years [1]. Among various techniques for drawing planar graphs, the *canonical orderings and canonical ordering trees* of 3-connected plane graphs have served as a fundamental step upon which drawing algorithms are built [7–9, 12]. The work by de Fraysseix, Pach and Pollack [9] is considered to be the first using the canonical orderings to produce straight-line drawings with polynomial sizes for planar graphs. The technique of canonical orderings has subsequently been applied to drawing graphs with respect to a variety of aesthetic constraints, including straight-line, convexity, orthogonality, visibility representation, 2-visibility, floor-planning, and others.

Later on, Chiang et. al. introduced the concept of *orderly spanning tree* [6], which generalizes canonical ordering tree and leads to several improvements in various styles of graph drawings [6, 5, 16]. In [3], Bonichon, Gavaille and Hanusse introduced *well-orderly trees*, which are canonical ordering trees with some special properties. These special properties have been successfully used in graph encoding, graph enumeration, and graph generation [3, 4]. More importantly, well-orderly trees are closely related to the concept of *Schnyder's realizers* [20, 21], which has also been widely used in graph drawing. We believe, well-orderly trees will be a promising technique of unifying known results as well as deriving new results in various styles in graph drawings. In this paper, we are going to derive an application of well-orderly trees in graph drawing.

A *visibility representation* (VR for short) of a plane graph  $G$  is a representation, where the vertices of  $G$  are represented by non-overlapping horizontal line

---

\* Research supported in part by NSF Grant CCR-0309953.

segments (called *vertex segment*), and each edge of  $G$  is represented by a vertical line segment touching the vertex segments of its end vertices. The problem of computing a compact VR is important not only in algorithmic graph theory, but also in practical applications such as VLSI layout. A simple linear time VR algorithm was given in [19,22] for a 2-connected plane graph  $G$ . It only uses an *st-orientation* of  $G$  and the corresponding *st-orientation* of its dual  $G^*$  to construct a VR of  $G$ .

One of the main concerns afterwards for VR is the size of the representation, i.e., the height and width of VR. Some work has been done to reduce the size of the VR by carefully choosing a special *st-orientation* of  $G$ . We summarize related previous results in the following table:

References	Plane graph $G$	4-Connected plane graph $G$
[19, 22]	Width of VR $\leq (2n - 5)$	Height of VR $\leq (n - 1)$
[13]	Width of VR $\leq \lfloor \frac{3n-6}{2} \rfloor$	
[17]	Width of VR $\leq \lfloor \frac{22n-42}{15} \rfloor$	
[14]		Width of VR $\leq (n - 1)$
[25]	Height of VR $\leq \lfloor \frac{5n}{6} \rfloor$	
[24, 26]	Width of VR $\leq \lfloor \frac{13n-24}{9} \rfloor$	Height of VR $\leq \lceil \frac{3n}{4} \rceil$

In this paper, we prove that every plane graph  $G$  has a VR with height at most  $\frac{4n-1}{5}$ , and it can be obtained in linear time.

The present paper is organized as follows. Section 2 introduces preliminaries. Section 3 presents the construction of a VR with height bounded by  $\frac{4n-1}{5}$ .

## 2 Preliminaries

In this section, we give definitions and preliminary results. Definitions not mentioned here are standard.

$G$  is called a *directed graph* (digraph for short) if each edge of  $G$  is assigned a direction. We abbreviate the words “counterclockwise” and “clockwise” as **ccw** and **cw** respectively.

An *orientation* of a graph  $G$  is a digraph obtained from  $G$  by assigning a direction to each edge of  $G$ . We will use  $G$  to denote both the resulting digraph and the underlying undirected graph unless otherwise specified. (Its meaning will be clear from the context.) For a 2-connected plane graph  $G$  and an exterior edge  $(s, t)$ , an orientation of  $G$  is called an *st-orientation* if the resulting digraph is acyclic with  $s$  as the only source and  $t$  as the only sink. For more information on *st-orientation*, we refer readers to [18].

