Canonical Orders and Schnyder Realizers

STEPHEN KOBOUROV

Department of Computer Science, University of Arizona, Tucson, AZ, USA

Years and Authors of Summarized Original Work

1990; de Fraysseix, Pach, and Pollack
1990; Schnyder

Keywords

planar graph drawing algorithms, canonical order, Schnyder realizer

Problem Definition

Every planar graph has a crossings-free drawing in the plane. Formally, a *straight-line drawing* of a planar graph G is one where each vertex is placed at a point in the plane and each edge is represented by a straight-line segment between the two corresponding points such that no two edges cross each other, except possibly at their common endpoints. A *straight-line grid drawing* of G is a straight-line drawing of G where each vertex of G is placed on an integer grid point. The area for such a drawing is defined by the minimum-area axis-aligned rectangle, or *bounding box*, that contains the drawing.

Wagner in 1936 [12], Fáry in 1948 [5], and Stein in 1951 [10] proved independently that every planar graph has a straight-line drawing. It was not until 1990 that the first algorithms for drawing a planar graph on a grid of polynomial area were developed. The concepts of canonical orders [4] and Schnyder realizers [9] were independently introduced for the purpose of efficiently computing straight-line grid drawings on the $O(n) \times O(n)$ grid. These two seemingly very different combinatorial structures turn out to be closely related and have since been used in many different problems and in many applications.

Key Results

We first describe canonical orders for planar graphs and a linear time procedure to construct them. Then we describe Schnyder realizers and a linear time procedure to compute them. Finally, we show how they can be used to compute straight-line grid drawings for planar graph.

Canonical Order

A planar graph G along with a planar embedding (a cyclic order of the neighbors for each vertex), is called a *plane graph.* Given a graph G, testing for planarity and computing a planar embedding can be done in linear time [6]. Let G be a maximal plane graph with outer vertices u, v, w in counter-clockwise order. Then a *canonical order* or *shelling order* of G is a total order of the vertices $v_1 = u$, $v_2 = v$, v_3, ..., $v_n = w$ that meets the following criteria for every $4 \leq i \leq n$.

(a) The subgraph $G_{i-1} \subseteq G$ induced by v_1, v_2, ..., v_{i-1} is 2-connected, and the boundary of its outerface is a cycle C_{i-1} containing the edge (v_1, v_2).
(b) The vertex v_i is in the outerface of G_{i-1}, and its neighbors in G_{i-1} form a subinterval of the path $C_{i-1} - (u, v)$ with at least two vertices; see Fig. 1(a)–(b).
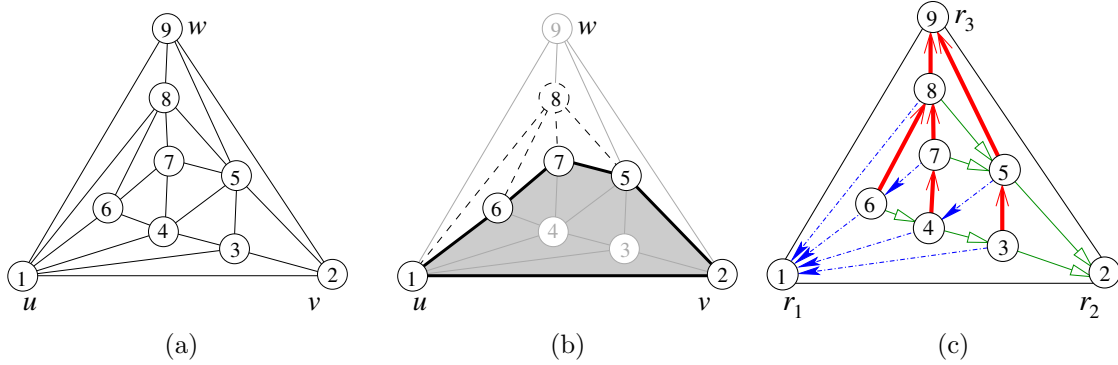
Fig. 1. (a) A canonical order of vertices for a maximal plane graph G, (b) insertion of v_8 in G_7, (c) a Schnyder realizer for G.

