Efficient Enumeration of Drawings and Combinatorial Structures for Maximal Planar Graphs

Giordano Da Lozzo, Giuseppe Di Battista, Fabrizio Frati$^{(\boxtimes)}$, Fabrizio Grosso, and Maurizio Patrignani

Roma Tre University, Rome, Italy
{giordano.dalozzo,giuseppe.dibattista,fabrizio.frati,fabrizio.grosso,
maurizio.patrignani}@uniroma3.it

Abstract. We propose efficient algorithms for enumerating the celebrated combinatorial structures of maximal planar graphs, called canonical orderings and Schnyder woods, and the related classical graph drawings by de Fraysseix, Pach, and Pollack [Combinatorica, 1990] and by Schnyder [SODA, 1990], called canonical drawings and Schnyder drawings, respectively. To this aim (i) we devise an algorithm for enumerating special e-bipolar orientations of maximal planar graphs, called *canonical orientations*; (ii) we establish bijections between canonical orientations and canonical drawings, and between canonical orientations and Schnyder drawings; and (iii) we exploit the known correspondence between canonical orientations and canonical orderings, and the known bijection between canonical orientations and Schnyder woods. All our enumeration algorithms have $\mathcal{O}(n)$ setup time, space usage, and delay between any two consecutively listed outputs, for an n-vertex maximal planar graph.

Keywords: Enumeration algorithms · Planar graphs · Canonical orderings · Schnyder woods · Worst-case delay

1 Introduction and Overview

An *enumeration algorithm* lists all the solutions of a problem, without duplicates, and then stops. Its efficiency is measured in terms of setup time, space usage, and *delay* between the outputs of two consecutive solutions; see, e.g., [3, 42, 50, 58]. In this paper, we present efficient algorithms to enumerate: (i) straight-line grid drawings produced with the algorithms by de Fraysseix, Pach, and Pollack [25, 26] (*FPP-algorithm*) and by Schnyder [52] (*S-algorithm*), and (ii) the corresponding combinatorial structures. To the best of our knowledge, these are the first enumeration algorithms for drawings of graphs.

Notable applications of graph drawing enumeration algorithms include:
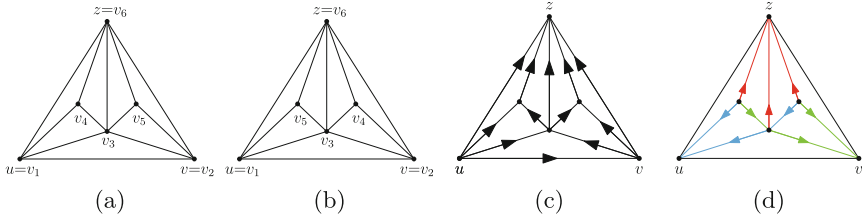
Fig. 1. (a), (b) The two canonical orderings with first vertex u of a maximal planar graph G. (c) The unique canonical orientation with first vertex u of G. (d) The unique Schnyder wood of G.

(1) Users can benefit from having multiple alternative drawings that highlight different features of the graph, enabling them to choose the most suitable for their specific needs; this strategy has been used already in [8].
(2) Machine-learning-based graph drawing tools profit from multiple drawings of a graph; small-delay enumeration algorithms may fuel the training process of these tools.
(3) Computer-aided systems that aim to verify geometric and topological statements can leverage enumeration algorithms to explore the solution space of graph drawing problems.

Preliminary Definitions. We consider graphs and digraphs with multiple edges. A *maximal planar graph* is a planar graph without parallel edges to which no edge can be added without losing planarity or simplicity. A *plane graph* is a planar graph with a prescribed embedding.

Let G be a maximal plane graph and let (u, v, z) be the cycle delimiting its outer face, where u, v, and z appear in this counter-clockwise order along the cycle. A *canonical ordering of G with first vertex u* is a labeling of the vertices $v_1 = u, v_2 = v, v_3, \ldots, v_{n-1}, v_n = z$ such that, for every $3 \le k \le n - 1$ (see Figs. 1(a), 1(b), and [26]):

(CO-1) The plane subgraph $G_k \subseteq G$ induced by $v_1, v_2, \ldots, v_k$ is 2-connected; let C_k be the cycle bounding its outer face;
(CO-2) v_{k+1} is in the outer face of G_k, and its neighbors in G_k form an (at least 2-element) subinterval of the path $C_k - (u, v)$.

A *canonical ordering* of G is a canonical ordering of G with first vertex x, where $x \in \{u, v, z\}$. If G' is a maximal planar graph, a *canonical ordering* of G' is a canonical ordering of a maximal plane graph isomorphic to G'. Let $\pi = (v_1, \ldots, v_n)$ be a canonical ordering of G with first vertex v_1. Orient every edge (v_i, v_j) of G from v_i to v_j if and only if $i < j$. The resulting orientation is the *canonical orientation of G with respect to π*. An orientation $\mathcal{D}$ of G is a *canonical orientation with first vertex u* if there exists a canonical ordering π of G with first vertex u such that $\mathcal{D}$ is the canonical orientation of G with respect to π; see Fig. 1(c). A *canonical orientation* of G is a canonical orientation with

first vertex x, where x is a vertex in $\{u, v, z\}$. Finally, if G' is a maximal planar graph, a *canonical orientation* of G' is a canonical orientation of a maximal plane graph isomorphic to G'.

A vertex or edge of G is *internal* if it is not incident to the outer face and *outer* otherwise. A *Schnyder wood* of G is an assignment of directions and of the colors 1, 2 and 3 to the internal edges of G such that (see Fig. 2 and [52]):

(S-1) For $i = 1, 2, 3$, each internal vertex x has one outgoing edge e_i of color i. The outgoing edges e_1, e_2, and e_3 appear in this counter-clockwise order at x. Further, for $i = 1, 2, 3$, all the incoming edges at x of color i appear in the clockwise sector between e_{i+1} and e_{i-1}, where $i + 1 = 1$ if $i = 3$ and $i - 1 = 3$ if $i = 1$.