Let  $G$  be a 2-connected plane graph and  $(s, t)$  an exterior edge. An *st-numbering* of  $G$  is a one-to-one mapping  $\xi : V \rightarrow \{1, 2, \dots, n\}$ , such that  $\xi(s) = 1$ ,  $\xi(t) = n$ , and each vertex  $v \neq s, t$  has two neighbors  $u, w$  with  $\xi(u) < \xi(v) < \xi(w)$ , where  $u$  ( $w$ , resp.) is called a *smaller neighbor* (*bigger*

neighbor, resp.) of  $v$ . Given an  $st$ -numbering  $\xi$  of  $G$ , we can orient  $G$  by directing each edge in  $E$  from its lower numbered end vertex to its higher numbered end vertex. The resulting orientation is called the *orientation derived from  $\xi$*  which, obviously, is an  $st$ -orientation of  $G$ . On the other hand, if  $G = (V, E)$  has an  $st$ -orientation  $\mathcal{O}$ , we can define an 1-1 mapping  $\xi : V \rightarrow \{1, \dots, n\}$  by topological sort. It is easy to see that  $\xi$  is an  $st$ -numbering and the orientation derived from  $\xi$  is  $\mathcal{O}$ . From now on, we will interchangeably use the term an  $st$ -numbering of  $G$  and the term an  $st$ -orientation of  $G$ , where each edge of  $G$  is directed accordingly.

Lempel et. al. [15] showed that for every 2-connected plane graph  $G$  and an exterior edge  $(s, t)$ , there exists an  $st$ -numbering. The following lemma was given in [19, 22]:

**Lemma 1.** *Let  $G$  be a 2-connected plane graph. Let  $\mathcal{O}$  be an  $st$ -orientation of  $G$ . A VR of  $G$  can be obtained from  $\mathcal{O}$  in linear time. The height of the VR is the length of the longest directed path in  $\mathcal{O}$ .*

Let  $T$  be a rooted spanning tree of a plane graph  $G$ . Two nodes are unrelated if neither of them is an ancestor of the other in  $T$ . An edge of  $G$  is unrelated if its endpoints are unrelated.

Bonichon et. al. introduced *well-orderly trees* [3], a special case of *orderly spanning trees* defined by Chiang, Lin and Lu in [6], referred as simply *orderly trees* afterwards. Let  $v_1, v_2, \dots, v_n$  be the ccw preordering of the nodes in  $T$ . A node  $v_i$  is *orderly* in  $T$  with respect to  $T$  if the incident edges of  $v_i$  in  $T$  form the following four blocks (possibly empty) in ccw order around  $v_i$ :

- $B_p(v_i)$ : the edge incident to the parent of  $v_i$ ;
- $B_<(v_i)$ : unrelated edges incident to nodes  $v_j$  with  $j < i$ ;
- $B_C(v_i)$ : edges incident to the children of  $v_i$ ; and
- $B_>(v_i)$ : unrelated edges incident to nodes  $v_j$  with  $j > i$ .

A node  $v_i$  is *well-orderly* in  $G$  with respect to  $T$  if it is orderly, and if:

- the first ccw edge  $(v_i, v_j) \in B_>(v_i)$ , if it exists, verifies that the parent of  $v_j$  is an ancestor of  $v_i$ .

$T$  is a *well-orderly tree* of  $G$  if all the nodes of  $T$  are well-orderly in  $G$ , and if the root of  $T$  belongs to the boundary of the exterior face of  $G$  (similarly for simply orderly tree). Note that an orderly tree (simply orderly or well-orderly) is necessarily a spanning tree.

A plane triangulation is a plane graph where every face is a triangle (including the exterior face). Let  $G$  be a plane triangulation of  $n$  vertices with three exterior vertices  $v_1, v_2, v_n$  in ccw order. A *realizer*  $\mathcal{R} = \{T_1, T_2, T_n\}$  of  $G$  is a partition of its interior edges into three sets  $T_1, T_2, T_n$  of directed edges such that the following holds:

- for each  $i \in \{1, 2, n\}$ , the interior edges incident to  $v_i$  are in  $T_i$  and directed toward  $v_i$ .

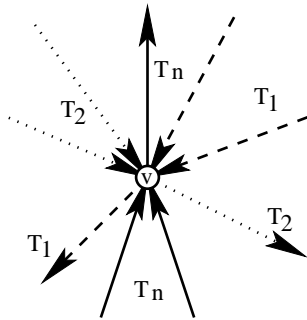


Fig. 1. Edge directions around an interior vertex  $v$ .

- For each interior vertex of  $G$ ,  $v$  has exactly one edge leaving  $v$  in each of  $T_1, T_2, T_n$ . The ccw order of the edges incident to  $v$  is: leaving in  $T_1$ , entering in  $T_n$ , leaving in  $T_2$ , entering in  $T_1$ , and entering in  $T_2$  (See Fig. 1). Each entering block could be empty.