Every maximal plane graph G admits a canonical order; see Fig. 1(a)–(b). Moreover, computing such an order can be done in $O(n)$ time where n is the number of vertices in G. Before before proving these claims, we need a simple lemma.

Lemma 1. *Let G be a maximal plane graph with canonical order v_1, $v_2 = v$, v_3, ..., v_n. Then for $i \in \{3, \ldots, n-1\}$, any separating pair $\{x, y\}$ of G_i is a chord of C_i.*

The proof of the lemma is simple. Recall that a *chord* of a cycle $\mathcal{C}$ is an edge between non-adjacent vertices of $\mathcal{C}$. Since G is a maximal plane graph and since each vertex v_j, $j \in \{n, \ldots, i+1\}$ is in the outerface of G_j, all the internal faces of G_i are triangles. Adding a dummy vertex d along with edges from d to each vertex on the outerface of G_i yields a maximal plane graph G'. Then G' is 3-connected and for each separation pair $\{x, y\}$ of G, the set $T = \{x, y, d\}$ is a separating set of G' since $G = G' \setminus d$. The set T is a separating triangle in G' [1] and therefore the edge (x, y) is a chord on C_i.

Theorem 1. *A canonical order of a maximal plane graph G can be computed in linear time.*

This is also easy to prove. If the number of vertices n in G is 3, then the canonical ordering of G is trivially defined. Let $n > 3$ and choose the vertices $v_n = w$, v_{n-1}, …, v_3 in this order so that conditions (a)–(b) of the definition are satisfied. Since G is a maximal plane graph, G is 3-connected and hence $G_{n-1} = G \setminus w$ is 2-connected. Furthermore the set of vertices adjacent to $v_n = w$ forms a cycle C_{n-1}, which is the boundary of the outerface of G_{n-1}. Thus conditions (a)–(b) hold for $k = n$.

Assume by induction hypothesis that the vertices $v_n, v_{n-1}, \ldots, v_{i+1}$, $i \geq 3$ have been appropriately chosen. We now find the next vertex v_i. If we can find a vertex x on C_i, which is not an end-vertex of a chord, then we can choose $v_k = x$. Indeed if deleting x from G_i violated the 2-connectivity, then the cut-vertex y of $G_i - x$, together with x would form a separating pair for G_i and hence (x, y) would be a chord in G_i (from the lemma above). We now show that we can find a vertex v_i on C_i which is not an end-vertex of a chord.

If there is no chord of C_i, then we can choose any vertex of C_i other than u and v as v_k. Otherwise label the vertices of $C_i - \{(u, v)\}$ by $p_1 = u, p_2, \ldots, p_t = v$ consecutively from u to v. By definition, any chord (p_k, p_l), $k < l$ must have $k < l - 1$. We say that a chord (p_k, p_l), $k < l$, includes another chord $(p_{k'}, p_{l'})$, $k' < l'$, if $k \leq k' < l' \leq l$. Then take an inclusion-minimal chord (p_k, p_l) and any vertex p_j for $k < j < l$ can be chosen as v_i. Since v_i is not an end-vertex of a chord for C_{i-1}, $G_{i-1} = G_i \setminus v_i$ remains 2-connected. Furthermore due to the maximal planarity of G, the neighborhood of v_i on C_{i-1} forms a subinterval for $C_{i-1} - (u, v)$.