(S-2) At the outer vertices u, v, and z, all the internal edges are incoming and of color 1, 2, and 3, respectively. If G' is a maximal planar graph, a *Schnyder wood* of G' is a Schnyder wood of a maximal plane graph isomorphic to G'.
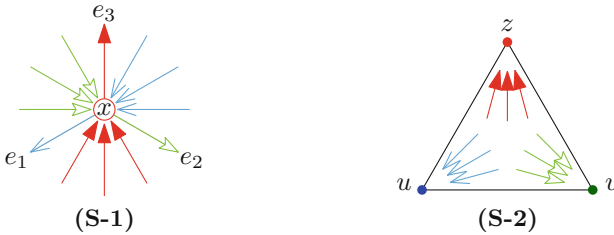

Fig. 2. Illustration for the properties of a Schnyder wood.

Our Contributions. First, we present an algorithm that enumerates all canonical orientations of an n-vertex maximal plane graph G by applying edge contraction or removals. This results in smaller graphs, whose canonical orientations are recursively enumerated and modified to obtain canonical orientations of G by orienting the contracted or removed edges. To achieve polynomial delay, contractions and removals should be applied only if the corresponding branch of computation produces at least one canonical orientation of G. We determine necessary and sufficient conditions for a subgraph of G to allow for an orientation that can be extended to a canonical orientation of G. We establish topological properties that determine whether applying a contraction or a removal results in a graph satisfying these conditions. Additionally, we create efficient data structures for testing and applying the operations based on these properties.

Second, we prove that canonical orderings are topological sortings of canonical orientations. This allows our algorithm for enumerating the former to be used for enumerating the latter. Moreover, since canonical orientations are in correspondence with Schnyder woods [22, Theorem 3.3], our algorithm for enumerating canonical orientations can also be used to enumerate all Schnyder woods of G.

Third, we show that applying the FPP-algorithm with different canonical orderings corresponding to the same canonical orientation yields the same drawing of G. This establishes a correspondence between the canonical orientations of G and the drawings produced by the FPP-algorithm. Together with our algorithm for enumerating canonical orientations, this allows us to enumerate such drawings.

Finally, we show that the drawings generated by the S-algorithm are in bijection with the Schnyder woods. This, the bijection between canonical orientations and Schnyder woods, and our algorithm for enumerating canonical orientations enable us to enumerate the drawings of G produced by the S-algorithm.

All our enumeration algorithms have $\mathcal{O}(n)$ setup time, space usage, and worst-case delay.

Related Results. The planar straight-line drawings of maximal planar graphs generated by the FPP-algorithm [25,26] and by the S-algorithm [52] are fundamental in graph drawing [28,45,57] and find applications in other fields, e.g., knot theory [15,36,37] and computational complexity [7,34,51]. Further, the combinatorial structures conceived for these algorithms, i.e., canonical orderings and Schynder woods, are used for a plethora of problems in graph drawing [1,2,4,20,23,27,30,31,33,41,46] and beyond [5,11–13,18,38,39]. Canonical orderings and Schnyder woods appear to be distant concepts. However, Schnyder [52] has shown how to get a Schnyder wood from a canonical ordering. Also, their relationship is explained by the concept of canonical orientations, which in [22] are proved to be in bijection with the Schnyder woods; see also [44].

While enumerating graph drawings is a novel subject, the enumeration of graph orientations has a rich literature. The enumeration of acyclic orientations and k-arc-connected orientations are studied in [6,19,55] and [10], respectively. An *st-orientation* of a graph G is an acyclic orientation of G such that s and t are its unique source and sink, respectively. In [24] a polynomial-delay algorithm is provided for enumerating *st*-orientations. In [53], the algorithm in [24] is refined to obtain linear delay, if the input is biconnected and planar. Our paper is connected to these algorithms through a result by de Fraysseix and Ossona de Mendez [22]: There is a bijection between the canonical orientations and the bipolar orientations *such that every internal vertex has at least two incoming edges.* Our enumeration algorithm for canonical orientations follows the strategy of [24,53] for enumerating bipolar orientations of biconnected planar graphs. However, requiring that every internal vertex has at least two incoming edges dramatically increases the complexity of the problem and reveals new and, in our opinion, interesting topological properties of the desired orientations.

The canonical orientations of a maximal plane graph form a distributive lattice $\mathcal{L}$ [32]. By the fundamental theorem of finite distributive lattices [9], there is a finite poset P whose order ideals correspond to the elements of $\mathcal{L}$ and it is known that $|P|$ is polynomial in n [32, page 10]. Enumerating the order ideals of P is a studied problem. In [35] an algorithm is presented that lists all order ideals of P in $\mathcal{O}(\Delta(P))$ delay, where $\Delta(P)$ is the maximum indegree of the covering graph of P. However, the algorithm has three drawbacks that make it

unsuitable for solving our problems. First, the guaranteed delay of the algorithm is amortized, and not worst-case. Second, the algorithm uses $\mathcal{O}(w(P) \cdot |P|) = \mathcal{O}(n^3)$ space, where $w(P) = \mathcal{O}(n)$ is the width of P, and $\mathcal{O}(|P|^2) = \mathcal{O}(n^4)$ preprocessing time. Third and most importantly, each order ideal is produced twice by the algorithm, rather than just once as required by an enumeration algorithm. Similarly, the algorithms in [48,54,56] are affected by all or by part of the above drawbacks.

Open Problems. Our research sparkles new questions. In general, for a graph G and a drawing style $\mathcal{D}$, we may ask for algorithms to enumerate the drawings of G respecting $\mathcal{D}$. Examples include: (i) enumerating the planar straight-line drawings of a planar graph on a given grid; (ii) enumerating the orthogonal representations of a plane graph with at most b bends; and (iii) enumerating the upward planar embeddings of a single-source or triconnected DAG.

Full details of omitted or sketched proofs can be found in the full version of the paper [21].

2   Canonical Orientations

In [22, Lemma 3.6–3.7, Theorem 3.3], the following characterization has been shown, for which we provide an alternative proof in the full version of the paper.