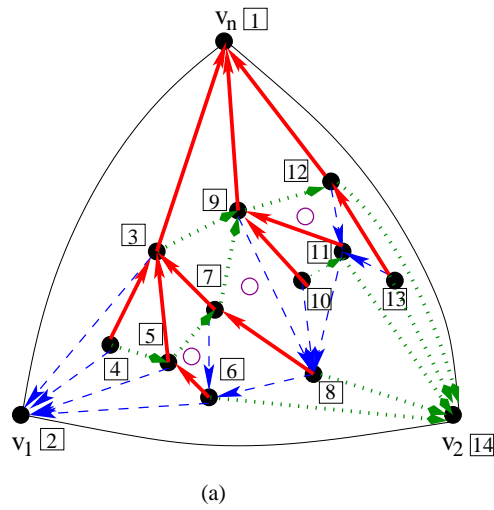


Fig. 2. A plane triangulation  $G$  and the minimum realizer  $\mathcal{R}_0$  of  $G$ .

Normally, realizers of a plane triangulation  $G$  are not unique. Among all the realizer of  $G$ , there is a unique realizer  $\mathcal{R}_0$  of  $G$ , where according to the edge directions in  $\mathcal{R}_0$ , there are no ccw-triangles. This realizer of  $G$  will be called the minimum realizer of  $G$ . For example, in Fig. 2, the three trees of the realizer are

drawn in solid lines, dashed lines and dotted lines respectively. There are three cw cyclic faces (marked by empty circles) but no ccw cyclic triangles, so it is the minimum realizer of  $G$ .

Schnyder showed in [20] that each set  $T_i$  of a realizer is a tree rooted at the exterior vertex  $v_i$ . For each tree  $T_i$  of a realizer, we denote by  $\bar{T}_i$  the tree composed of  $T_i$  augmented with the two edges of the exterior face incident to the root of  $T_i$ , i.e. the vertex  $v_i$ . For example, in Fig. 2,  $\bar{T}_i$  is  $T_n$  (the tree in thick solid lines) augmented with edges  $(v_n, v_1)$  and  $(v_n, v_2)$ .

We summarize related results in the following lemma [3, 6, 20, 21]:

**Lemma 2.** *Let  $G$  be a plane triangulation of  $n$  vertices with three exterior vertices  $v_1, v_2, v_n$  in ccw order. Let  $\mathcal{R} = \{T_1, T_2, T_n\}$  be any realizer of  $G$ . Then,*

1. *Each  $\bar{T}_i$ ,  $i \in \{1, 2, n\}$  is a simply orderly tree. In addition, if  $\mathcal{R}$  is the minimum realizer  $\mathcal{R}_0$ , then each  $\bar{T}_i$ ,  $i \in \{1, 2, n\}$  is a well-orderly tree.*
2. *Given the tree  $\bar{T}_1$  ( $\bar{T}_2$ ,  $\bar{T}_n$  resp.), all the first ccw edge  $(u, v_j) \in B_{>}(u)$  for each node  $u$  with respect to  $\bar{T}_1$  ( $\bar{T}_2$ ,  $\bar{T}_n$  resp.) form the tree  $\bar{T}_n$ . ( $\bar{T}_1$ ,  $\bar{T}_2$  resp.)*
3. *The minimum realizer can be computed in linear time.*

For example, in Fig. 2,  $\bar{T}_n$  is a well-orderly tree for  $G$ . And the first ccw edge  $(9, 12)$  in  $B_{>}(9)$  for the node 9 is in  $\bar{T}_2$ .

Let  $v_1, v_2, \dots, v_n$  be the ccw preordering of the nodes of a tree  $T$ . The subsequence  $v_i, \dots, v_j$  is a *branch* of  $T$  if it is a chain (i.e.,  $v_t$  is the parent of  $v_{t+1}$  for every  $i \leq t < j$ ), and if  $j - i$  is maximal. Branches partition the nodes of  $T$ , and each branch contains exactly one leaf.

Bonichon et. al. proved the following [3]: The well-orderly tree  $\bar{T}_n$  of a minimum realizer  $\mathcal{R}_0 = \{T_1, T_2, T_n\}$  has the *branch property*: All nodes of a given branch of  $\bar{T}_n$  must have the same parent in  $\bar{T}_1$  (except the root of  $\bar{T}_n$ ). (Similar results hold for  $\bar{T}_1$  and  $\bar{T}_2$ .) For example, in Fig. 2, nodes 3, 4 form a branch, they have the same parent in  $\bar{T}_1$ .

### 3 More compact VR of Plane Graphs

Let  $T$  be a tree drawn in the plane. Let  $t_1, t_2, \dots, t_n$  be the cw postordering of the nodes of  $T$ . A node of  $T$  is a *glue node* of  $T$  if it is right before a leaf node in the ordering  $t_1, t_2, \dots, t_n$ . For example, considering  $\bar{T}_n$  in Fig. 2, nodes 14, 12, 11, 9, 7, 5, 3 are the glue nodes. Note that, the set of the first node of all branches of  $T$  except the root is the set of glue nodes. Also observe that the number of glue nodes of  $T$  is the number of leaves of  $T$  minus 1.