The algorithm, implicit in the above argument, can be implemented to run in linear time by keeping a variable for each vertex x on C_i, counting the number of chords x is incident to. After each vertex v_i is chosen, the variables for all its neighbors can be updated in $O(deg(v_i))$ time. Summing over all vertices in the graph leads to an overall linear running time [2] and this concludes the proof of the theorem.
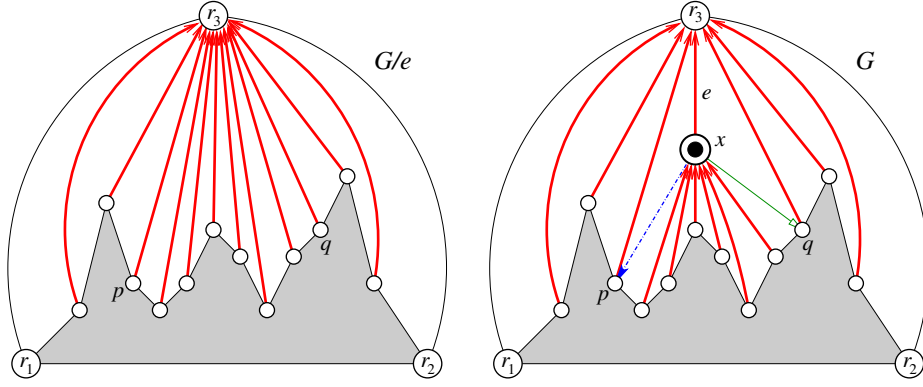
Schnyder Realizer

Let G be a maximal plane graph. A *Schnyder realizer* $\mathcal{S}$ of G is a partition of the internal edges of G into three sets T_1, T_2, T_3 of directed edges, so that for each interior vertex v, the following conditions hold:

(a) v has out-degree exactly one in each of T_1, T_2 and T_3,
(b) the clockwise order of edges incident to v is: outgoing T_1, incoming T_2, outgoing T_3, incoming T_1, outgoing T_2, incoming T_3; see Fig. 1(c).

Since a maximal plane graph has exactly $n - 3$ internal vertices and exactly $3n - 9$ internal edges, the three outgoing edges for each internal vertex imply that all the edges incident to the outer vertices are incoming. In fact, these two conditions imply that for each outer vertex r_i, $i = 1, 2, 3$, the incident edges belong to the same set, say T_i, where r_1, r_2, r_3 are in counter-clockwise order around the outerface and each set of edges T_i forms a directed tree, spanning all the internal vertices and one external vertex r_i, oriented towards r_i [3]; see Fig. 1(c). Call r_i the root of T_i for $i = 1, 2, 3$.

Note that the existence of a decomposition of a maximal planar graph G into three trees was proved earlier by Nash-Williams [8] and by Tutte [11]. Kampen [7] showed that these three trees can be oriented towards any three specified root vertices r_1, r_2, r_3 of G so that each vertex other than these roots have exactly one outgoing edges in each tree. Schnyder [9] proved the existence of the special decomposition defined above, along with a linear time algorithm to compute it. Before we describe the algorithm, we need to define the operation of *edge contraction*. Let G be a graph and $e = (x, y)$ be an edge of G. Then we denote by G/e, the simple graph obtained by

Fig. 2. Computing a Schnyder realizer of a maximal plane graph G from that of G/e.

deleting x, y and all their incident edges from G, adding a new vertex z and inserting an edge (z, v) for each vertex v that is adjacent to either x or y in G. Note that for a maximal plane graph G, contracting an edge $e = (x, y)$ yields a maximal plane graph if and only if there are exactly two common neighbors of x and y. Two end-vertices of an edge $e = (x, y)$ have exactly two common neighbors if and only if the edge e is not on the boundary of a separating triangle.

Lemma 2. *Let G be a maximal plane graph with at least 4 vertices, where u is an outer vertex of G. Then there exits an internal vertex v of G such that (u, v) is an edge in G and u, v have exactly two common neighbors.*

This is easy to prove. If G has exactly 4 vertices, then it is K_4 and the internal vertex of G has exactly two common neighbors. Consider graph G with more than 4 vertices. If u is not on the boundary of any separating triangle, then taking any neighbor of u as v is sufficient. Else, if u is on the boundary of a separating triangle Δ we can find a desired vertex v by induction on the subgraph of G inside Δ.