Theorem 1. ([22]). *Let G be an n-vertex maximal plane graph and let (u, v, z) be the cycle delimiting its outer face, where u, v, and z appear in this counterclockwise order along the cycle. An orientation $\mathcal{D}$ of G is a canonical orientation with first vertex u if and only if $\mathcal{D}$ is a uz-orientation in which every internal vertex has at least two incoming edges.*

Our proof of Theorem 1 also implies the following.

Lemma 1. *Any topological sorting of a canonical orientation with first vertex u of a maximal plane graph G is a canonical ordering of G with first vertex u.*

Let h_1 and h_2 be parallel edges in a plane graph. We denote by $\ell(h_1, h_2)$ the open region bounded by h_1 and h_2, and say that $\ell(h_1, h_2)$ is a *multilens* if it contains no vertices in its interior. Note that $\ell(h_1, h_2)$ might contain edges parallel to h_1 and h_2 in its interior, or it might coincide with an internal face of the graph; see Fig. 3. We now provide some crucial definitions.

Definition 1. *A biconnected plane graph G with two distinguished vertices s and t is called* well-formed *if it satisfies the following conditions (refer to Fig. 3):*

(1) s and t (which are called the poles *of G) are incident to the outer face of G and s immediately precedes t in clockwise order along the cycle C_o bounding such a face;*

(2) all the internal faces of G have either two or three incident vertices;

(3) multiple edges, if any, are all incident to s; and
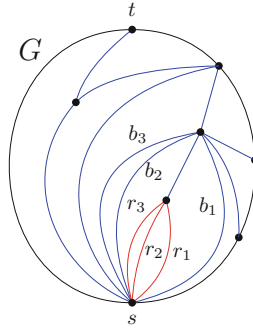
Fig. 3. Any two of the edges r_1, r_2, and r_3 form a multilens, the edges b_2 and b_3 form a multilens, whereas neither b_2 nor b_3 forms a multilens with b_1; the multilenses $\ell(r_1, r_2)$, $\ell(r_2, r_3)$, and $\ell(b_2, b_3)$ are also faces.

(4) if two parallel edges h_1 and h_2 with end-vertices s and x exist such that $\ell(h_1, h_2)$ is not a multilens, then two parallel edges h_1' and h_2' between s and a vertex $y \neq x$ exist such that $\ell(h_1', h_2')$ is a multilens and such that $\ell(h_1', h_2') \subset \ell(h_1, h_2)$.

Definition 2. *An st-orientation $\mathcal{D}$ of a well-formed biconnected plane graph G with poles s and t is* inner-canonical *if every internal vertex has at least two incoming edges in $\mathcal{D}$.*

Definition 3. *Let G be a plane graph. The* contraction of *an edge $e = (u, v)$ removes e from G and "merges" u and v into a new vertex w. Let $e, e_1^u, \ldots, e_h^u$ and $e, e_1^v, \ldots, e_k^v$ be the clockwise order of the edges incident to u and to v, respectively. Then the clockwise order of the edges incident to w is $e_1^u, \ldots, e_h^u, e_1^v, \ldots, e_k^v$.*

Note that the contraction of an edge may introduce parallel edges or self-loops.

Let G be a well-formed biconnected plane graph with poles s and t. Let $e_1, e_2, \ldots, e_m$ be the counter-clockwise order of the edges incident to s, where e_1 and e_m are the rightmost and leftmost edge incident to s, respectively, and let $v_1, \ldots, v_m$ be the end-vertices of $e_1, \ldots, e_m$ different from s, respectively. Moreover, let G^* be the plane multigraph resulting from the contraction of e_1 in G; see Fig. 4(a). Also, if G contains parallel edges, let $j \in \{1, \ldots, m-1\}$ be the smallest index such that e_j and e_{j+1} define a multilens of G; denote by G^- the plane graph resulting from the removal of $e_1, \ldots, e_j$ from G; see Fig. 4(b). The next lemmas prove that, under certain conditions, G^* and G^- are well-formed multigraphs and can be used to obtain inner-canonical orientations of G.

Lemma 2. *Suppose that G does not contain parallel edges between s and w_1, where w_1 is the vertex that follows s in counter-clockwise direction along the outer face of G. Then G^* is a well-formed biconnected plane graph with poles s and t. Also, let $\mathcal{D}^*$ be an inner-canonical orientation of G^*. The orientation $\mathcal{D}$ of G*
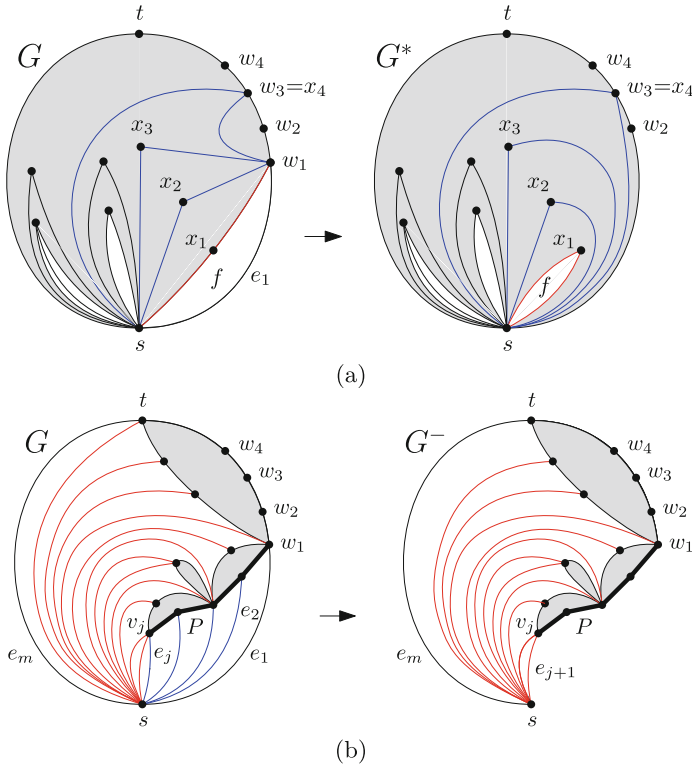
Fig. 4. Illustration for the contraction of e_1 (a) and the removal of $e_1, \ldots, e_j$ (b).

obtained from $\mathcal{D}^*$ by orienting the edge (s, w_1) away from s and by keeping the orientation of all other edges unchanged is inner-canonical.