Next, let's explore another property of a well-orderly tree of a plane triangulation.

**Lemma 3.** *Let  $\mathcal{R}_0 = \{T_1, T_2, T_n\}$  be the minimum realizer of a plane triangulation  $G$  with  $n$  vertices. Let  $\xi_1, \xi_2, \xi_n$  be the number of internal nodes (i.e., non-leaf node) of  $\bar{T}_1, \bar{T}_2, \bar{T}_n$ ,  $l_1, l_2, l_n$  be the number of the leaves of  $\bar{T}_1, \bar{T}_2, \bar{T}_n$  respectively. Then,*

1. The internal nodes of  $\bar{T}_2$  ( $\bar{T}_n, \bar{T}_1$  resp.) must be the glue nodes of  $\bar{T}_n$  ( $\bar{T}_1, \bar{T}_2$  resp.).
2.  $l_n - 1 \geq \xi_2$ ,  $l_1 - 1 \geq \xi_n$ ,  $l_2 - 1 \geq \xi_1$ .

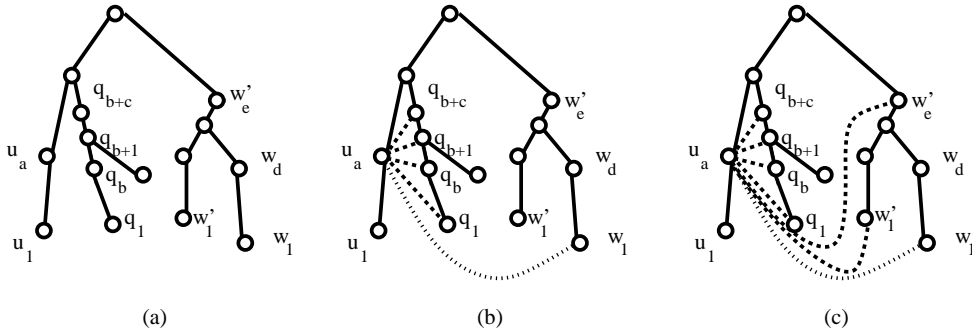
*Proof.* According to Lemma 2, each  $\bar{T}_i$  is a well-orderly tree of  $G$ . We only prove the case of  $\bar{T}_2$ . The other two cases are similar.

1. Let  $w$  be an internal node in  $\bar{T}_2$ . Therefore, there is an edge  $(u, w)$  in  $\bar{T}_2$  such that  $w$  is the parent of  $u$  in  $\bar{T}_2$ . Applying Lemma 2 2, for the node  $u$  in  $\bar{T}_n$ ,  $(u, w)$  is the first ccw edge in  $B_{>}(u)$  with respect to  $\bar{T}_n$ . Since  $\bar{T}_n$  is a well-orderly tree, the parent of  $w$  must be the ancestor of  $u$  in  $\bar{T}_n$ . So  $w$  must be a glue node of  $\bar{T}_n$ .
2. Applying to the observation that the number of glue nodes of  $T$  is the number of leaves of  $T$  minus 1, we have  $l_n - 1 \geq \xi_2$ .

For example, in Fig. 2, the internal nodes of  $\bar{T}_2$  are 14, 12, 9, 7, 5, 11. All of them are glue nodes of  $\bar{T}_n$ .

Next we use the three well-orderly trees from the minimum realizer to obtain more compact VR of a plane triangulation  $G$ .

Let  $\mathcal{R}_0 = \{T_1, T_2, T_n\}$  be the minimum realizer of a plane triangulation  $G$  with  $n$  vertices.



**Fig. 3.** (a) There is no edge between  $u_a$  and  $w_1$ . (b) There is an edge between  $u_a$  and  $w_1$ , and there is no edge between  $u_a$  and  $w'_e$ . (Therefore, there is no edge between  $u_1$  and  $w'_e$ .)  $(u_a, w_1)$  could be in  $\bar{T}_1$  or  $\bar{T}_2$ . Then there must be edges between  $q_1, \dots, q_b, q_{b+1}, \dots, q_{b+c}$  with  $u_a$  and they must be in  $\bar{T}_1$ . (c) There is an edge between  $u_a$  and  $w_1$ , and there is an edge between  $u_a$  and  $w'_e$ . Therefore, there is no edge between  $q_b$  and  $w_1$ .

