Two-page Book Embedding and Clustered Graph Planarity

Seek-Hee Hong[1] Hiroshi Nagamochi[2]

[1]School of Information Technologies, University of Sydney, shhong@it.usyd.edu.au
[2]Department of Applied Mathematics and Physics, Kyoto University, nag@amp.i.kyoto-u.ac.jp

Abstract: A *2-page book embedding* of a graph places the vertices linearly on a spine (a line segment) and the edges on two pages (two half planes sharing the spine) so that each edge is embedded in one of the pages without edge crossings. Testing whether a given graph admits a 2-page book embedding is known to be NP-complete.

In this paper, we study the problem of testing whether a given graph admits a 2-page book embedding with a fixed edge partition. Based on structural properties of planar graphs, we prove that the problem of testing and finding a 2-page book embedding of a graph with a partitioned edge set can be solved in linear time.

As an application of our main result, we consider the problem of testing planarity of clustered graphs. The complexity of testing clustered graph planarity is a long standing open problem in Graph Drawing. Recently, polynomial time algorithms have been found for several classes of clustered graphs in which both the structure of the underlying graphs and clustering structure are restricted. However, when the underlying graph is disconnected, the problem remains open. Our book embedding results imply that the clustered planarity problem can be solved in linear time for a certain class of clustered graphs with arbitrary underlying graphs (i.e. possibly disconnected).

1 Introduction

For an integer $k \geq 1$, a k-page book embedding (or a k-stack layout) of a graph is to place the vertices linearly on a spine (a line segment) and the edges on k pages (k half planes sharing the spine) so that each edge is embedded in one of the pages without generating edge-crossings. The book embedding problem has applications in the routing of multilayer printed circuit boards and in the design of fault-tolerant processor arrays [2, 17]. See [10] for numerous applications of book embeddings.

Graphs with 1-page book embeddings are the outerplanar graphs. Yannakakis proved that every planar graph admits a 4-page book embedding [19]. Bernhart and Kainen [1] show that a planar graph has a 2-page book embedding if and only if it is sub-Hamiltonian. A planar graph is sub-Hamiltonian if and only if it is Hamiltonian or can be made Hamiltonian by inserting additional edges without violating planarity. The problem of testing sub-Hamiltonicity is NP-complete [20]. Hence the problem of determining whether a given planar graph $G = (V, E)$ is a 2-page book embedding or not is NP-complete. See [10, 11] for a survey on book embeddings and graph linear layouts.

The 2-page book embedding problem contains two combinatorial aspects. One is how to partition an edge set in two edge subsets, each corresponds to one of the two sides along the spine. The other is how to decide an ordering of the vertices on the spine. Note that if an ordering π of all the vertices along the spine is fixed, then we can test whether a given graph admits a 2-page book embedding with π or not in linear time; the problem can be converted into a planarity testing problem by adding edges between every two consecutive vertices in π (where the last vertex is

[1]Technical report 2009-004, January 22, 2009.

connected by the first one). However, it is not known whether the problem remains NP-complete or can be solved in polynomial time if a partition of the edge set is prescribed.

In this paper, we consider the problem of testing whether a given graph admits a 2-page book embedding for a fixed edge partition. Based on structural properties of biconnected planar graphs, we show that the problem of finding a 2-page book embedding of a graph with a partitioned edge set can be solved in linear time.

As an application of our main result, we consider the clustered graph planarity testing problem of flat clustered graphs in which the root cluster has exactly two child clusters. The computational complexity status of the c-planarity testing problem is a long standing open problem, and polynomial time algorithms are known to several classes of clustered graphs for which the structures on the underlying graphs and cluster structures are both restricted. However, even for a "flat" clustered graph (a clustered graph in which all internal clusters are the children of the root cluster) with exactly two internal clusters, which is the simplest nontrivial case of cluster structures, the complexity status is not known [4]. We prove that this fundamental case of the c-planarity testing problem with arbitrary underlying graphs (possibly disconnected) can be solved in linear time by reducing the problem to the 2-page book embedding problem with partitioned edge sets.

The paper is organized as follows. After reviewing basic terminologies on plane graphs and the triconnected component decomposition of biconnected graphs in Section 2, we define an obstacle for a planar graph to admit a 2-page book embedding, called "splitter," and characterize 2-page book embeddings as "splitter-free" and "disjunctive" plane embeddings in Section 3. Then we show how to check whether a given graph G has a special type of splitters, "rigid splitter" or not in Section 4. In Section 5, we show that whether G has a "splitter-free" plane embedding or not can be tested in linear time. Then we prove that whether G has a "disjunctive" plane embedding or not can be tested in linear time in Section 6. Section 7 presents a linear time algorithm for testing whether G has a "splitter-free" and "disjunctive" plane embedding or not; i.e., G has a 2-page book embedding. Finally Section 8 shows that for a new class of clustered graphs, the c-planarity can be tested in linear time using our algorithm for the 2-page book embedding problem.

2 Preliminaries

Let $G = (V, E)$ be a graph. The set of edges incident to a vertex $v \in V$ is denoted by $E(v; G)$. A path with end vertices u and v is called a *u, v-path*. The degree of a vertex v in G is denoted by $deg(v; G)$. For a subset $X \subseteq E$ (resp., $X \subseteq V$), $G - X$ denotes the graph obtained from G by removing the edges in X (resp., the vertices in X together with the edges in $\cup_{v \in X} E(v; G)$). A planar graph $G = (V, E)$ with a fixed embedding F of G is called a *plane* graph. The set of vertices, set of edges and set of facial cycles of a plane graph H may be denoted by $V(H)$, $E(H)$ and $F(H)$, respectively.

2.1 SPQR tree representations

To consider the all possible plane combinatorial embeddings of a biconnected planar graph $G = (V, E_1 \cup E_2)$, we use the SPQR tree $\mathcal{T}$ of G by di Battista and Tamassia [8, 9] (see Appendix 1 for detail). Throughout the paper, we assume that G is not a simple cycle, since otherwise G always has a 2-page book embedding.

The SPQR tree of a biconnected graph $G = (V, E)$ represents the adjacency of the triconnected components of G. Each node ν in the SPQR tree is associated with G called the *skeleton* of ν, denoted by $\sigma(\nu) = (V_\nu, E_\nu)$ ($V_\nu \subseteq V$), which corresponds to a triconnected component. Since G is not a simple cycle, the SPQR tree of G has an R- or P-node. We treat the SPQR tree of a graph G as a rooted tree $\mathcal{T}$ by choosing an R- or P-node ν^* as its root. For a node ν, let $Ch(\nu)$ denote the set of all children of ν. We denote the graph formed from $\sigma(\nu)$ by deleting its *parent virtual edge* as $\sigma^-(\nu)$, if ν is not the root of $\mathcal{T}$. Let $G^-(\nu)$ denote the subgraph induced from G by the set of all vertices in the graphs $\sigma^-(\mu)$ for all descendants μ of ν, including ν itself. For a notational convenience, we use $G^-(\nu^*)$ and $\sigma^-(\nu^*)$ to denote $G(\nu^*)$ and $\sigma(\nu^*)$ for the root ν^*. For each nonroot node ν, we denote by $G^+(\nu)$ the graph $G - (V(G^-(\mu)) - \{u, v\})$. For a node ν and a virtual edge $e \in E(\sigma(\nu))$, we let G_e denote the graph $G^-(\mu_e)$ for the corresponding child node $\mu_e \in Ch(\nu)$ if $e \in E(\sigma^-(\nu))$, and G_e denote the graph $G^+(\mu_e)$ if e is the parent virtual edge of ν. For a real edge $e \in E(\sigma(\nu))$, G_e denote the graph consisting of the edge e.

In this paper, for each nonroot node ν, an embedding of a graph $\sigma^-(\nu)$ or $G^-(\nu)$ means a plane embedding of the graph such that the both end vertices u and v of the parent virtual edge of ν appear in the boundary of the plane embedding. For an embedding ψ_ν of the graph $G^-(\nu)$ for a nonroot node ν, a plane embedding ψ_G of G is called an *extension* from ψ_ν if ψ_ν appears as part of the embedding ψ_G.

For a nonroot R-node ν, we let the plane graph $\sigma^-(\nu)$ mean the plane embedding of $\sigma^-(\nu)$ such that the two end vertices of the parent virtual edge of ν appear on the boundary of the embedding (note that such an embedding is unique). For the root R-node ν^*, the plane graph $\sigma^-(\nu^*)$ means a plane embedding of the triconnected planar graph $\sigma(\nu^*)$.

3 Characterization of Two-page Book Embeddings

In this section, we let $G = (V, E_1 \cup E_2)$ denote a planar graph with a partition E_1 and E_2 of $E(G)$, where two vertices may be joined by two edges $e \in E_1$ and $e' \in E_2$; Each E_i induces a simple and outerplanar subgraph (V, E_i).