Lemma 3. *Suppose that G contains parallel edges and let $j \in \{1, \ldots, m-1\}$ be the smallest index such that e_j and e_{j+1} define a multilens of G. Suppose also that either $j = 1$, or $j > 1$ and $v_2, \ldots, v_j$ are not incident to the outer face of G. Then the graph G^- is a well-formed biconnected plane graph with poles s and t. Also, let $\mathcal{D}^-$ be an inner-canonical orientation of G^-. The orientation $\mathcal{D}$ of G obtained from $\mathcal{D}^-$ by orienting the edges $e_1, e_2, \ldots, e_j$ away from s and by keeping the orientation of all other edges unchanged is inner-canonical.*

By induction on $|E(G)|$ and using Lemmas 2 and 3 we can prove the following.

Lemma 4. *Every well-formed biconnected plane graph G with poles s and t has at least one inner-canonical orientation.*

Section 2.1 is devoted to the proof of the following main result.

Theorem 2. *Let G be a well-formed biconnected plane graph with φ edges. There is an algorithm with $\mathcal{O}(\varphi)$ setup time and space usage listing all the inner-canonical orientations of G with $\mathcal{O}(\varphi)$ delay.*

Provided that Theorem 2 holds, we can prove the following.

Lemma 5. *Let G be an n-vertex maximal plane graph and (u, v, z) be the cycle delimiting its outer face f_o, where u, v, and z appear in this counter-clockwise order along f_o. There is an algorithm with $\mathcal{O}(n)$ setup time and space usage listing all the canonical orientations of G with first vertex u with $\mathcal{O}(n)$ delay.*

Proof (sketch): We have that G is a well-formed biconnected plane graph with poles u and z. By Theorem 1, any canonical orientation of G with first vertex u is a uz-orientation such that every internal vertex has at least two incoming edges, i.e., an inner-canonical orientation. Also, any inner-canonical orientation of G is canonical. This, combined with G having $\mathcal{O}(n)$ edges, implies that the algorithm in Theorem 2 enumerates all canonical orientations of G within the stated bounds. □

Theorem 3. *There is an algorithm $\mathcal{A}_1$ (resp. $\mathcal{A}_2$) with $\mathcal{O}(n)$ setup time and space usage listing all canonical orientations of an n-vertex maximal plane (planar) graph with $\mathcal{O}(n)$ delay.*

Proof (sketch): Algorithm $\mathcal{A}_1$ uses the algorithm for the proof of Lemma 5 three times, i.e., once for each choice of the first vertex among the three vertices incident to the outer face of the input graph G. Algorithm $\mathcal{A}_2$ applies $4n - 8$ times algorithm $\mathcal{A}_1$, since there are $4n - 8$ maximal plane graphs that are isomorphic to G. □

2.1 The Inner-Canonical Enumerator Algorithm

We now describe the INNER-CANONICAL ENUMERATOR (ICE) algorithm that enumerates all the inner-canonical orientations of a well-formed biconnected plane graph G with poles s and t (see Theorem 2). In the full version of the paper [21], we provide implementation details, data structures, and pseudocode for achieving the claimed worst-case bounds.

The ICE algorithm works recursively as follows. In the base case, G is the single edge $e_m = (s, t)$, and its unique inner-canonical orientation is the one with the edge e_m directed from s to t. Otherwise, the algorithm considers four cases.

In Cases 1 and 2, G contains parallel edges and e_1 is the unique edge between s and w_1. Let $j \in \{2, \ldots, m-1\}$ be the smallest index such that e_j and e_{j+1} define a multilens of G. In Case 1, there exists an index $i \in \{2, \ldots, j\}$ such that v_i is incident to the outer face of G, while in Case 2 such an index does not exist. In Case 3, G does not contain parallel edges. Finally, in Case 4, G contains parallel edges between s and w_1. Note that exactly one of Cases 1–4 applies to G.

In Cases 1 and 3, **we contract** the edge (s, w_1). Let G^* be the resulting plane graph which, by Lemma 2, is biconnected and well-formed. Thus, the algorithm can be applied recursively to enumerate all inner-canonical orientations of G^*. The algorithm obtains all inner-canonical orientations of G from the ones of G^* as in Lemma 2.

In Case 4, **we remove** the edges $e_1, e_2, \ldots, e_j$. Let G' be the resulting plane graph which, by Lemma 3, is biconnected and well-formed. Thus, the algorithm can be applied recursively in order to enumerate all inner-canonical orientations of G'. The algorithm obtains all inner-canonical orientations of G from the ones of G' as in Lemma 3.

In Case 2, the algorithm branches and applies **both the contraction and the removal operations**. Precisely, first we contract the edge (s, w_1), obtaining a well-formed biconnected plane graph G^*. After all inner-canonical orientations of G^* have been used to produce inner-canonical orientations of G as in Lemma 2, we remove the edges $e_1, e_2, \ldots, e_j$ from G, obtaining a well-formed biconnected plane graph G', from which the remaining inner-canonical orientations of G are produced as in Lemma 3.

Note that the ICE algorithm outputs an inner-canonical orientation each time the base case applies. The next lemma summarizes its correctness.

Lemma 6. *The* ICE *algorithm outputs all and only the inner-canonical orientations of G without repetitions.*

3   Enumeration of Canonical Orderings and Drawings

We show how to efficiently enumerate the canonical orderings and drawings of a maximal plane or planar graph G. By Theorem 3, the canonical orientations of G can be generated efficiently. By Lemma 1, for every canonical orientation $\mathcal{D}$ of G, the canonical orderings π of G such that $\mathcal{D}$ is the canonical orientation of G with respect to π are the topological sortings of $\mathcal{D}$. Since there exist $\mathcal{O}(1)$-delay algorithms [47,49] for listing all such topological sortings, we get the following.