Let's construct an  $st$ -numbering of  $G$  using  $\bar{T}_n$  step by step. (The cases of using  $\bar{T}_1, \bar{T}_2$  are similar.)

Each step begins from a leaf of  $\bar{T}_n$ . Suppose the leftmost unassigned leaf is  $u_1$ , the second leftmost unassigned leaf is  $q_1$ . The rightmost unassigned leaf is  $w_1$ , the second rightmost unassigned leaf is  $w'_1$ . The ordering of vertices of  $G$  by **ccw post-ordering**, starting from  $u_1$  with respect to  $\bar{T}_n$  is  $u_1, u_2, \dots, u_a, q_1, \dots, q_b$ . And

$q_b$  is the last vertex before the third leaf in this **ccw postordering**. The branch of  $\bar{T}_n$  containing  $q_1$  contains  $q_{b+c}, \dots, q_{b+1}, q_b, \dots, q_1$  (which will be needed later). The ordering of vertices of  $G$  by **ccw postordering**, starting from  $w_1$  with respect to  $\bar{T}_n$  is  $w_1, w_2, \dots, w_d, w'_1, \dots, w'_e$ . And  $w'_e$  is the last vertex before the third leaf in this **ccw postordering** (Namely, the parent of  $w'_e$  in  $\bar{T}_n$  has a child on the left of  $w'_e$ ).

See Fig. 3 for an illustration. Only part of the graph is drawn. Edges and paths of  $T_n$  are drawn in solid lines. Note that  $q_{b+1}$  must have a child on the right of  $q_b$ .

Each step is classified into one of the following three cases:

**Case 1:** If there is no edge between  $u_a$  and  $w_1$ , then we first assign numbers to  $u_1, u_2, \dots, u_a$  by **ccw postordering** with respect to  $\bar{T}_n$ , then continue to assign numbers to  $w_1, \dots, w_d$  by **ccw postordering** with respect to  $\bar{T}_n$ .

**Case 2:**  $(u_a, w_1)$  is an edge in  $G$ , and there is no edge between  $u_a$  and  $w'_e$ . Note that  $q_1$  is a leaf in  $\bar{T}_n$ , and  $u_a$  is the only vertex of  $G$  in  $B_{<}(q_1)$ . Therefore  $(q_1, u_a)$  must be an edge of  $G$  and it is in  $T_1$ . According to the branch property for  $\bar{T}_n$ , all the edges  $(u_a, q_i)$ ,  $i = 1, \dots, b, (b+1), \dots, (b+c)$  must also be in  $\bar{T}_1$ . For the vertex  $q_{b+c}$ ,  $u_a$  is the only vertex of  $G$  in  $B_{<}(q_{b+c})$ , and  $(q_{b+c}, u_a)$  is in  $\bar{T}_1$ . Hence,  $q_{b+c}$  cannot be an internal node in the tree  $\bar{T}_2$ . Also observe that  $q_{b+c}$  is a glue node of  $\bar{T}_n$ . In this case, we first assign numbers to  $w_1, \dots, w_d, w'_1, \dots, w'_e$  by **ccw postordering** with respect to  $\bar{T}_n$ . Then we assign numbers to  $u_1, u_2, \dots, u_a$  by **ccw postordering** with respect to  $\bar{T}_n$ .

**Case 3:**  $(u_a, w_1)$  is an edge in  $G$ , and  $(u_a, w'_e)$  is an edge in  $G$ . Similar to Case 2, all the edges  $(u_a, q_i)$ ,  $i = 1, \dots, b, (b+1), \dots, (b+c)$  must be in  $\bar{T}_1$ . For the vertex  $q_{b+c}$ ,  $u_a$  is the only vertex of  $G$  in  $B_{<}(q_{b+c})$ , and  $(q_{b+c}, u_a)$  is in  $\bar{T}_1$ . Hence,  $q_{b+c}$  cannot be an internal node in the tree  $\bar{T}_2$ . Also observe that  $q_{b+c}$  is a glue node of  $\bar{T}_n$ . Note that, all the edges  $(u_a, w'_i)$ ,  $i = 1, \dots, e$  are also in  $\bar{T}_1$ . Obviously, in this case, there is no edge between  $q_b$  and  $w_1$ . Therefore, we first assign numbers to  $u_1, \dots, u_a$  by **ccw postordering** with respect to  $\bar{T}_n$ , then continue to  $q_1, \dots, q_b$  by **ccw postordering** with respect to  $\bar{T}_n$ . Then we assign numbers to  $w_1, \dots, w_d$  by **ccw postordering** with respect to  $\bar{T}_n$ .