We call the edges in E_1 (resp., E_2) *red edges* (resp., *blue edges*). A subgraph H in G is called *red* (resp., *blue*) if $E(H)$ consists of only red (resp., blue) edges. A vertex to which only red (resp., blue) edges are incident is called an *r-vertex* (resp., *b-vertex*). A vertex to which both red and blue edges are incident is called a *br-vertex*.

A *2-page book embedding* π of a graph $G = (V, E_1 \cup E_2)$ is a linear ordering of the vertices such that all vertices are placed in this order on a spine and all red edges are drawn above the spine and all blue edges are drawn below the spine without any edge-crossings.

3.1 Characterizing 2-page book embeddings via plane embeddings

In a 2-page book embedding π of a graph $G = (V, E_1 \cup E_2)$, if we join the first and last vertices on the spine with a new curve so that the spine together with the curve forms a simple closed curve which encloses all red edges but no blue edges (see Fig. 1(a)). Thus, a 2-page book embedding π can be regarded as a plane embedding ψ of G in which a simple closed curve λ visits each vertex exactly once without intersecting any edge and encloses all red edges but no blue edges. We call such a curve λ a *separating curve* of the embedding ψ. Note that the first and last vertices appear on the outer facial cycle in the plane embedding ψ. However by rechoosing the outer face, any vertex v can appear along the outer facial cycle. This does not change the combinatorial embedding, and thereby the vertex v can appear as the first vertex on the spine in the 2-page book embedding obtained from the resulting plane embedding ψ'. See Fig. 1(b), where the vertices v_3 and v_2 are chosen as the first and last vertices on the spine.

Lemma 1 *A planar graph $G = (V, E_1 \cup E_2)$ admits a 2-page book embedding π with partition E_1 and E_2 if and only if each block H of G has a 2-page book embedding π_H for the partition $E(H) \cap E_1$ and $E(H) \cap E_2$ of $E(H)$.*

Proof. Since the only if part is trivial, we consider the if part. Let $H_1, H_2, \ldots, H_p$ be the blocks of G indexed so that each H_i, $i = 2, 3, \ldots, p$ has a common vertex v_{H_i} with some block H_j with $j < i$, where such v_{H_i} is unique to H_i and is called the parent of H_i. Assume that each block H_i of G has a 2-page book embedding π_{H_i} with the partition $E(H_i) \cap E_1$ and $E(H_i) \cap E_2$. As observed above, we can regard each embedding π_{H_i} as a plane embedding of H_i, and by choosing the outer face so that the parent v_{H_i} appears on the boundary, we can obtain a 2-page book embedding π'_{H_i} of H_i in which v_{H_i} appears as the first vertex on the spine. Let $\pi := \pi'_{H_1}$. For each $i = 2, 3, \ldots, p$, we place the 2-page book embedding π'_{H_i} in the space between vertex v_{H_i} and the vertex next to v_{H_i} in the current 2-page book embedding π, letting π'_{H_i} share the vertex v_{H_i} with π. In this way, we can construct a desired 2-page book embedding π of G. □

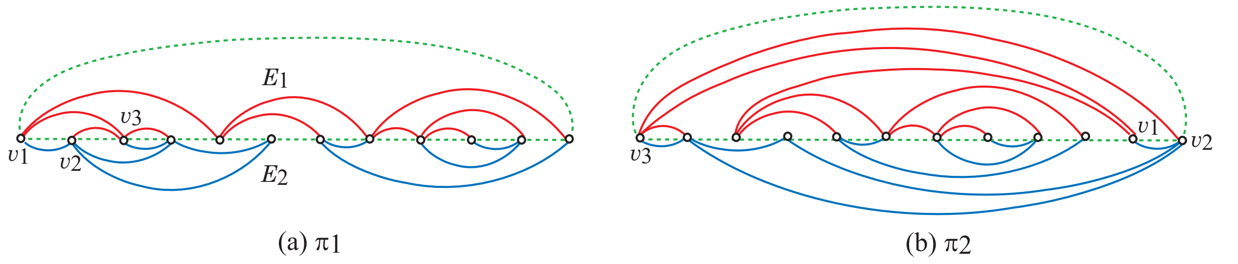

Figure 1: (a) A 2-page book embedding π_1 of a planar graph $G = (V, E_1 \cup E_2)$; (b) a 2-page book embedding π_2 obtained from π_1 by choosing the vertices v_3 and v_2 as the first and last vertices on the spine.

If $E_1 = \emptyset$ or $E_2 = \emptyset$, then any planar graph G has a 2-page book embedding if and only if G is outerplanar. In what follows, we assume that $G = (V, E_1 \cup E_2)$ is a biconnected planar graph with $E_1 \neq \emptyset$ and $E_2 \neq \emptyset$.

In a plane embedding ψ of G, a red (resp., blue) cycle Q of G is called a *splitter* if each of the two regions obtained by the cycle Q contains a vertex $v \in V - V(Q)$ or a blue (resp., red) edge.

In ψ, a vertex v is called *disjunctive* if for each $i = 1, 2$, all the edges in $E_i(v; G)$ appear consecutively around v. We call ψ *disjunctive* if all vertices in V are disjunctive.

Theorem 2 *Let $G = (V, E_1 \cup E_2)$ be a biconnected planar graph with $E_1 \neq \emptyset \neq E_2$. Then G admits a 2-page book embedding π if and only if G admits a disjunctive and splitter-free plane embedding ψ. Moreover, a 2-page book embedding π of G can be obtained from a disjunctive and splitter-free plane embedding ψ of G in linear time.*

Proof. As observed above, a 2-page book embedding π of G can be regarded as a plane embedding which admits a separating curve λ, where we treat λ as an oriented curve so that the red edges appear on our left hand side when we traverse λ along its orientation. Conversely, if a plane embedding admits such a separating curve λ, then we can obtain a 2-page book embedding π of G.

Only if part: Assume that G has a 2-page book embedding π. Let λ be a separating curve of π. The curve λ visits each vertex exactly once and it encloses only red edges. Hence the plane embedding π is disjunctive at any vertex. If the plane embedding π has a red splitter, then λ cannot visit a vertex (or a blue edge) enclosed by the splitter and a vertex (or a blue edge) outside the splitter, because λ cannot intersect the splitter due to the disjunctive condition on the vertices in the splitter. Hence π has no red splitter. Similarly, π has no blue splitter.

If part: Assume that G has a disjunctive and splitter-free plane embedding ψ. It suffices to show that ψ admits a separating curve λ. For this, we construct an Eulerian plane digraph G^* and its Eulerian trail λ by the following procedure.

Step 1. Place a new vertex v_f inside each face f of ψ.

Step 2. For each br-vertex $v \in V$, there exists a unique pair of faces f and f' sharing the vertex v such that all the red (resp., blue) edges incident to v appear in the clockwise order after visiting face f (resp., f') around v, because ψ is disjunctive. We join such two vertices v_f and $v_{f'}$ via a new directed edge (v_f, v) and $(v, v_{f'})$.

Step 3. For each b-vertex $u \in V$, there exists a face f_u containing at least one red edge, because otherwise the union of the faces that contain v would include a blue splitter Q that separates v from a vertex or a red edge outside the region Q. We add new directed edges (v_{f_u}, u) and (u, v_{f_u}) in the clockwise order around u.

Step 4. For each r-vertex $u \in V$, we also introduce new directed edges symmetrically with Step 3.

Step 5. We call all the new edges introduced in Steps 2-4 *green edges*. Let $G^* = (V \cup V', E^g)$ be the Eulerian plane bipartite digraph that consists of all the green edges and their end vertices, where V' denotes the set of the end vertices v_f of all green edges; G^* is a plane, bipartite and Eulerian digraph which has no isolated vertices (see Fig. 2(a)).

Step 6. Convert G^* into a simple directed cycle λ consisting of all the green edges without self-intersection in the plane, by splitting each vertex $v_f \in V'$ into $deg(v_f; G^*)/2$ vertices of indegree 1 and outdegree 1. Such a cycle λ visits each vertex $v \in V$ exactly once and encloses all the red edges but no blue edges, i.e., a separating cycle of ψ.

To complete the proof, it suffices to show that Step 6 can be executed. For this, we first show that G^* is connected. If G^* has two components H and H' (as shown in Fig. 2(a)), then one of H and H' would be enclosed by a red or blue cycle Q of G, which separates a vertex $v \in V \cap V(H)$ and a vertex $v' \in V \cap V(H')$, indicating that Q is a splitter of ψ, a contradiction. Hence G^* is connected (see Fig. 2(b)). Since G^* is a connected Eulerian digraph, it has an Eulerian trail (a simple directed cycle consisting of all the green edges). We show that there is an Eulerian trail which has no self-intersection in the plane, i.e., it plays a role as a separating curve.