Theorem 4. *There is an algorithm with $\mathcal{O}(n)$ setup time and space usage listing all canonical orderings of an n-vertex maximal plane/planar graph with $\mathcal{O}(n)$ delay.*
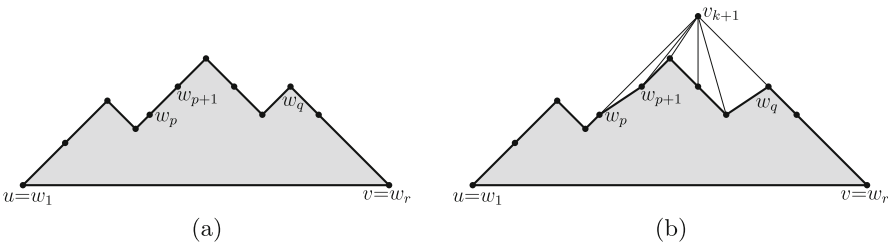
Fig. 5. Illustrations for the FPP-algorithm. (a) Γ_k. (b) Γ_{k+1}.

We show how to enumerate the planar straight-line drawings produced by the *FPP-algorithm* [26]. Its input is an n-vertex maximal plane graph G, whose outer face is delimited by a cycle (u, v, z) and a canonical ordering $\pi = (v_1 = u, v_2 = v, v_3, \ldots, v_n = z)$ of G. The FPP-algorithm works in steps.

The first step constructs a planar straight-line drawing Γ_3 of G_3 with v_1, v_2, and v_3 at $(0,0)$, $(2,0)$, and $(1,1)$, respectively, and defines sets $M_3(v_1) = \{v_1, v_2, v_3\}$, $M_3(v_3) = \{v_2, v_3\}$, and $M_3(v_2) = \{v_2\}$.

For $k = 3, \ldots, n-1$, step $k-1$ constructs a planar straight-line drawing Γ_{k+1} of G_{k+1} by modifying Γ_k as follows; see Fig. 5. Let $w_1 = u, w_2, \ldots, w_r = v$ be the clockwise order of the vertices along the outer face of G_k. Assume that step $k-2$ has defined, for $i = 1, \ldots, r$, a subset $M_k(w_i)$ of the vertices of G_k, where $M_k(w_1) \supset \cdots \supset M_k(w_r)$. Let $w_p, w_{p+1}, \ldots, w_q$ be the neighbors of v_{k+1} in G_k, where $p < q$. Then Γ_{k+1} is obtained from Γ_k by increasing the x-coordinate of each vertex in $M_k(w_{p+1})$ by one unit, the x-coordinate of each vertex in $M_k(w_q)$ by one additional unit, and placing v_{k+1} at the intersection of the line through w_p with slope $+1$ and the line through w_q with slope -1. Then step $k-1$ proceeds to define the sets:

(i) $M_{k+1}(w_i) = M_k(w_i) \cup \{v_{k+1}\}$, for $i = 1, \ldots, p$;
(ii) $M_{k+1}(v_{k+1}) = M_k(w_{p+1}) \cup \{v_{k+1}\}$; and
(iii) $M_{k+1}(w_i) = M_k(w_i)$, for $i = q, \ldots, r$.

We call *canonical drawing with base edge (u, v)* the drawing Γ_n of G constructed by the FPP-algorithm; we often say that Γ_n *corresponds* to π. The following is the main tool for our enumeration algorithm.

Theorem 5. *Let G be an n-vertex maximal plane graph and (u, v, z) be the cycle delimiting the outer face of G, with u, v, and z in this counter-clockwise order along the cycle. There is a bijective function f from the canonical orientations of G with first vertex u to the canonical drawings of G with base edge (u, v). Also, given a canonical orientation with first vertex u, the corresponding canonical drawing with base edge (u, v) can be constructed in $\mathcal{O}(n)$ time.*

Proof (sketch): The function f is as follows. Consider any canonical orientation $\mathcal{D}$ of G with first vertex u and let π be any canonical ordering with first vertex u that *extends* $\mathcal{D}$ (that is, the canonical orientation of G with respect π is $\mathcal{D}$). Then $f(\mathcal{D})$ is the canonical drawing with base edge (u, v) that corresponds to π. Since π can be computed as any topological sorting of $\mathcal{D}$ in $\mathcal{O}(n)$ time [40] and the FPP-algorithm can be implemented in $\mathcal{O}(n)$ time [17], the second part of the statement follows. Clearly, f is injective. Indeed, any distinct canonical orientations $\mathcal{D}_1$ and $\mathcal{D}_2$ of G differ on the orientation of some edge (a, b). Thus, for any two canonical orderings π_1 and π_2 that extend $\mathcal{D}_1$ and $\mathcal{D}_2$, respectively, we have that b follows a in π_1 and precedes a in π_2 (or vice versa). Hence, the y-coordinate of b is larger than the one of a in $f(\mathcal{D}_1)$ and smaller than the one of a in $f(\mathcal{D}_2)$ (or vice versa), thus $f(\mathcal{D}_1)$ and $f(\mathcal{D}_2)$ are not the same drawing. The core of the proof that f is surjective is in the proof of the following statement.

Claim 1. *Any two canonical orderings π and τ that extend $\mathcal{D}$ are such that the canonical drawings of G corresponding to π and τ are the same drawing.*

The proof of Claim 1 is by induction on $|V(G)|$ and it relies on a natural extension of the concepts of canonical ordering, orientation, and drawing to biconnected

internally-triangulated plane graphs; hence, in the following, the outer face of G might have more than three incident vertices. The proof of Claim 1 exploits the following claim, which is also proved by induction on the size of G.