Continue to next step if there are leaves left unassigned.

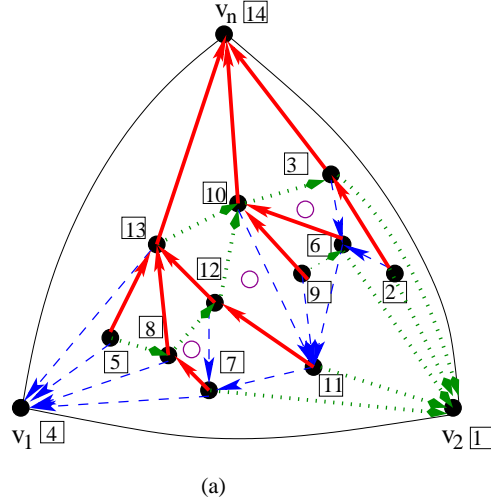
Note: If there are only 1 or 2 leaves left in the end, then we assign the remaining numbers to them either using **ccw postordering** or using **cw postordering** until we finish at the root of  $T$ . We do not count this as a step. Note that, for each node, either it is assigned a number in a **ccw postordering** setting, or it is assigned a number in a **ccw postordering** setting.

We have the following two key observations:

**Observation 1:** For each step, at most three leaves are assigned numbers.

**Observation 2:** If Case 2 and Case 3 are applied  $k_n$  times altogether, then  $k_n$  glue nodes (the nodes  $q_{b+c}$ ) of  $\bar{T}_n$  cannot be internal nodes of  $\bar{T}_2$ . Therefore, according to Lemma 3 (1),  $l_n - k_n - 1 \geq \xi_2$ .

**Lemma 4.** *Let  $G$  be a plane triangulation,  $\mathcal{R}_0 = \{T_1, T_2, T_n\}$  be the minimum realizer of  $G$ . Then, using  $\bar{T}_i$ ,  $i = 1, 2, n$ ,*



**Fig. 4.** An  $st$ -numbering of  $G$  in Fig. 2, obtained from  $\bar{T}_n$  by using our numbering scheme.

1. The numbering of the vertices of  $G$  constructed by the above numbering scheme is an  $st$ -numbering of  $G$ .
2. If Case 2 and Case 3 are applied  $k_i$  times altogether for  $\bar{T}_i$ , then any directed path in the resulting  $st$ -orientation is at most  $n - \frac{l_i - k_i}{2}$ .
3. Any directed path in the resulting  $st$ -orientation is at most  $n - \frac{l_i}{3}$ ,  $i = 1, 2, n$ .

*Proof.* We only prove the case  $i = n$ . The other two cases are similar.

1. First observe that, for any node other than the root of  $\bar{T}_n$ , its parent is assigned a bigger number. And the root of  $\bar{T}_n$  is assigned  $n$ . For any internal node of  $\bar{T}_n$ , their children are assigned smaller numbers. For a leaf  $u \neq v_1, v_2$  of  $\bar{T}_n$ , either it is assigned a number in a **ccw** setting, then the non-empty block  $B_{<}(u)$  contains its smaller neighbors; or it is assigned a number in a **cw** setting, then the non-empty block  $B_{>}(u)$  contains its smaller neighbors. For  $v_1, v_2$ , one of them is assigned 1, and it becomes a smaller neighbor for the other. Therefore, this numbering is an  $st$ -numbering for  $G$ .
2. Observe that, if Case 1 is applied to a step, then one of  $u_a$  and  $w_1$  has to be bypassed by any directed path, and they are assigned consecutive numbers by our numbering scheme. If Case 2 is applied to a step, then one of  $w'_e$  and  $u_1$  must be bypassed by any directed path, and they are assigned consecutive numbers by our numbering scheme (This is because, if  $u_a = u_1$ , then there is no edge between  $w'_e$  and  $u_a$  according to our condition in Case 2. If  $u_a \neq u_1$ , then it is not possible to have an edge  $(w'_e, u_1)$  because  $G$  is a plane graph). If Case 3 is applied to a step, then one of  $q_b$  and  $w_1$  must be bypassed by any directed path, and they are assigned consecutive numbers by our numbering



scheme. Therefore, from the nodes assigned numbers within the same step, at least one node has to be bypassed by any directed path.

Suppose Case 2 and Case 3 are applied  $k_n$  times altogether, then the total number of steps is at least  $\frac{l_n-3k_n}{2} - 1 + k_n$ . (The subtraction of 1 comes from the last 1 or 2 leaves which do not form a step.) Therefore, any directed path has to bypass at least  $\frac{l_n-3k_n}{2} - 1 + k_n$  vertices. Therefore, its length is at most  $n - (\frac{l_n-3k_n}{2} - 1 + k_n) - 1 = n - \frac{l_n-k_n}{2}$ .