By the construction of the plane digraph G^*, for each vertex $v_f \in V'$, the incoming and outgoing green edges incident to v_f appear alternately around v_f. Hence we can traverse the boundary of G^* following the edge directions. Based on this observation, we partition the set E^g of green edges into $E^g_1, E^g_2, \ldots, E^g_p$ as follows. Let E^g_1 be the set of the edges in the boundary, and $G^*_1 = G^* - E^g_1$. Analogously we can traverse the boundary of each component of G^*_1 following the edge directions. We repeatedly define E^g_i as the set of edges in the boundaries of the components of $G^* - (E^g_1 \cup \cdots \cup E^g_{i-1})$ until $G^* - (E^g_1 \cup \cdots \cup E^g_i)$ has no edge for some $i = p$. Each set E^g_i gives a collection of Eulerian trail with no self-intersection. The set of all these trails forms a rooted tree structure in which the trail in E^g_1 is the root and a trail in E^g_i is a child of a trail E^g_{i-1} if these trails share a vertex v_f. We can combine all these trails into a single Eulerian trail λ with no self-intersection by traversing the trails in a DFS manner in the tree structure. This proves the if part.

It is not difficult to see that the above procedure can be implemented to run in linear time to obtain a separating curve λ, from which a 2-page book embedding of G can be obtained in linear time. This proves the second statement in the theorem. □

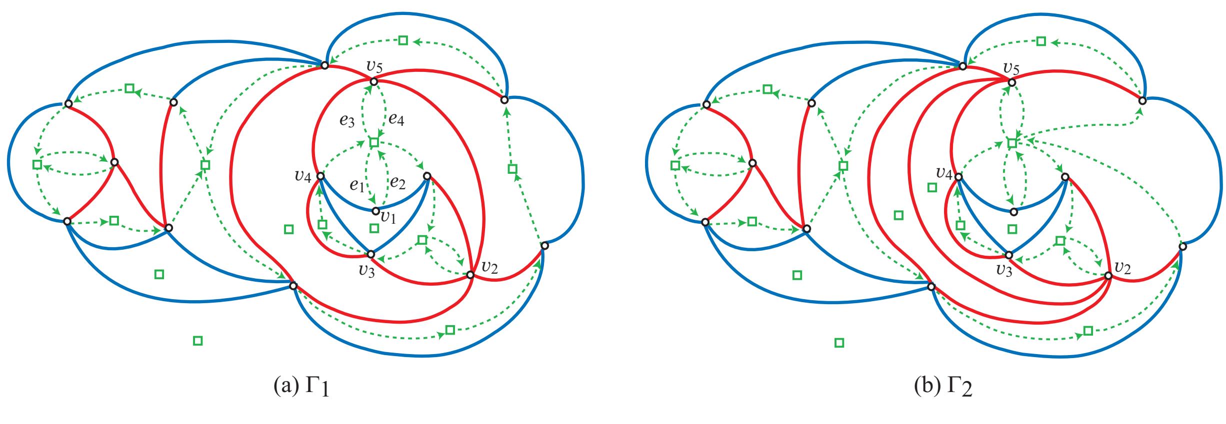

Figure 2: Two embeddings of a planar graph $G = (V, E_1 \cup E_2)$; (a) a plane embedding ψ_1 of G which has no separating curve λ; (b) A disjunctive and splitter-free plane embedding ψ_2 of G, where the circles represent the vertices in G, the squares represent the new vertices v_f for the faces $f \in F(G)$, and the dashed arrows represent the green edges defined in the proof of Lemma 2.

Fig. 3(a) and (b) show an example of a graph G_1 which has a splitter-free embedding ψ_1 and a disjunctive embedding ψ_2, but has no disjunctive and splitter-free embedding.

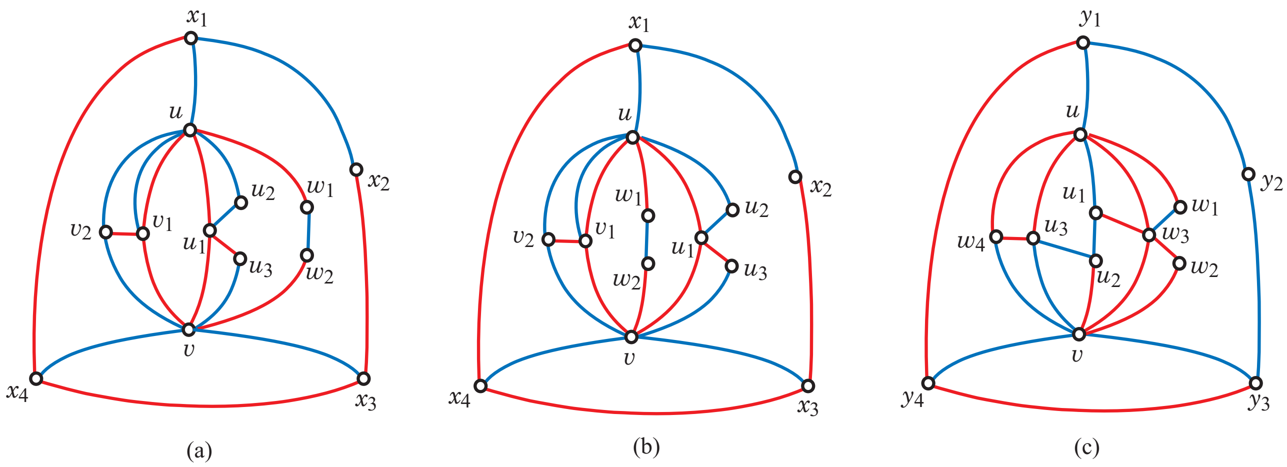

Figure 3: (a) A splitter-free embedding ψ_1 of the graph G_1, which has the P-node ν_c with $V(\sigma^-(\nu_c)) = \{u, v, u_1, u_2, u_3, v_1, v_2, w_1, w_2\}$; (b) a disjunctive embedding ψ_2 of the same graph G_1 in (a), where u is not disjunctive in ψ_1 and the red cycle $Q = (u, u_1, v, v_1)$ is a splitter in ψ_2; (c) a plane embedding of graph G_2, where the blue cycle $(u, u_1, u_2, u_3, v, y_3, y_2, y_1)$ is a blue rigid splitter at the R-node ν_c with $V(\sigma^-(\nu_c)) = \{u, v, u_1, u_2, u_3, w_1, w_2, w_3, w_4\}$.

4 Detecting Rigid Splitters

This section considers whether G has a special type of splitter, called "rigid splitters."

Let $e = (u, v)$ be a virtual edge of the skeleton of a node ν, and ν_e be the node corresponding to the virtual edge e. The graph $G_e = G^-(\nu_e)$ is called a *br-graph* if it has both red and blue u, v-paths. The graph $G_e = G^-(\nu_e)$ is called an *r-graph* (resp., *b-graph*) if it has a red (resp., blue) u, v-path, but it has no blue (resp., red) u, v-path. A virtual edge e is called an *r-edge* (resp., *b-edge* and *br-edge*) if $G^-(\nu_e)$ is an *r-graph* (resp., *b-graph* and *br-graph*). We also treat a red (resp., blue) real edge as an r-edge (resp., b-edge). Note that we can compute the types of the virtual edges in all nodes in the SPQR tree $\mathcal{T}$ in $O(|V|)$ time by traversing $\mathcal{T}$ in a bottom-up manner.

Let Q be a cycle of G, and ν be a node of $\mathcal{T}$. An edge $e \in E(\sigma(\nu))$ is called *Q-disjoint* if the graph G_e contains no edge in Q, where e may be a real edge or the parent virtual edge of ν. For a Q-disjoint virtual edge $e \in E(\sigma(\nu))$, a vertex $v \in V(G_e) - V(Q)$ is called *Q-disjoint*.

A node ν of $\mathcal{T}$ is *Q-intersecting* if there is no virtual edge $e \in E(\sigma(\nu))$ such that Q is contained in the graph G_e; i.e., $\sigma(\nu)$ contains a cycle Q' which corresponds to Q.

A red cycle Q is called a *rigid* splitter at an R-node ν if

(i) ν is Q-intersecting; and

(ii) For any embedding of G, each of the two regions obtained by Q contains a blue real edge from G_e or a vertex from $G_e - Q$ for some Q-disjoint edge $e \in E(\sigma(\nu))$.

We define rigid blue splitters symmetrically exchanging the role of red and blue. For example, the blue cycle $(u, u_1, u_2, u_3, v, y_3, y_2, y_1)$ in Fig. 3(c) is a blue rigid splitter at the R-node ν_c with $V(\sigma^-(\nu_c))$ (see Fig. 12).