Theorem 2. *A Schnyder realizer of a maximal plane graph G can be computed in linear time.*

The proof of the theorem is by induction. If G has exactly 3 vertices, its Schnyder realizer is computed trivially. Consider graph G with more than 3 vertices. Let r_1, r_2 and r_3 be the three outer vertices in counter-clockwise order. Then by the above lemma, there is an internal vertex x in G and edge $e = (r_3, x)$ so that r_3 and x have exactly two common neighbors. Let $G' = G/e$. Then by the induction hypothesis, G' has a Schnyder realizer with the three trees T_1, T_2, T_3, rooted at r_1, r_2, r_3. We now modify this to find a Schnyder realizer for G. The orientation and partitioning of all the edges not incident to x remain unchanged from G/e. Among the edges incident to x, we add e to T_3, oriented towards r_3. We add the two edges, that are just counter-clockwise of e and just clockwise of e in the ordering around x, to T_1 and T_2, respectively, both oriented away from x. Finally we put all the remaining edges in T_3, oriented towards x; see Fig. 2. It is now straight-forward to check that these assignment of edges to the trees satisfy the two conditions. The algorithm implicit in the proof can be implemented in linear time, given the edge-contraction sequence. The edge-contraction sequence itself can be computed in linear time by taking the reverse order of a canonical order of the vertices and in every step contracting the edge between r_3 and the current vertex in this order.

Drawing Planar Graphs

We now show how canonical orders and Schnyder realizers can be used to compute straight-line grid drawings of maximal plane graphs.

Theorem 3. *Let G be a maximal plane graph with n vertices. A straight-line grid drawing of G on the $(2n-4) \times (n-2)$ grid can be computed in linear time.*

This is a constructive proof. Let $\mathcal{O} = v_1, \ldots, v_n$ be a canonical order of G, G_i, the subgraph of G induced by the vertices $v_1, \ldots, v_i$, and C_i the boundary of the outerface of G_i, $i = 3, 4, \ldots, n$. We incrementally obtain straight-line drawing Γ_i of G_i for $i = 3, 4, \ldots, n$. We also maintain the following invariants for Γ_i:

(i) the x-coordinates of the vertices on the path $C_i \setminus \{(v_1, v_2)\}$ are monotonically increasing as we go from v_1 to v_2.
(ii) each edge of the path $C_i - \{(v_1, v_2)\}$ is drawn with slope 1 or -1.

We begin with G_3, giving v_1, v_2 and v_3 coordinates $(0,0)$, $(2,0)$ and $(1,1)$; the drawing Γ_3 satisfies conditions (i)–(ii); see Fig. 3(a). Suppose the drawing for Γ_{i-1} for some $i > 3$ has already been computed; we now show how to obtain Γ_i. We need to add vertex v_i and its incident edges in G_i to Γ_{i-1}. Let $w_1 = v_1, \ldots, w_t = v_2$ be the vertices on $C_{i-1} \setminus \{(v_1, v_2)\}$ in this order from v_1 to v_2. By the property of canonical orders, v_i is adjacent to a subinterval of this path. Let $w_l, \ldots, w_r$, $1 \leq l < r \leq t$, be the vertices adjacent to v_i, in this order. We want to place v_i at the intersection point p between the straight-line from w_l with slope 1 and the straight-line from w_r with slope -1. Note that by condition (ii), the two vertices w_l and w_r are at even Manhattan distance in Γ and hence point p is a grid point. However, if we place v_i at p, then the edges $v_i w_l$ and $v_i w_r$ might overlap with the edges $w_l w_{l+1}$ and $w_{r-1} w_r$, since they are drawn with slopes 1 or -1. We thus shift all the vertices to the left of w_l in Γ_{i-1} (including w_l) one unit to the left at all the vertices to the right of w_r in Γ_{i-1} (including w_r) one unit to the right; see Fig. 3.

Consider a rooted tree T, spanning all the internal vertex of G along with one external vertex v_n, where v_n is the root of T and for each internal vertex x, the parent of x is the highest-numbered successor in G. (Later we see that T can be one of the three trees in a Schnyder realizer of G.) For any internal vertex x of G, denote by $U(x)$ the set of vertices that are in the subtree of T rooted at v (including v itself). Then the shifting of the vertices above can be obtained by shifting the vertices in $U(w_i)$, $i = 1, \ldots, l$ one unit to the left and the vertices in $U(w_i)$, $i = r, \ldots, t$ one unit to the right. After these shifts, v_i can be safely placed at the intersection of the line with slope 1 from w_l and the line with slope -1 from w_r. Note that this drawing satisfies conditions (i)–(ii). This algorithm can be implemented in linear time, even though the efficient vertex shifting requires careful relative offset computation [2].