3. In the worst scenario, each step assigns numbers to three leaves, then we have  $\lfloor \frac{l_n}{3} \rfloor$  steps. So any directed path must bypass at least  $\lfloor \frac{l_n}{3} \rfloor$  vertices, so its length is at most  $n - \lfloor \frac{l_n}{3} \rfloor - 1 \leq n - \frac{l_n}{3}$ .

For example, Fig. 4 shows an  $st$ -numbering of  $G$ , using our numbering scheme to  $\bar{T}_n$ . The first step numbers 1, 2, 3 by **cw postordering**, then it numbers 4 by **ccw postordering**. The second step numbers 5 by **ccw postordering**, then it numbers 6 by **cw postordering**.

Next we present our main theorem:

**Theorem 1.** *Let  $G$  be a plane triangulation with  $n$  vertices, then there is a VR of  $G$  whose height is at most  $\frac{4n-1}{5}$ . And it can be constructed in linear time.*

*Proof.* Let  $\mathcal{R}_0 = \{T_1, T_2, T_n\}$  be the minimum realizer of  $G$ . Apply our  $st$ -numbering scheme, suppose for  $\bar{T}_1, \bar{T}_2, \bar{T}_n$ , the number of their Case 2 and Case 3 steps altogether are  $k_1, k_2, k_n$  respectively. Then we have  $\xi_2 \leq l_n - k_n - 1$ . Symmetrically, we have  $\xi_1 \leq l_2 - k_2 - 1$ , and  $\xi_n \leq l_1 - k_1 - 1$ . Summing them up and moving 3 to the left side, we have:

$$\xi_1 + \xi_2 + \xi_n + 3 \leq (l_1 + l_2 + l_n) - (k_1 + k_2 + k_n). \quad (1)$$

Pick a longest directed path for each  $st$ -orientation. By Lemma 4 (2), the sum of their lengths is at most:

$$\begin{aligned} & (n - \frac{l_n - k_n}{2}) + (n - \frac{l_2 - k_2}{2}) + (n - \frac{l_1 - k_1}{2}) \\ &= 3n - \frac{l_1 + l_2 + l_n}{2} + \frac{k_1 + k_2 + k_n}{2} \\ &= 3n - \frac{(l_1 + l_2 + l_n) - (k_1 + k_2 + k_n)}{2} \\ &\leq 3n - \frac{\xi_1 + \xi_2 + \xi_n + 3}{2} \end{aligned} \quad (2)$$

The last inequality comes from Equation (1).

By Lemma 4 (3), the sum of their length is at most:

$$\begin{aligned} & n - \frac{l_1}{3} + n - \frac{l_2}{3} + n - \frac{l_n}{3} \\ &= 3n - \frac{l_1 + l_2 + l_n}{3} \end{aligned} \quad (3)$$

Multiply Equation (2) by 2 and multiply Equation (3) by 3. Adding them up, we have that 5 times the sum of the lengths of the three longest directed paths is at most:

$$\begin{aligned}
 & 6n - (\xi_1 + \xi_2 + \xi_n + 3) + 9n - (l_1 + l_2 + l_n) \\
 &= 15n - (\xi_1 + \xi_2 + \xi_n + l_1 + l_2 + l_n) - 3 \\
 &= 15n - 3n - 3 \\
 &= 12n - 3.
 \end{aligned} \tag{4}$$

Therefore, one of the longest directed path from these three paths must be at most  $\frac{12n-3}{15} \leq \frac{4n-1}{5}$ . Applying Lemma 1,  $G$  admits a VR whose height is at most  $\frac{4n-1}{5}$ , and it can be constructed in linear time.