Note that a rigid splitter Q in a plane embedding of G remains a splitter in any other plane embedding of G, whereas a non-rigid splitter in a plane embedding of G may not be a splitter in some other plane embedding of G. For example, the red splitter $Q = (u, u_1, v, v_1)$ in the plane embedding ψ_2 in Fig. 3(b) is no longer a splitter in the plane embedding ψ_1 in Fig. 3(a). The next subsection shows how to check whether a given graph G contains a rigid splitter.

A nonroot node ν is called *r-joined* (resp., *b-joined*) if $G^+(\nu)$ has a red (resp., blue) u, v-path. An r-joined and b-joined node is called *br-joined*. The parent virtual edge of a nonroot node ν is treated as an r-edge (resp., b- and br-edges) if ν is r-joined (resp., b- and br-joined). We show how to compute the r- and b-joinedness after describing how to check the existence of rigid splitters (see Lemmas 4 and 5).

Let ν be an R-node of $\mathcal{T}$. To detect a rigid red splitter Q at the R-node ν, we construct a plane graph $H|^r$ from $H = \sigma(\nu)$ so that a rigid red splitter Q at ν corresponds to a red splitter in the graph $H|^r$. The plane graph $H|^r = (V(H) \cup W, E'_1 \cup E'_2)$ is obtained from $H = \sigma(\nu)$ by subdividing each virtual or blue real edge $e \in E(H)$ into two edges as follows, where W denotes the set of the vertices introduced to subdivide virtual/blue real edges, and E'_1 (resp., E'_2) denotes the set of all red (resp., blue) edges in $H|^r$.

(i) If e is an r- or br-edge (where e is not a red real edge), then subdivide the edge $e = (u, v)$ into two red edges (u, w_e) and (w_e, v) with a new vertex w_e.

(ii) If e is neither an r- nor br-edge (where e is possibly a blue real edge), then subdivide the edge $e = (u, v)$ into two blue edges (u, w_e) and (w_e, v) with a new vertex w_e.

Note that a pair of red and blue real edges with the same end vertices is treated as a virtual edge, for which (i) is applied.

We claim the following property.

Lemma 3 *Graph $G = (V, E_1 \cup E_2)$ has a rigid red splitter Q at an R-node ν of $\mathcal{T}$ if and only if the $H|^r$ has a red splitter Q' separating some two vertices in $V(H) \cup W$. Moreover, whether the $H|^r$ has a red splitter or not can be tested in $O(|E(\sigma(\nu))|)$ time.*

Proof. Let ν be an R-node, and let $H = \sigma(\nu)$.

Only if part: Let Q be a rigid red splitter in at the R-node ν. Each virtual edge $e = (u, v) \in E(H)$ is an r- or br-edge if e is not Q-disjoint. Hence such a virtual edge is replaced with two red edges (u, w_e) and (w_e, v) in $H|^r$. Thus Q is replaced with a simple red cycle Q' in $H|^r$. Let R_1 and R_2 be the two regions obtained by Q, where each R_i contains a blue real edge $e \in E(H)$ or a Q-disjoint vertex v. We show that, in $H|^r$, each R_i still contains a vertex $v \in V(H) \cup W$. If R_i contains a blue real edge $e = (u, v)$ in H, then R_i contains the vertex $w_e \in W$ that subdivides the blue real edge e in $H|^r$ by the construction rule (ii). If R_i contains no blue real edges, then it contains a Q-disjoint vertex v. If $v \in V(H) - V(Q)$, then R_i contains the same vertex v in $H|^r$. If $v \in V(G_e) - V(Q)$ for a Q-disjoint virtual edge $e \in E(H)$, then R_i contains the vertex $w_e \in W$ that subdivides the edge e in $H|^r$ by the construction rules (i) and (ii). Therefore, Q' is a red splitter in $H|^r$.

If part: Assume that $H|^r$ has a red splitter Q'' separating some two vertices in $V(H) \cup W$. Let R_1 and R_2 be the two regions obtained by Q'', where each R_i contains a vertex in $V(H) \cup W - V(Q'')$. By construction of $H|^r$ from H, the graph G contains a red cycle Q corresponding to Q''; i.e., Q gives the same two regions R_1 and R_2. We show that Q is a rigid red splitter at ν. First consider the case where R_i contains a vertex u. If $u \in V(H) - V(Q'')$, then R_i still contains the same vertex $v \in V(H) - V(Q)$ in H. Let $u \in W - V(Q'')$. Then in H, R_i contains a virtual edge or a blue real edge $e \in E(H)$ that is subdivided by the vertex $v = w_e$. When e is a virtual edge, R_i contains a vertex $u \in V(G_e) - V(Q'')$ for the virtual edge e. Note that the virtual edge e is Q-disjoint, because otherwise, Q'' would pass through the vertex $v = w_e$. Therefore, Q is a rigid red splitter in H.

Finally we show how to test whether $H|^r$ has a red splitter in $O(|V(H)|)$ time. Since G contains at least one blue edge by assumption, at least one of the edges in H is a blue real edge or a virtual edge.

First consider the case where $H|^r$ has a vertex $u \in V(H|^r) = V(H) \cup W$ such that only red edges are incident to u and all faces containing u consists of red edges in $H|^r$. We claim that if such a vertex u exists then a red rigid splitter exists. Choose u so that $deg(u; H|^r)$ is maximized. Then the union of edges in the faces not incident to u forms a red cycle Q', which divides the plane into two regions R_1 and R_2, where R_1 contains u. It suffices to show that R_2 contains a vertex in $V(H) \cup W$, which implies that Q' corresponds to a rigid splitter at ν, and we are done. Assume that R contains no vertex in $V(H) \cup W$. Note that $deg(u; H|^r) = 2$ means that each neighbor u' of u in $H|^r$ to which now only red edges are incident should have been chosen as u since $deg(u'; H|^r) \geq 3$. Hence $deg(u; H|^r) \geq 3$. However, in this case, $V(Q') \cup \{u\}$ induces a subdivision of a wheel from $H|^r$, which means that E_1 induces a non-outerplanar subgraph from G, contradicting the assumption on G. This proves the claim. It is easy to see that testing if such a vertex u exists or not can be tested in $O(|V(H)|)$ time.

Next we assume that $H|^r$ contains no such vertex $u \in V(H|^r) = V(H) \cup W$. Hence $H|^r$ contains at least one blue edge in E'_2. Let $\mathcal{H}$ denote the plane dual graph of the plane graph $H|^r$, which establishes a one-to-one correspondence between the two edge sets of $\mathcal{H}$ and $H|^r$, where we preserve the color classes. Let $\mathcal{H}_b$ be the subgraph of $\mathcal{H}$ induced by the blue edges of $\mathcal{H}$. We claim that $H|^r$ has no red splitter if and only if $\mathcal{H}_b$ is connected. If $\mathcal{H}_b$ has two components H_1 and H_2, then the set of edges of $\mathcal{H}$ between $V(H_1)$ and $V(\mathcal{H}) - V(H_1)$ is a minimal cut, which corresponds to a simple cycle Q' in $H|^r$, where we can see that Q' is a red splitter of $H|^r$. Now consider the converse. Assume that $H|^r$ has a red splitter Q' separating two vertices $x_1, x_2 \in V(H) \cup W$ into two regions R_1 and R_2, respectively. It suffices to show that each R_i contains at least one blue edge in E'_2, which implies that $\mathcal{H}_b$ is not connected. This is trivial if a blue edge in E'_2 is incident to x_i. When only red edges in E'_1 are incident to x_i, vertex x_i is contained in a face f_{x_i} of $H|^r$ which contains a blue edge in E'_2 by the assumption on H. Since x_i and any edge in f_{x_i} cannot be separated by Q', region R_i also contain the blue edges in f_{x_i}. This proves that $H|^r$ has no red splitter if and only if $\mathcal{H}_b$ is connected. The connectivity of $\mathcal{H}_b$ can be checked in $O(|V(\mathcal{H})| + |E(\mathcal{H})|) = O(|E(H)|)$ time. □

We can define $H|^b$ symmetrically exchanging the role of red and blue, and obtain an analogous result on detecting rigid blue splitters.

To construct graphs $H|^r$ and $H|^b$ of $H = \sigma(\nu)$, we need to know whether ν is r-joined or b-joined if ν is not the root ν^*. The following lemmas tell that we can compute the r- and b-joinedness and the existence of rigid splitters inductively.

Lemma 4 *Let ν be a P- or S-node in $\mathcal{T}$ such that $G^-(\nu)$ is an r- or br-graph. Assume that when ν is a non-root, the r-joinedness of ν has been computed. Then the r-joinedness of all child nodes $\mu \in Ch(\nu)$ of $\mathcal{T}$ can be computed in $O(|E(\sigma(\nu))|)$ time.*

Proof. We consider the r-joinedness (the b-joinedness can be discussed symmetrically).