The next theorem shows how Schnyder realizers can be used to compute a straight-line grid drawing of a plane graph. Let T_1, T_2 and T_3 be the trees in a Schnyder realizer of G, rooted at outer vertices r_1, r_2 and r_3. Since each internal vertex v of G has exactly one outgoing edge in each of the trees, there is a directed path $P_i(v)$ in each of the three trees T_i from v to r_i. These three paths $P_1(v)$, $P_2(v)$, $P_3(v)$ are vertex-disjoint except for v and they define three regions $R_1(v)$, $R_2(v)$, $R_3(v)$ for v. Here $R_i(v)$ is the region between the two paths $P_{i-1}(v)$ and $P_{i+1}(v)$, where the addition and subtraction are modulo 3. Let $\eta_i(v)$ denote the number of vertices in $R_i(v) \setminus P_{i-1}(v)$, where $i = 1, 2, 3$ and the subtraction is modulo 3. Extend these definitions to the outer vertices as follows: $\eta_i(r_i) = n - 2$, $\eta_{i+1}(r_i) = 1$, $\eta_{i-1}(r_i) = 0$.

Theorem 4. *The coordinates $((\eta_1(v), \eta_2(v)))$ for each vertex v in G give a straight-line drawing Γ of G on a grid of size $(n-2) \times (n-2)$.*

Place each vertex v at the point with coordinates $(\eta_1(v), \eta_2(v), \eta_3(v))$. Since $\eta(v) = \eta_1(v) + \eta_2(v) + \eta_3(v)$ counts the number of vertices in all the three regions of v, each
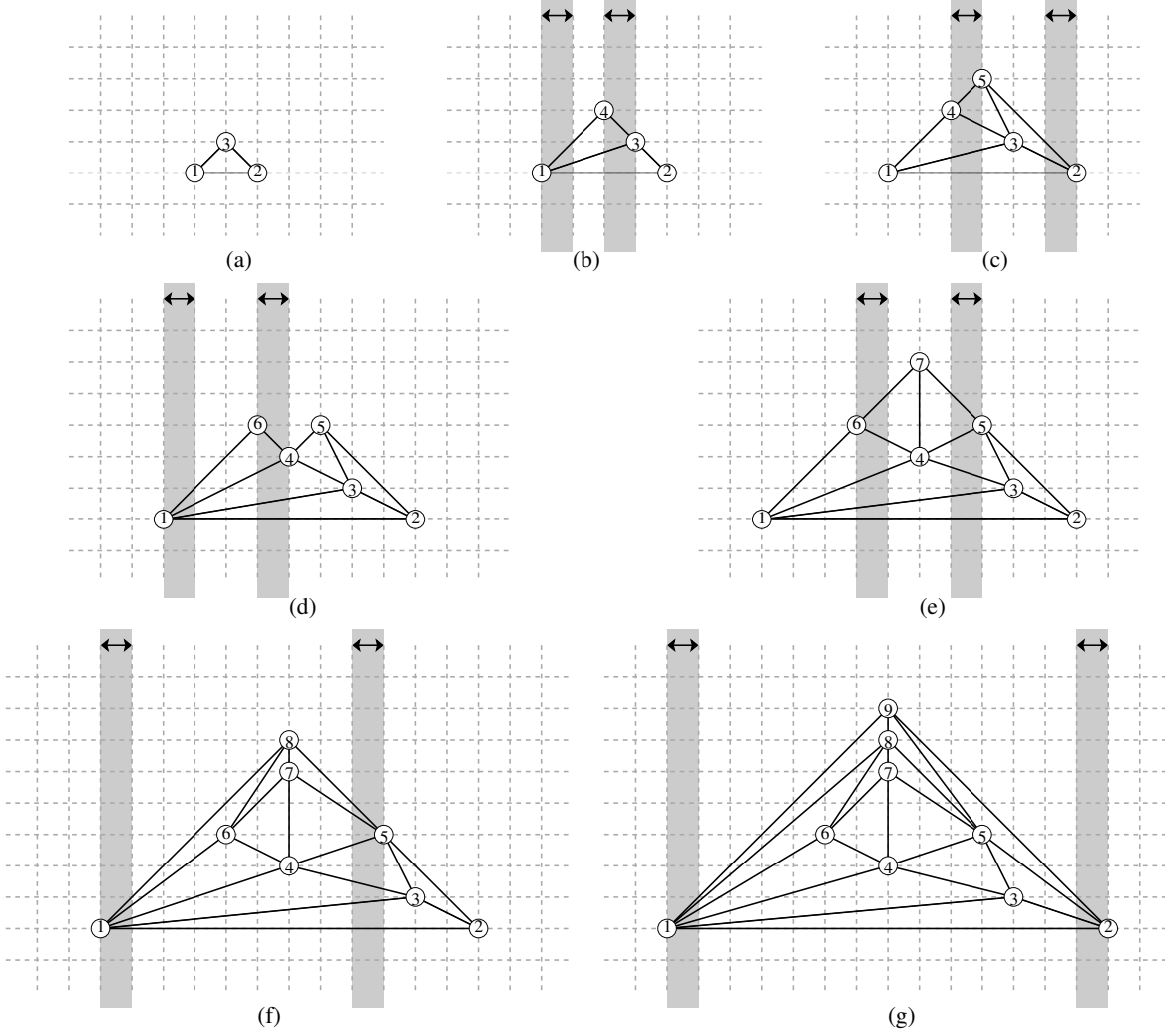
**Fig. 3.** Illustration for the straight-line drawing algorithm using canonical order.