Claim 2. *Let $z_1, \ldots, z_r$ be the clockwise order of the vertices along the outer face of G, let $M^\pi(z_1), \ldots, M^\pi(z_r)$ (let $M^\tau(z_1), \ldots, M^\tau(z_r)$) be the sets associated to $z_1, \ldots, z_r$, respectively, by the FPP-algorithm, when applied with canonical ordering π (resp. τ). For $i = 1, \ldots, r$, the sets $M^\pi(z_i)$ and $M^\tau(z_i)$ coincide.*

The inductive proof of Claim 1 distinguishes two cases.

In Case 1, π and τ have the same last vertex. This can be removed from both, resulting in canonical orderings λ and ξ, respectively, of a smaller graph G'. By induction, the canonical drawings of G' corresponding to λ and ξ coincide. This and the fact that the sets associated to the vertices along the boundary of G' by the FPP-algorithm when applied with canonical orderings λ and ξ coincide imply that the canonical drawings of G corresponding to π and τ also coincide.

In Case 2, π and τ do not have the same last vertex. Then we define a sequence of canonical orderings of G such that: (i) the first canonical ordering in the sequence is τ; (ii) any two canonical orderings consecutive in the sequence coincide, except for two vertices, whose positions are adjacent and swapped in the two canonical orderings; and (iii) the last canonical ordering in the sequence has the same last vertex as π. Note that the last canonical ordering in the sequence and π are such that the corresponding canonical drawings of G are the same drawing, by Case 1. The proof that the canonical drawings of G corresponding to two consecutive canonical orderings in the sequence are the same drawing relies on the similarity of such canonical orderings. By transitivity, we get that the canonical drawings of G corresponding to π and τ are the same drawing. □

Theorem 3 and Theorem 5 imply the following.

Theorem 6. *There is an algorithm with $\mathcal{O}(n)$ setup time and space usage listing all canonical drawings of an n-vertex maximal plane/planar graph with $\mathcal{O}(n)$ delay.*

4   Enumeration of Schnyder Woods and Drawings

We now show how to efficiently enumerate the Schnyder woods and drawings of an n-vertex maximal plane graph G. The Schnyder woods and the canonical orientations of G are in bijection [22]. Further, given a canonical orientation of G, the corresponding Schnyder wood of G can be constructed in $\mathcal{O}(n)$ time [14, 16,26,27,29,43,52]. This, together with Theorem 3, implies the following.

Theorem 7. *There is an algorithm with $\mathcal{O}(n)$ setup time and space usage listing all Schnyder woods of an n-vertex maximal plane/planar graph with $\mathcal{O}(n)$ delay.*

We now deal with the enumeration of the planar straight-line drawings produced by the algorithm by Schnyder [52], known as *Schnyder drawings*. The S-algorithm takes as input (see Fig. 6(a)) an n-vertex maximal plane graph G,

whose outer face is delimited by a 3-cycle (u_1, u_2, u_3), where u_1, u_2, and u_3 appear in this counter-clockwise order along the cycle, and a Schnyder wood $\mathcal{W} = (\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3)$ of G, where $\mathcal{T}_i$ contains u_i, for $i = 1, 2, 3$.
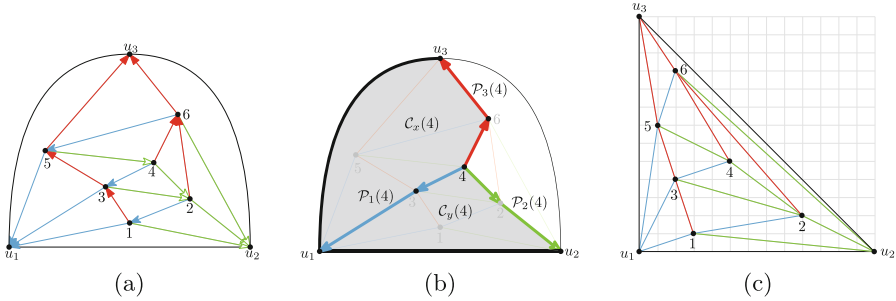
Fig. 6. (a) A Schnyder wood $\mathcal{W}$ of a maximal plane graph G. (b) Paths $\mathcal{P}_1(4)$, $\mathcal{P}_2(4)$, and $\mathcal{P}_3(4)$, and cycles $\mathcal{C}_x(4)$ and $\mathcal{C}_y(4)$. (c) The Schnyder drawing $s(\mathcal{W})$.

The S-algorithm assigns coordinates $(0, 0)$, $(2n - 5, 0)$, and $(0, 2n - 5)$ to u_1, u_2, and u_3, respectively. For a cycle $\mathcal{C}$, let $\#_f(\mathcal{C})$ be the number of internal faces of G inside $\mathcal{C}$. For $i = 1, 2, 3$, properties (S-1) and (S-2) of $\mathcal{W}$ imply that $\mathcal{T}_i$ contains a directed path $\mathcal{P}_i(w)$ from any internal vertex w to u_i; see Fig. 6(b). Also, $\mathcal{P}_1(w)$, $\mathcal{P}_2(w)$, and $\mathcal{P}_3(w)$ only share w [52]. Let $\mathcal{C}_x(w)$ and $\mathcal{C}_y(w)$ be the cycles $\mathcal{P}_1(w) \cup \mathcal{P}_3(w) \cup (u_1, u_3)$ and $\mathcal{P}_1(w) \cup \mathcal{P}_2(w) \cup (u_1, u_2)$, respectively. Then the algorithm assigns coordinates $(\#_f(\mathcal{C}_x(w)), \#_f(\mathcal{C}_y(w)))$ to w; see Fig. 6(c).

The following is the main tool for our enumeration algorithm.