## References

1. G. di Battista, P. Eades, R. Tamassia, and I. Tollis, Graph Drawing: Algorithms for the Visualization of Graphs, Princeton Hall, 1998
2. N. Bonichon, B. Le Saëc and M. Mosbah, Wagner's theorem on realizers, in: Proc. ICALP'02, Lecture Notes in Computer Science, Vol. 2380, (Springer, Berlin, 2002) 1043-1053.
3. N. Bonichon, C. Gavoille, and N. Hanusse, An information-theoretic upper bound of planar graphs using triangulation, in *Proc. STACS'03*, pp 499-510, Lectures Notes in Computer Science, Vol. 2607, Springer-Verlag, 2003.
4. N. Bonichon, C. Gavoille, and N. Hanusse, Canonical decomposition of outerplanar maps and application to enumeration, coding and generation, In *Proc. WG'2003*, pp. 81-92, Lecture Notes in Computer Science, Vol. 2880, Springer-Verlag, 2003
5. H.-L. Chen, C.-C. Liao, H.-I. Lu and H.-C. Yen, Some applications of orderly spanning trees in graph drawing, in *Proc. Graph Drawing'02*, pp. 332-343, Lecture Notes in Computer Science, Vol. 2528, Springer-Verlag, Berlin, 2002.
6. Y.-T. Chiang, C.-C. Lin and H.-I. Lu, Orderly spanning trees with applications to graph encoding and graph drawing, in *Proc. of the 12th Annual ACM-SIAM SODA*, pp. 506-515, ACM Press, New York, 2001.
7. M. Chrobak and G. Kant, Convex grid drawings of 3-connected planar graphs, Technical Report RUU-CS-93-45, Department of Computer Science, Utrecht University, Holland, 1993.
8. U. Fößmeier, G. Kant and M. Kaufmann, 2-Visibility drawings of planar graphs, in *Proc. 4th International Symposium on Graph Drawing*, pp. 155-168, Lecture Notes in Computer Science, Vol. 1190, Springer-Verlag, Berlin, 1996.
9. H. de Fraysseix, J. Pach and R. Pollack, How to draw a planar graph on a grid. *Combinatorica* 10 (1990), 41-51.
10. G. Kant, Drawing planar graphs using the lmc-ordering, in *Proc. 33rd Symposium on Foundations of Computer Science*, pp.101-110, IEEE, Pittsburgh, 1992.
11. G. Kant, Algorithms for drawing planar graphs, Ph.D. Dissertation, Department of Computer Science, University of Utrecht, Holland, 1993.
12. G. Kant, Drawing planar graphs using the canonical ordering, *Algorithmica* 16 (1996), 4-32.
13. G. Kant, A more compact visibility representation. *International Journal of Computational Geometry and Applications* 7 (1997), 197-210.

14. G. Kant and X. He, Regular edge labeling of 4-connected plane graphs and its applications in graph drawing problems. *Theoretical Computer Science* 172 (1997), 175-193.
15. A. Lempel, S. Even and I. Cederbaum, An algorithm for planarity testing of graphs, in *Theory of Graphs (Proc. of an International Symposium, Rome, July 1966)*, pp. 215-232, Rome, 1967.
16. C.-C. Liao, H.-I. Lu and H.-C. Yen, Floor-planning via orderly spanning trees, in *Proc. 9th International Symposium on Graph Drawing*, pp. 367-377, Lecture Notes in Computer Science, Vol. 2265, Springer-Verlag, Berlin, 2002.
17. C.-C. Lin, H.-I. Lu and I-F. Sun, Improved compact visibility representation of planar graph via Schnyder's realizer, in: *SIAM Journal on Discrete Mathematics*, 18 (2004), 19-29.
18. P. Ossona de Mendez, Orientations bipolaires. PhD thesis, Ecole des Hautes Etudes en Sciences Sociales, Paris, 1994.
19. P. Rosenstiehl and R. E. Tarjan, Rectilinear planar layouts and bipolar orientations of planar graphs. *Discrete Comput. Geom.* 1 (1986), 343-353.
20. W. Schnyder, Planar graphs and poset dimension. *Order* 5 (1989), 323-343.
21. W. Schnyder, Embedding planar graphs on the grid, in *Proc. of the First Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 138-148, SIAM, Philadelphia, 1990.
22. R. Tamassia and I.G.Tollis, An unified approach to visibility representations of planar graphs. *Discrete Comput. Geom.* 1 (1986), 321-341.
23. H. Zhang and X. He, Compact Visibility Representation and Straight-Line Grid Embedding of Plane Graphs, in: Proc. WADS'03, Lecture Notes in Computer Science, Vol. 2748, (Springer-Verlag Heidelberg, 2003) 493-504.
24. H. Zhang and X. He, On Visibility Representation of Plane Graphs, in: Proc. STACS'04, Lecture Notes in Computer Science, Vol. 2996, (Springer-Verlag Heidelberg, 2004) 477-488.
25. H. Zhang and X. He, *New Theoretical Bounds of Visibility Representation of Plane Graphs*, in: Proc. GD'2004, Lecture Notes in Computer Science, Vol. 3383, (Springer-Verlag, 2005) pp. 425-430.
26. H. Zhang and X. He, Improved Visibility Representation of Plane Graphs, *Discrete Comput. Geom.* 30 (2005), 29-29.