Let ν be the root P-node ν^*. Let $\{u, v\} = V(\sigma(\nu^*))$ and $E(\sigma(\nu^*)) = \{e_1, e_2, \ldots, e_p\}$. For each virtual edge e_i, let $\mu_{e_i} \in Ch(\nu^*)$ denote the corresponding child node. Then a child node μ_{e_i} is r-joined if and only if there is an r- or br-edge e_j with $j \neq i$. Hence we can find all r-joined nodes in $Ch(\nu^*)$ in $O(|E(\sigma(\nu^*))|)$ time. Let ν be a nonroot P-node. This case can be also treated in the similar manner with the root P-node case as follows. If ν is not r-joined, then we apply the same procedure for the root P-node case to the edge set $E(\sigma^-(\nu))$. If ν is r-joined, then we apply the procedure to the edge set $E(\sigma(\nu))$, where the parent virtual edge of ν is treated as an r-edge.

Let ν be a nonroot S-node, and $E(\sigma^-(\nu)) = \{e_1, e_2, \ldots, e_p\}$. For each a virtual edge e_i, let $\mu_{e_i} \in Ch(\nu^*)$ denote the corresponding child node. A node μ_{e_i} is r-joined if and only if ν is r-joined and all other edges e_j $(j \neq i)$ are r- or br-edges.

Therefore, we can find all r-joined child nodes of a P- or S-node ν in $O(|E(\sigma^-(\nu))|)$ time. □

Lemma 5 *Let ν be an R-node in $\mathcal{T}$ such that $G^-(\nu)$ is an r- or br-graph. Assume that the r-joinedness of ν has been computed. Then a rigid red splitter at ν can be detected in $O(|E(\sigma(\nu))|)$ time. If there is no rigid red splitter at ν, then the r-joinedness of all child nodes $\mu \in Ch(\nu)$ of $\mathcal{T}$ can be computed in $O(|E(\sigma(\nu))|)$ time.*

Proof. We consider the r-joinedness (the b-joinedness can be discussed symmetrically). We first show how to compute the r-joinedness of child nodes in $Ch(\nu^*)$ for the root ν^* of $\mathcal{T}$.

Let ν be the root R-node ν^*, and ζ_{ν^*} be a plane embedding of the triconnected planar graph $H = \sigma(\nu^*)$. By Lemma 3, we can find a rigid red splitter at ν^* in $O(|E(\sigma(\nu^*))|)$ time. Hence assume that there is no rigid red splitter at ν^*.

Let $\mu \in Ch(\nu^*)$ be a node which corresponds to a virtual r- or br-edge e_μ. If μ is r-joined, then the corresponding edge e_μ is contained in a cycle Q_e of H that consists of r-edges. A face is called *red* if its facial cycle contains only r- and br-edges. Clearly any virtual edge on the boundary of a red face corresponds to an r-joined node. We claim that any r-joined node can be found in this way. If neither of the two faces sharing the edge e_μ is a red face, then the cycle Q_e containing e would be a rigid red splitter at ν^*, a contradiction to the assumption. Therefore, all r-joined nodes in $Ch(\nu^*)$ which virtual r- or br-edges are those virtual edges in red faces of ζ_{ν^*}. Finding all virtual edges in all red faces of ζ_{ν^*}. This takes $O(|V(\sigma(\nu^*))|)$ time.

Let ν be a nonroot R-node. To compute the r-joinedness of nodes in $Ch(\nu)$, we let ζ_ν be a plane embedding of $\sigma(\nu)$ (resp., $\sigma^-(\nu)$) in which vertices u and v appear on its boundary if ν is r-joined (resp., ν is not r-joined). Then we can find all the r-joined nodes by applying the same procedure for the root R-node to the embedding ζ_ν, where the parent virtual edge of ν is treated as an r-edge in ζ_ν if ν is r-joined. □

By Lemmas 3, 4 and 5, we can test whether G has a rigid splitter at some R-node or not in linear time. Moreover, if G has no rigid splitter at any R-node, then we can compute the r-joinedness of all nodes which correspond to virtual r- or br-edges, and the b-joinedness of all nodes which correspond to virtual b- or br-edges in linear time.

Theorem 6 *Let $G = (V, E_1 \cup E_2)$ be a planar graph with $E_1 \cap E_2 = \emptyset$, and $\mathcal{T}$ be a rooted SQRT-tree of G. Then whether G contains a rigid splitter at some R-node can be tested in linear time.* □

5 Detecting Splitter-free Embeddings

In this section, we assume that a given planar graph $G = (V, E_1 \cup E_2)$ has no rigid splitter at any R-node, and present a procedure for testing whether the graph G admits a splitter-free plane embedding.

Let ν be a nonroot of $\mathcal{T}$, (u, v) be its virtual parent edge, and ψ be a plane embedding of the graph $G^-(\nu)$ such that u and v appear in the boundary. When we traverse the boundary of ψ_ν, we denote the walk along the boundary from u to v (resp., from v to u) by $B_{u,v}(\psi_\nu)$ (resp., $B_{v,u}(\psi_\nu)$). If $G^-(\nu)$ has no red u, v-path, then the path $B_{u,v}(\psi_\nu)$ is always r-rimmed; when $G^-(\nu)$ has a red u, v-path, path $B_{u,v}(\psi_\nu)$ is called *r-rimmed* if

(i) $B_{u,v}(\psi_\nu)$ is a red u, v-path of $G^-(\nu)$, and any other red u, v-path of $G^-(\nu)$ passes through only vertices in $B_{u,v}(\psi_\nu)$; and

(ii) neither a blue edge nor a vertex is enclosed by path $B_{u,v}(\psi_\nu)$ and any red u, v-path of $G^-(\nu)$.

We define *b-rimmed* boundaries symmetrically by exchanging red and blue colors in the definition.

For example, the path $B_{u,v}(\psi_1)$ of the embedding ψ_1 in Fig. 4(a) is not r-rimmed and the path $B_{v,u}(\psi_1)$ of ψ_1 is not b-rimmed. For the embedding ψ_2 in Fig. 4(b), $B_{u,v}(\psi_2)$ and $B_{v,u}(\psi_2)$ are r-rimmed and b-rimmed, respectively.

It is easy to see that, for a splitter-free plane embedding ψ_G of G, if a nonroot ν is r-joined then the embedding ψ_ν of the graph $G^-(\nu)$ induced from ψ_G must have an r-rimmed path $B_{u,v}(\psi_\nu)$ or $B_{u,v}(\psi_\nu)$.

Now we consider a plane embedding ζ of $H = \sigma^-(\nu)$ of a nonroot R-node ν. When we traverse the boundary of ζ, we denote the walk along the boundary from u to v (resp., from v to u) by $B_{u,v}(\zeta)$ (resp., $B_{v,u}(\zeta)$). We call a u, v-path in H an *r-u, v-path* (resp., *b-u, v-path*) if it consists of r-edges and br-edges (resp., b-edges and br-edges). For the embedding ζ in Fig. 4(c), the path $B_{u,v}(\zeta)$ is an r-u, v-path and the path $B_{v,u}(\zeta)$ is a b-u, v-path.

We first observe the structure of R-nodes when G has no rigid splitters.

Lemma 7 *Assume that $G = (V, E_1 \cup E_2)$ has no rigid splitters at any R-node. Then the following holds for each nonroot R-node ν of $\mathcal{T}$ and an arbitrary plane embedding ζ_ν of $H = \sigma^-(\nu)$ such that the end vertices of the parent virtual edge $e_\nu = (u, v)$ of ν appear in its boundary:*

(i) *One of $B_{u,v}(\zeta_\nu)$ and $B_{v,u}(\zeta_\nu)$ is not an r-u,v-path;*

(ii) *One of $B_{u,v}(\zeta_\nu)$ and $B_{v,u}(\zeta_\nu)$ is not a b-u,v-path;*

(iii) *If $G^-(\nu)$ is an r-graph or br-graph and ν is r-joined, then one of $B_{u,v}(\zeta_\nu)$ and $B_{v,u}(\zeta_\nu)$ is an r-u,v-path and H has no other r-u,v-path; and*

(iv) *If $G^-(\nu)$ is a b-graph or br-graph and ν is b-joined, then one of $B_{u,v}(\zeta_\nu)$ and $B_{v,u}(\zeta_\nu)$ is a b-u,v-path and H has no other b-u,v-path.*

Proof. We show (i) and (iii) since (ii) and (iv) can be shown symmetrically.

(i) If both $B_{u,v}(\zeta_\nu)$ and $B_{v,u}(\zeta_\nu)$ are r-u,v-paths, then they form a cycle Q' in $\sigma^-(\nu)$ which corresponds to a red cycle Q in G, and Q separates an inner vertex $w \in V(\sigma^-(\nu)) - V(Q')$ from the vertices in $V - V(G^-(\nu))$, contradicting that G has no rigid splitter at any R-node.