vertex of G except v is counted exactly once in $\eta(v)$. Thus $\eta_1(v) + \eta_2(v) + \eta_3(v) = n-1$ for each vertex v. Thus the drawing Γ' obtained by these coordinates $(\eta_1(v), \eta_2(v), \eta_3(v))$ lies on the plane $x + y + z = n-1$. Furthermore, the drawing does not induce any edge-crossings; see [9]. Thus Γ' is a straight-line drawing of G on the plane $x + y + z = n-1$. Then Γ is just a projection of Γ' on the plane $z = 0$ and hence is planar. Since each coordinate in the drawing is bounded between 0 and $n-1$, the area is at most $(n-2) \times (n-2)$; see Fig. 4(a).

Equivalency of Canonical Orders and Schnyder Realizers

Here we show that canonical orders and Schnyder realizers, are in fact equivalent, in the sense that a canonical order of a graph defines a Schnyder realizer and vice versa [3].

Lemma 3. *A canonical order for a maximal plane graph G defines a unique Schnyder realizer where the three parents for each vertex v of G are its leftmost predecessor, its rightmost predecessor and its highest-labeled successor.*

See Fig. 1(a) and (c) for a canonical order $\mathcal{O}$ and the corresponding Schnyder realizer $\mathcal{S}$ defined by $\mathcal{O}$ for a maximal plane graph. One can easily verify that this definition of $\mathcal{S}$ satisfies the two conditions for each internal vertex for a maximal plane graph. A canonical order $\mathcal{O}$ and the Schnyder realizer $\mathcal{S}$ obtained from $\mathcal{O}$ for a maximal plane graph are said to be *compatible*.
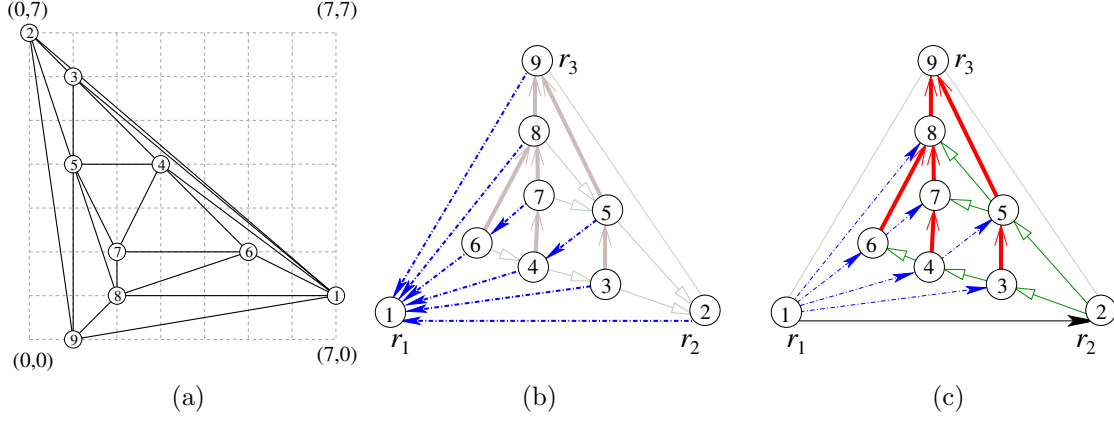
Fig. 4. (a) A straight-line drawing for the graph in Fig. 1 using Schnyder realizer, (b)–(c) computation of a canonical order from a Schnyder realizer.