Theorem 8. *Let G be an n-vertex maximal plane graph. There is a bijective function s from the Schnyder woods of G to the Schnyder drawings of G. Also, given a Schnyder wood of G, the corresponding Schnyder drawing of G can be constructed in $\mathcal{O}(n)$ time.*

Proof (sketch): The function s is the S-algorithm, which can be implemented in $\mathcal{O}(n)$ time [52], from which the second part of the statement follows. The core of the proof that s is bijective consists of proving that a Schnyder drawing Γ uniquely determines the Schnyder wood $\mathcal{W} = (\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3)$ such that $s(\mathcal{W}) = \Gamma$. This follows from the fact that in Γ, for each vertex v of G, the edges of $\mathcal{T}_1$, $\mathcal{T}_2$, and $\mathcal{T}_3$ incoming into v have slopes in the intervals $(0°, 90°)$, $(135°, 180°)$, and $(270°, 315°)$, respectively, while the edges of $\mathcal{T}_1$, $\mathcal{T}_2$, and $\mathcal{T}_3$ outgoing from v have slopes in the intervals $(180°, 270°)$, $(315°, 360°)$, and $(90°, 135°)$, respectively; see [27]. Thus, whether each edge (u, v) of G belongs to $\mathcal{T}_1$, $\mathcal{T}_2$, or $\mathcal{T}_3$ and whether it is directed from u to v or vice versa is uniquely determined by its slope in Γ. $\square$

Theorem 7 and Theorem 8 imply the following.

Theorem 9. *There is an algorithm with $\mathcal{O}(n)$ setup time and space usage listing all Schnyder drawings of an n-vertex maximal plane graph with $\mathcal{O}(n)$ delay.*

References

1. Alam, M.J., Biedl, T., Felsner, S., Kaufmann, M., Kobourov, S., Ueckerdt, T.: Computing cartograms with optimal complexity. Discrete Comput. Geom. **50**(3), 784–810 (2013). https://doi.org/10.1007/s00454-013-9521-1

2. Angelini, P., Chaplick, S., Cornelsen, S., Da Lozzo, G., Roselli, V.: Morphing triangle contact representations of triangulations. Discrete Comput. Geom. **70**, 991–1024 (2023). https://doi.org/10.1007/s00454-022-00475-9

3. Avis, D., Fukuda, K.: Reverse search for enumeration. Discrete Appl. Math. **65**(1–3), 21–46 (1996)

4. Bárány, I., Rote, G.: Strictly convex drawings of planar graphs. Documenta Math. **11**, 369–391 (2006)

5. Barbay, J., Aleardi, L.C., He, M., Munro, J.I.: Succinct representation of labeled graphs. Algorithmica **62**(1–2), 224–257 (2012). https://doi.org/10.1007/s00453-010-9452-7

6. Barbosa, V.C., Szwarcfiter, J.L.: Generating all the acyclic orientations of an undirected graph. Inf. Process. Lett. **72**(1–2), 71–74 (1999)

7. Been, K., Daiches, E., Yap, C.: Dynamic map labeling. IEEE Trans. Vis. Comput. Graph. **12**(5), 773–780 (2006)

8. Biedl, T., Marks, J., Ryall, K., Whitesides, S.: Graph multidrawing: finding nice drawings without defining nice. In: Whitesides, S.H. (ed.) GD 1998. LNCS, vol. 1547, pp. 347–355. Springer, Heidelberg (1998). https://doi.org/10.1007/3-540-37623-2_26

9. Birkhoff, G.: Rings of sets. Duke Math. J. **3**(3), 443–454 (1937)

10. Blind, S., Knauer, K., Valicov, P.: Enumerating k-arc-connected orientations. Algorithmica **82**(12), 3588–3603 (2020). https://doi.org/10.1007/s00453-020-00738-y

11. Bonichon, N., Gavoille, C., Hanusse, N., Poulalhon, D., Schaeffer, G.: Planar graphs, via well-orderly maps and trees. Graphs Comb. **22**(2), 185–202 (2006). https://doi.org/10.1007/s00373-006-0647-2

12. Bose, P., Dujmovic, V., Hurtado, F., Langerman, S., Morin, P., Wood, D.R.: A polynomial bound for untangling geometric planar graphs. Discrete Comput. Geom. **42**(4), 570–585 (2009). https://doi.org/10.1007/s00454-008-9125-3

13. Bose, P., Gudmundsson, J., Smid, M.H.M.: Constructing plane spanners of bounded degree and low weight. Algorithmica **42**(3–4), 249–264 (2005). https://doi.org/10.1007/s00453-005-1168-8

14. Brehm, E.: 3-orientations and Schnyder 3-tree-decompositions. Master's thesis, Freie Universität Berlin (2000)

15. Cantarella, J., Kusner, R.B., Sullivan, J.M.: On the minimum ropelength of knots and links. Invent. Math. **150**, 257–286 (2002). https://doi.org/10.1007/s00222-002-0234-y

16. Castelli Aleardi, L.: Algorithms for graphs on surfaces: from graph drawing to graph encoding. Habilitation thesis, Université de Paris (2021)

17. Chrobak, M., Payne, T.H.: A linear-time algorithm for drawing a planar graph on a grid. Inf. Process. Lett. **54**(4), 241–246 (1995)

18. Chuang, R.C., Garg, A., He, X., Kao, M., Lu, H.: Compact encodings of planar graphs via canonical orderings and multiple parentheses. In: Larsen, K.G., Skyum, S., Winskel, G. (eds.) ICALP 1998. LNCS, vol. 1443, pp. 118–129. Springer, Heidelberg (1998). https://doi.org/10.1007/BFb0055046

19. Conte, A., Grossi, R., Marino, A., Rizzi, R.: Efficient enumeration of graph orientations with sources. Discrete Appl. Math. **246**, 22–37 (2018)

20. Da Lozzo, G., D'Angelo, A., Frati, F.: On the area requirements of planar greedy drawings of triconnected planar graphs. In: Kim, D., Uma, R., Cai, Z., Lee, D. (eds.) COCOON 2020. LNCS, vol. 12273, pp. 435–447. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58150-3_35

21. Da Lozzo, G., Di Battista, G., Frati, F., Grosso, F., Patrignani, M.: Efficient enumeration of drawings and combinatorial structures for maximal planar graphs. CoRR, abs/2310.02247 (2023)