(iii) Since $G^-(\nu)$ is an r-graph or br-graph, it has a red u,v-path P. Since ν is r-joined, $G^+(\nu)$ contains a red u,v-path P^+. Hence these u,v-paths form a red cycle Q in G. Let P' be the r-u,v-path in $\sigma^-(\nu)$ that corresponds to P. If P' is neither $B_{u,v}(\zeta_\nu)$ nor $B_{v,u}(\zeta_\nu)$, then Q would be a rigid splitter at ν. By (i), at most one of $B_{u,v}(\zeta_\nu)$ and $B_{v,u}(\zeta_\nu)$ can be an r-u,v-path. This also implies that $\sigma^-(\nu)$ has no other r-u,v-path than P'. □

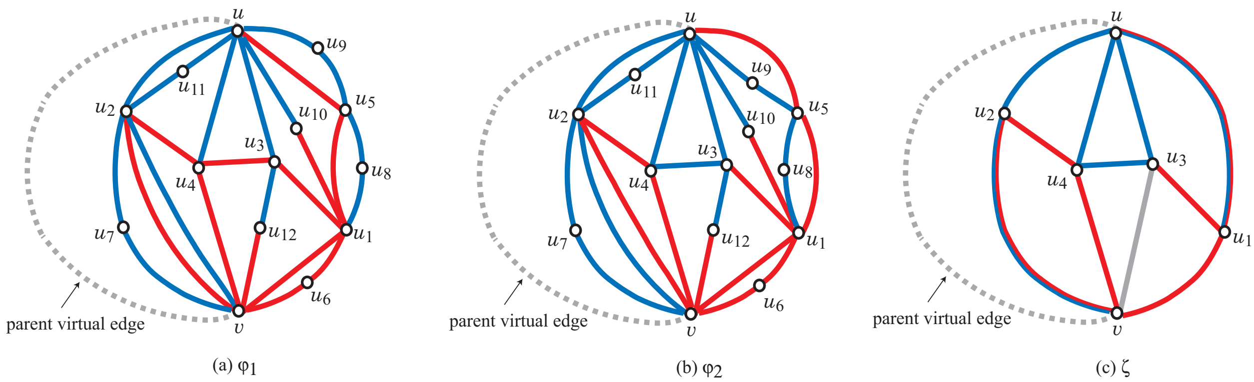

Figure 4: Illustration for the definition of r-rimmed embeddings of graphs $G^-(\nu)$ and $\sigma^-(\nu)$ for a nonroot R-node ν. (a) a plane embedding ψ_1 of a graph $G^-(\nu)$, where $B_{u,v}(\psi_1)$ is neither r-rimmed nor b-rimmed; (b) a plane embedding ψ_2 of another graph $G^-(\nu)$, where $B_{u,v}(\psi_2)$ and $B_{v,u}(\psi_2)$ are r-rimmed and b-rimmed, respectively; (c) the plane embedding ζ of graph $\sigma^-(\nu)$ of the graph $G^-(\nu)$ in (b), where the path $B_{u,v}(\zeta)$ is an r-u,v-path and the path $B_{v,u}(\zeta)$ is a b-v,u-path.

We will show that whether G admits a splitter-free plane embedding or not depends on the structure of skeletons for P-nodes. Consider P-nodes including the case of the root P-node; the parent virtual edge e of a nonroot node ν is treated as a br-edge (resp., b-edge and r-edge) if the virtual edge e_ν corresponding to the node ν is b-joined and r-joined (resp., "b-joined but not r-joined," and "r-joined but not b-joined"). In the skeleton $\sigma(\nu)$ of a P-node ν, a pair of red and blue real edges between the two vertices in $V(\sigma(\nu))$ will be treated as as two real edges.

For a notational simplicity, we let $E^r(H)$ (resp., $E^b(H)$ and $E^{br}(H)$) denote the set of *virtual* r-edges (resp., b-edges and br-edges) in the skeleton $H = \sigma(\nu)$ of a node ν (note that H contains the parent virtual edge of ν).

Lemma 8 *Assume that $G = (V, E_1 \cup E_2)$ has no rigid splitters at any R-node. Then G has a splitter-free plane embedding ψ_G if and only if every P-node ν of $\mathcal{T}$ satisfies the following, where $H = \sigma(\nu)$:*

(i) $|E^r(H)| + |E^{br}(H)| \leq 2$ *and* $|E^b(H)| + |E^{br}(H)| \leq 2$ *hold; and*

(ii) *If $|E^{br}(H)| = 2$ holds, then H has exactly two virtual edges (i.e., $E^r(H) = E^b(H) = \emptyset$).*

In what follows, we give a proof of the lemma.

Only if part: We first prove the only if part. Let $\{u, v\} \in V(H)$, and $E^r(H) \cup E^{br}(H) = \{e_1, e_2, \ldots, e_p\}$. For edge $e_i \in E(\sigma^-(\nu))$, let G_{e_i} denote the subgraph $G^-(\mu_{e_i})$ for the node μ_{e_i} corresponding to the edge e_i. If e_i is the parent virtual edge of ν, then let G_{e_i} denote the graph $G^+(\nu)$. Then each graph G_{e_i} contains a red u, v-path P_i. The two paths P_i and P_j $(1 \leq i < j \leq p)$ form a red cycle $Q_{i,j}$.

(i) Assume that $E^r(H) \cup E^{br}(H)$ contains three edges e_1, e_2, e_3 and that, in a splitter-free plane embedding ψ of G, G_1, G_2 and G_3 appear in this order, i.e., $Q_{1,3}$ surrounds the graph $G_{e_2} - \{u, v\}$. Since both red cycles $Q_{1,2}$ and $Q_{2,3}$ are not splitters, neither of them surrounds a vertex or a blue edge. Hence any blue edge exists in the exterior of cycle $Q_{1,3}$, which now separates such a blue edge and a vertex in $G_{e_2} - \{u, v\}$. This means that $Q_{1,3}$ is a red splitter, contradicting that ψ is splitter-free.

(ii) Let $e_1, e_2 \in E^{br}(H)$, and assume that there is a virtual edge e_3 in $E(H) - \{e_1, e_2\}$. Since each G_{e_i}, $i = 1, 2$ contains a red u, v-path P_i and a blue u, v-path P'_i, G has a red cycle Q formed by P_1 and P_2, and a blue cycle Q' formed by P'_1 and P'_2. Let ψ be a splitter-free plane embedding of G. Since the red cycle Q is not a splitter, one of the interior and exterior (say interior) of Q contains a blue edge or a vertex. Now the blue cycle Q' separates a vertex in $G_{e_3} - \{u, v\}$ and a red edge in Q, implying that Q' is a blue splitter, a contradiction to that ψ is splitter-free.

If part: We next prove the if part of Lemma 8 by giving a procedure for constructing a splitter-free plane embedding ψ_G of G.

By induction we construct the following embedding ψ_ν of $G^-(\nu)$ for each node of $\mathcal{T}$.

(a) Any red (resp., blue) cycle in ψ_ν encloses only red (resp, blue) edges; i.e., no red or blue cycle in ψ_ν can be a splitter in any extension ψ_G of G from ψ_ν for the graph $G^-(\nu)$.

(b) If ν is a nonroot r-joined (resp., b-joined) node, then one of $B_{u,v}(\psi_\nu)$ and $B_{v,u}(\psi_\nu)$ is r-rimmed (resp., b-rimmed) for the parent virtual edge (u, v) of ν.

We call such an embedding ψ_ν *proper*.

Root nodes: Suppose that we have a proper embedding ψ_μ for all child nodes $\mu \in Ch(\nu^*)$ of the root ν^* of $\mathcal{T}$. Then we can obtain a splitter-free embedding ψ_G of G, by combining the embeddings ψ_μ, $\mu \in Ch(\nu^*)$ as follows.

(1) R-node: Let ν^* be an R-node: Let ζ_{ν^*} be a plane embedding of $\sigma(\nu^*)$, where a face of ζ_{ν^*} is called *red* (resp., *blue*) if its facial cycle contains only r- and br-edges (resp., b- and br-edges). We construct a plane embedding of G by replacing each virtual edge $e = (u, v)$ in ζ_{ν^*} with a proper embedding ψ_{μ_e} of its corresponding node μ_e. For each proper embedding ψ_{μ_e}, there are exactly two ways of placing the embedding ψ_{μ_e} between the end vertices u and v; we flip the embedding ψ_{μ_e} if necessary so that if μ is r-joined (resp, b-joined) then the r-rimmed (resp., b-rimmed) path of the boundary of ψ_{μ_e} appear along a red face (resp., blue face) of ζ_{ν^*}. If μ_e is r-joined (resp, b-joined) and e is a virtual r- or br-edge (resp., b- or br-edge), then one of the two faces in ζ_{ν^*} that share the edge e is a red (resp., blue) face (as observed in the proof of Lemma 5) and also not both of them are red (resp., blue) faces (since otherwise G would have a red (resp., blue) rigid splitter). Hence we can place all proper embeddings of child nodes in the above way. The resulting embedding for

G is splitter-free, because by the definition (i) of proper embeddings, there is no splitter which is completely contained in some proper embedding ψ_{μ_e}. Also by the structure of ζ_{ν^*}, a red (resp., blue) splitter can appear along a red (resp., blue) face of ζ_{ν^*}. However, any such red (resp., blue) cycle cannot be a splitter since each virtual edge e along a red cycle is replaced by an r-rimmed (resp., b-rimmed) embedding ψ_{μ_e}.