We now describe two ways to obtain a canonical order from a Schnyder realizer $\mathcal{S}$. In both cases we obtain a canonical order which is compatible with $\mathcal{S}$.

Lemma 4. *Let G be a maximal plane graph with outer vertices r_1, r_2, r_3 in counter-clockwise order and let T_1, T_2, T_3 be the three trees in a Schnyder realizer of G rooted at r_1, r_2 and r_3. Then a compatible canonical order of G can be obtained as follows:*

1. *by taking the counter-clockwise depth-first traversal order of the vertices in the graph $T_1 \cup \{(r_2, r_1), (r_3, r_1)\}$; see Fig. 4(b).*
2. *by taking the topological order of the directed acyclic graph $T_1^{-1} \cup T_2^{-1} \cup T_3$, where T_i^{-1}, $i = 1, 2$ is the Schnyder tree T_i with reversed edge directions; see Fig. 4(c).*

It is not difficult to show that the directed graph $T_1^{-1} \cup T_2^{-1} \cup T_3$ is in fact acyclic [3; 9]. Then it is easy to verify that the canonical orders obtained from a Schnyder realizer $\mathcal{S}$ are compatible with $\mathcal{S}$; i.e., defining a Schnyder realizer from them by Lemma 3 produces the original Schnyder realizer $\mathcal{S}$.

Recommended Reading

1. I. Baybars. On k-path hamiltonian maximal planar graphs. *Discrete Mathematics*, 40(1):119–121, 1982.
2. M. Chrobak and T. H. Payne. A linear-time algorithm for drawing a planar graph on a grid. *Information Processing Letters*, 54(4):241–246, 1995.
3. H. de Fraysseix and P. O. de Mendez. On topological aspects of orientations. *Discrete Mathematics*, 229(1–3):57–72, 2001.
4. H. de Fraysseix, J. Pach, and R. Pollack. How to draw a planar graph on a grid. *Combinatorica*, 10(1):41–51, 1990.
5. I. Fáry. On straight lines representation of planar graphs. *Acta Scientiarum Mathematicarum*, 11:229–233, 1948.
6. J. E. Hopcroft and R. E. Tarjan. Efficient planarity testing. *Journal of the ACM*, 21(4):549–568, 1974.
7. G. Kampen. Orienting planar graphs. *Discrete Mathematics*, 14(4):337–341, 1976.
8. C. S. J. A. Nash-Williams. Edge-disjoint spanning trees of finite graphs. *Journal of the London Mathematical Society*, 36:445–450, 1961.
9. W. Schnyder. Embedding planar graphs on the grid. In *ACM-SIAM Symposium on Discrete Algorithms (SODA 1990)*, pages 138–148, 1990.
10. S. K. Stein. Convex maps. *American Mathematical Society*, 2(3):464–466, 1951.
11. W. T. Tutte. On the problem of decomposing a graph into n connected factors. *Journal of the London Mathematical Society*, 36:221–230, 1961.
12. K. Wagner. Bemerkungen zum Vierfarbenproblem. *Jahresbericht der Deutschen Mathematiker-Vereinigung*, 46:26–32, 1936.