22. de Fraysseix, H., de Mendez, P.O.: On topological aspects of orientations. Discrete Math. **229**(1–3), 57–72 (2001)

23. de Fraysseix, H., de Mendez, P.O., Rosenstiehl, P.: On triangle contact graphs. Comb. Probab. Comput. **3**, 233–246 (1994)

24. de Fraysseix, H., de Mendez, P.O., Rosenstiehl, P.: Bipolar orientations revisited. Discrete Appl. Math. **56**(2–3), 157–179 (1995)

25. de Fraysseix, H., Pach, J., Pollack, R.: Small sets supporting Fáry embeddings of planar graphs. In: Simon, J. (ed.) STOC 1998, pp. 426–433 (1988)

26. de Fraysseix, H., Pach, J., Pollack, R.: How to draw a planar graph on a grid. Combinatorica **10**(1), 41–51 (1990). https://doi.org/10.1007/BF02122694

27. Dhandapani, R.: Greedy drawings of triangulations. Discrete Comput. Geom. **43**(2), 375–392 (2010). https://doi.org/10.1007/s00454-009-9235-6

28. Di Battista, G., Eades, P., Tamassia, R., Tollis, I.G.: Graph drawing: algorithms for the visualization of graphs (1999)

29. Di Battista, G., Tamassia, R., Vismara, L.: Output-sensitive reporting of disjoint paths. Algorithmica **23**(4), 302–340 (1999). https://doi.org/10.1007/PL00009264

30. Dujmovic, V., Eppstein, D., Suderman, M., Wood, D.R.: Drawings of planar graphs with few slopes and segments. Comput. Geom. **38**(3), 194–212 (2007)

31. Felsner, S.: Convex drawings of planar graphs and the order dimension of 3-polytopes. Order **18**(1), 19–37 (2001)

32. Felsner, S.: Lattice structures from planar graphs. Electron. J. Comb. **11**(1), R15 (2004)

33. Felsner, S., Zickfeld, F.: Schnyder woods and orthogonal surfaces. Discrete Comput. Geom. **40**(1), 103–126 (2008). https://doi.org/10.1007/s00454-007-9027-9

34. Ganian, R., et al.: Are there any good digraph width measures? J. Comb. Theory Ser. B **116**, 250–286 (2016)

35. Habib, M., Medina, R., Nourine, L., Steiner, G.: Efficient algorithms on distributive lattices. Discrete Appl. Math. **110**(2–3), 169–187 (2001)

36. Hass, J., Lagarias, J.C.: The number of Reidemeister moves needed for unknotting. J. Am. Math. Soc. **14**, 399–428 (2001)

37. Hass, J., Lagarias, J.C., Pippenger, N.: The computational complexity of knot and link problems. J. ACM **46**(2), 185–211 (1999)

38. Hayes, T.P.: A simple condition implying rapid mixing of single-site dynamics on spin systems. In: FOCS 2006, pp. 39–46 (2006)

39. He, X., Kao, M., Lu, H.: Linear-time succinct encodings of planar graphs via canonical orderings. SIAM J. Discrete Math. **12**(3), 317–325 (1999)

40. Kahn, A.: Topological sorting of large networks. Commun. ACM **5**(11), 558–562 (1962)

41. Kant, G.: Drawing planar graphs using the canonical ordering. Algorithmica **16**(1), 4–32 (1996). https://doi.org/10.1007/BF02086606

42. Knuth, D.E.: The Art of Computer Programming, Volume 4A: Combinatorial Algorithms, Part 1 (2011)

43. Kobourov, S.G.: Canonical orders and Schnyder realizers. In: Encyclopedia of Algorithms, pp. 277–283 (2016)

44. Miura, K., Azuma, M., Nishizeki, T.: Canonical decomposition, realizer, Schnyder labeling and orderly spanning trees of plane graphs. Int. J. Found. Comput. Sci. **16**(1), 117–141 (2005)
45. Nishizeki, T., Rahman, M.S.: Planar Graph Drawing, Volume 12 of Lecture Notes Series on Computing (2004)
46. Nöllenburg, M., Prutkin, R., Rutter, I.: On self-approaching and increasing-chord drawings of 3-connected planar graphs. J. Comput. Geom. **7**(1), 47–69 (2016)
47. Ono, A., Nakano, S.: Constant time generation of linear extensions. In: Liskiewicz, M., Reischuk, R. (eds.) FCT 2005. LNCS, vol. 3623, pp. 445–453. Springer, Heidelberg (2005). https://doi.org/10.1007/11537311_39
48. Pruesse, G., Ruskey, F.: Gray codes from antimatroids. Order **10**(3), 239–252 (1993). https://doi.org/10.1007/BF01110545
49. Pruesse, G., Ruskey, F.: Generating linear extensions fast. SIAM J. Comput. **23**(2), 373–386 (1994)
50. Ruskey, F.: Combinatorial generation. University of Victoria, Victoria BC, Canada (2003). Preliminary working draft
51. Schaefer, M., Stefankovic, D.: Decidability of string graphs. J. Comput. Syst. Sci. **68**(2), 319–334 (2004)
52. Schnyder, W.: Embedding planar graphs on the grid. In: Johnson, D.S. (ed.) SODA 1990, pp. 138–148 (1990)
53. Setiawan, A., Nakano, S.-I.: Listing all *st*-orientations. IEICE Trans. Fund. Electr. Comm. Comp. Sci. **94**(10), 1965–1970 (2011)
54. Squire, M.B.: Gray codes and efficient generation of combinatorial structures. Ph.D. thesis, North Carolina State University (1995)
55. Squire, M.B.: Generating the acyclic orientations of a graph. J. Algorithms **26**(2), 275–290 (1998)
56. Steiner, G.: An algorithm to generate the ideals of a partial order. Oper. Res. Lett. **5**(6), 317–320 (1986)
57. Tamassia, R. (ed.): Handbook on Graph Drawing and Visualization (2013)
58. Wasa, K.: Enumeration of enumeration algorithms. CoRR, abs/1605.05102 (2016)