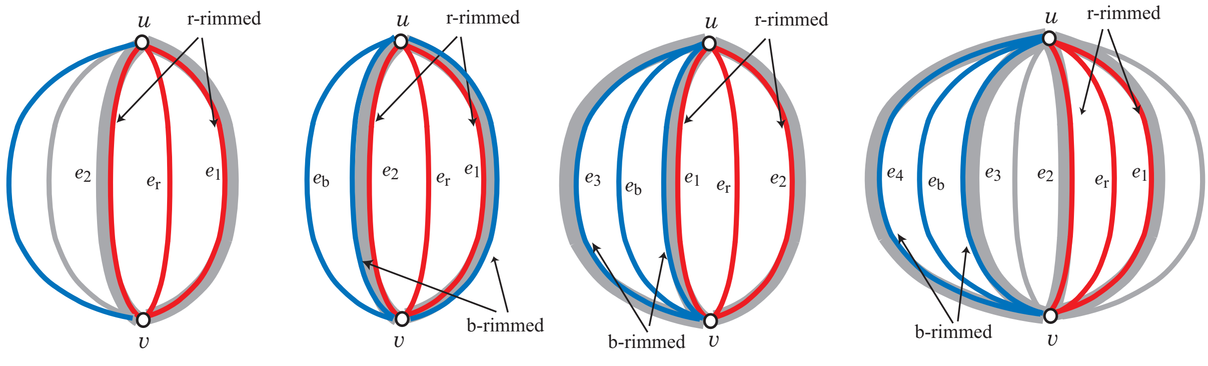

Figure 5: Illustration for constructing proper embeddings for a P-node; (a) an ordering of edges e_1, e_2, e_r; (b) an ordering of edges e_1, e_1, e_r, e_b; (c) an ordering of edges e_1, e_2, e_3, e_r, e_b; (d) an ordering of edges $e_1, e_2, e_r, e_3, e_4, e_b$.

(2) P-node: Let ν^* be a P-node: Let $H = \sigma(\nu^*)$ and $\{u, v\} = V(H)$. When H has a real red (resp., blue) edge (u, v) then we denote it by e_r (resp., e_b). By assumption, the condition (i) of Lemma 8 holds. If $|E^r(H)| + |E^{br}(H)| \leq 1$ and $|E^b(H)| + |E^{br}(H)| \leq 1$ hold, then we let ζ_{ν^*} be an arbitrary plane embedding of $\sigma(\nu^*)$, and replace each virtual edge $e = (u, v)$ in ζ_{ν^*} with a proper embedding ψ_{μ_e} of its corresponding node μ_e so that no splitting passing through $\{u, v\}$ is created. This is possible because at most one of a red or blue cycle passing through $\{u, v\}$ can be created and we can make it surround nothing in its interior by flipping a proper embedding ψ_{μ_e} if necessary.

We next consider the case where $|E^r(H)| + |E^{br}(H)| = 2$ and $|E^b(H)| + |E^{br}(H)| \leq 1$ (the case of $|E^r(H)| + |E^{br}(H)| \leq 1$ and $|E^b(H)| + |E^{br}(H)| = 2$ can be treated symmetrically). For the two edges $e_1, e_2 \in E^r(H) \cup E^{br}(H)$, we let ζ_{ν^*} be a plane embedding of $\sigma(\nu^*)$ satisfies the following

constraint 1: there is no other virtual edge between e_1 and e_2, and e_r (if any) appears between e_1 and e_2 (see Fig. 5(a)).

We then replace each virtual edge $e = (u, v)$ in ζ_{ν^*} with a proper embedding ψ_{μ_e} of its corresponding node μ_e. We place the proper embeddings $\psi_{\mu_{e_1}}$ and $\psi_{\mu_{e_2}}$ so that their r-rimmed paths meet each other at the same face f of ζ_{ν^*}, when we neglect e_r (if any). By a similar reason with the case of the root R-node, we see that the resulting plane embedding ψ_G of G is splitter-free.

Finally consider the case where $|E^r(H)| + |E^{br}(H)| = |E^b(H)| + |E^{br}(H)| = 2$.

If $|E^{br}(H)| = 2$, then, by the condition (ii) of Lemma 8, there is no other virtual edge than $e_1, e_2 \in E^{br}(H)$. Let ζ_{ν^*} be a plane embedding of $\sigma(\nu^*)$ satisfies the following

constraint 2: the edges appear in the order e_1, e_r, e_2, e_b or e_1, e_b, e_2, e_r (see Fig. 5(b)).

Then we can obtain a splitter-free plane embedding of G by replacing the virtual edges e_i with proper embeddings $\psi_{\mu_{e_i}}$ so that their r-rimmed (resp., b-rimmed) paths meet at the same face, when we neglect e_r and e_b (if any).

Let $|E^{br}(H)| = 1$. Then $|E^r(H)| = |E^b(H)| = 1$. Let $e_1 \in E^{br}(H)$, $e_2 \in E^r(H)$, $e_3 \in E^b(H)$, and $e_4, e_5, \ldots, e_p$ be the other virtual edges in H (if any). Let ζ_{ν^*} be a plane embedding of $\sigma(\nu^*)$ satisfies the following

constraint 3: the edges e_1, e_2, e_3, e_r, e_b appear consecutively in the order e_2, e_r, e_1, e_b, e_3 or in its reversal in ζ_{ν^*} (see Fig. 5(c)).

Hence the other virtual edges e_i must appear between e_2 and e_3 in the cyclic order by ζ_{ν^*}. We then replace each virtual edge e_i with its proper embedding $\psi_{\mu_{e_i}}$ so that the r-rimmed (resp., b-rimmed) paths of $\psi_{\mu_{e_1}}$ and $\psi_{\mu_{e_2}}$ (resp., $\psi_{\mu_{e_1}}$ and $\psi_{\mu_{e_3}}$) meet at the same face, when we neglect e_r and e_b (if any). Similarly we see that the resulting plane embedding of G is splitter-free.

Let $|E^{br}(H)| = 0$. Then $|E^r(H)| = |E^b(H)| = 2$. Let $e_1, e_2 \in E^r(H)$, $e_3, e_4 \in E^b(H)$, and $e_5, e_6, \ldots, e_p$ be the other virtual edges in H (if any). Let ζ_{ν^*} be a plane embedding of $\sigma(\nu^*)$ satisfies the following

constraint 4: the edges e_1, e_2, e_r (resp., e_3, e_4, e_b) appear consecutively in the order e_1, e_r, e_2 or in its reversal (resp., in the order e_3, e_b, e_4 or in its reversal). See Fig. 5(d).

Then we can obtain a splitter-free plane embedding of G by replacing the virtual edges e_i with proper embeddings $\psi_{\mu_{e_i}}$ so that r-rimmed (resp., b-rimmed) paths of $\psi_{\mu_{e_1}}$ and $\psi_{\mu_{e_2}}$ (resp., $\psi_{\mu_{e_3}}$ and $\psi_{\mu_{e_4}}$) meet at the same face, when we neglect e_r and e_b (if any).

Therefore, to prove the if part, it suffices to show that we can construct proper embeddings inductively.

Leaf nodes: Let ν be a leaf node of $\mathcal{T}$, and (u, v) be the parent virtual edge of ν.

(1) P-node: Let ν be a P-node. In this case, $G^-(\nu) = \sigma^-(\nu)$ consists of a pair of red and blue edges. Obviously, any embedding ψ_ν of $\sigma^-(\nu)$ satisfies the conditions (a) and (b), and ψ_ν is proper.

(2) S-node: Let ν be an S-node. In this case, $G^-(\nu) = \sigma^-(\nu)$ is a u, v-path P. Let $e_i = (u_i, u_{i+1})$, $i = 1, 2, \ldots, p$ be the edges in P, where $u_1 = u$ and $u_{p+1} = v$. Let ψ_ν be the unique plane embedding of $\sigma^-(\nu)$. Clearly ψ_ν contains no cycle and satisfies the condition (a). Also ψ_ν is always r-rimmed and b-rimmed, satisfying the condition (b).

(3) R-node: Let ν be an R-node. Let ψ_ν be a plane embedding of $G^-(\nu) = \sigma^-(\nu)$ in which the vertices u and v appear in its boundary. We prove that ψ_ν is proper. If ψ_ν contains a red or blue cycle Q which can be a splitter in an extension ψ_G of G from ψ_ν for the graph $G^-(\nu)$, then such Q must be the boundary of ψ_ν, since G has no rigid splitter. However, by the conditions (i)-(ii) in Lemma 7, such Q cannot exist, and ψ_ν satisfies (a). We see that ψ_ν satisfies (b) by the conditions (iii)-(iv) in Lemma 7.

Internal nodes: Let ν be an internal node of $\mathcal{T}$, and (u, v) be the parent virtual edge of ν. For each virtual edge $e \in E(\sigma^-(\nu))$, the corresponding child node in $Ch(\nu)$ is denoted by μ_e.

(1) S-node: Let ν be an S-node. In this case, an embedding ζ_ν of $\sigma^-(\nu)$ is a single path joining u and v. Let ψ_ν be an embedding of $G^-(\nu)$ obtained from ζ_ν by replacing each virtual edge $e \in E(\sigma^-(\nu))$ with a proper embedding ψ_{μ_e} of the graph $G^-(\mu_e)$. Since any cycle in $G^-(\nu)$ is contained in some $G^-(\mu_e)$, the resulting embedding ψ_ν also satisfies the condition (a), even if we flip some embedding ψ_{μ_e} in ψ_ν. If ν is not r-joined or b-joined, then the condition (b) also holds for ψ_ν. Consider the case where $G^-(\nu)$ is a br-graph and ν is r-joined and b-joined (the case where ν is r-joined or b-joined can be treated analogously). Since $G^-(\nu)$ is a br-graph, all child nodes $\mu_e \in Ch(\nu)$ are r-joined, and every edge in $E(\sigma^-(\nu))$ is a virtual edge (since any red (resp., blue) real edge in $E(\sigma^-(\nu))$ would imply that there is no blue (resp., red) u, v-path in $G^-(\nu)$). Hence each proper embedding ψ_{μ_e} has an r-rimmed path and a b-rimmed path on its boundary. Then we can make the paths $B_{u,v}(\psi_\nu)$ and $B_{u,v}(\psi_\nu)$ r-rimmed and b-rimmed, respectively, by flipping some embeddings ψ_{μ_e} in ψ_ν if necessary. Thus, the resulting embedding ψ_ν also satisfies the condition (b).

(2) R-node: Let ν be an R-node, and ζ_ν be a plane embedding of $H = \sigma^-(\nu)$ in which u and v appear in its boundary. Consider the case where $G^-(\nu)$ is an r- or br-graph and ν is r-joined, but not b-joined (the case where ν is b-joined can be treated analogously). By the condition (iii) in Lemma 7, we can assume without loss of generality that $B_{u,v}(\zeta_\nu)$ is an r-u,v-path P. We replace each virtual edge e in H by a proper embedding ψ_{μ_e} of the graph $G^-(\mu_e)$ of the corresponding child node $\mu_e \in Ch(\nu)$ in the same manner of the root R-node case, where we place the proper embedding ψ_{μ_e} for each edge in the r-u,v-path P so that its r-rimmed path appear along the outer face of ζ_ν. Let ψ_ν be the resulting embedding of $G^-(\nu)$. Since the r-u,v-path P along the boundary of ζ_ν is the unique r-u,v-path of $\sigma^-(\nu)$ by Lemma 7(iii), we see that ψ_ν satisfies the condition (b).

Now we consider whether a red cycle Q of $G^-(\nu)$ can be a splitter in some plane embedding of G. If Q is contained in some $G^-(\mu_e)$ of a virtual edge $e \in E(\sigma^-(\nu))$, then Q cannot be a splitter in any embedding of G since ψ_{μ_e} satisfies the condition (a). Assume that Q is not contained in any $G^-(\mu_e)$, i.e., the corresponding cycle Q' appear in $\sigma^-(\nu)$. We show that Q encloses only red real edges in ψ_ν. Since Q' is not a rigid splitter by the assumption, it encloses only red real edges in the embedding ζ_ν, implying that the region enclosed by Q' consists of red faces. Moreover each virtual edge $e \in E(Q')$ is replaced with a proper embedding ψ_{μ_e} so that its r-rimmed path appear along the red faces. This means that Q encloses only red real edges in ψ_ν, and hence ψ_ν satisfies the condition (a). Therefore ψ_ν is a proper embedding of $G^-(\nu)$.

(3) P-node: Let ν be a P-node. To compute a proper embedding of $G^-(\nu)$, we consider an embedding ζ_ν of $\sigma(\nu)$, i.e., a cyclic ordering of all edges in $\sigma(\nu)$ including the parent virtual edge e_ν. We regard e_ν as an r-edge (resp., b-edge; br-edge) if ν is r-joined, but not b-joined (resp., b-joined but not r-joined; r-joined and b-joined). Then we apply the same procedure for the root P-node case to the skeleton $\sigma(\nu)$ to obtain the same embedding ζ_ν. We then replace each virtual edge $e \in E(\sigma^-(\nu))$ in ζ_ν with a proper embedding ψ_{μ_e}. Let ψ^* be the resulting embedding of $G^-(\nu)$ plus the edge e_ν. Let ψ_ν be the resulting embedding of $G^-(\nu)$ obtained from ψ^* by deleting e_ν.

For example, we consider the P-node ν_c in Fig. 11. We have $e_e \in E^r(H), e_f \in E^{br}(H)$, $e_d, e' \in E(H) - \{e_e, e_f\}$ for $H = \sigma(\nu_c)$, where e' denotes the parent virtual edge of ν_c. In this case, $|E^r(H)| + |E^{br}(H)| = 2$ and $|E^b(H)| + |E^{br}(H)| \leq 1$ holds, and by constraint 1, the edges e_e and e_f appear consecutively in the embedding ζ of H, and the r-rimmed paths of the embeddings $\psi_{G_{e_e}}$ and $\psi_{G_{e_f}}$ of the graphs G_{e_e} and G_{e_f} need to meet each other, as shown in Fig. 3(a).

We show that ψ_ν is a proper embedding.

We first consider the case where ν is neither r-joined nor b-joined. In ψ^*, we regard e_ν as a u,v-path P which is not a red or blue path, i.e., we replace e_ν with a path of a red edge (u, w) and a blue edge (w, v). Since such P always has a proper embedding if we regard ν as the root node, we see that ψ^* must be a splitter-free embedding by the result in the root P-node case. Since no red or blue cycle passes through P in ψ^*, any red or blue cycle Q passes through the graph $G^-(\nu)$. Since ψ^* is splitter-free, any such cycle cannot be a splitter in any extension of ψ_ν. Thus the condition (a) holds for ψ_ν. Clearly the condition (b) holds for ψ_ν, since ν is neither r-joined nor b-joined.

We next consider the case where ν is r-joined but not b-joined (the other cases can be discussed analogously). Regard e_ν as a red u,v-path in ψ^*, i.e., replace e_ν with a path of two red edges (u, w) and (w, v). Then ψ^* must be a splitter-free embedding by the result in the root P-node case. This means that exactly one of the u,v-paths along the boundary of ψ_ν must be r-rimmed. Hence ψ_ν satisfies the condition (b). Let Q be a red or blue cycle in $G^-(\nu)$. If Q does not pass through u and v, then it is contained in some $G^-(\mu_e)$ and such Q cannot be a splitter in any extension of ψ_ν, because ψ_{μ_e} already satisfies such a property (condition (a)). If Q passes through u and v, then there are two edges $e, e' \in E(\sigma^-(\nu))$ such that Q is contained in the union of $G^-(\mu_e)$ and $G^-(\mu_{e'})$. In this case, at least one of e and e' must be a real red edge by the condition (i) of Lemma 8, and e, e' and the parent virtual edge e_ν appear consecutively in ζ_ν. Then such Q cannot be a splitter in any extension of ψ_ν, since we have shown that ψ_ν satisfies the condition (b).

This completes the inductive proof for the existence of proper embeddings. Thus we have proved the if part of Lemma 8. It is not difficult see that the above procedure for constructing a splitter-free plane embedding ψ_G of G can be implemented to run in linear time.

Theorem 9 *Let $G = (V, E_1 \cup E_2)$ be a planar graph with $E_1 \cap E_2 = \emptyset$. Then whether G has a splitter-free plane embedding or not can be tested in linear time.* □

6 Detecting Disjunctive Embeddings

This subsection shows how to test whether a given planar graph $G = (V, E_1 \cup E_2)$ has a disjunctive embedding. Let ν be a node in a rooted SPQR tree $\mathcal{T}$ of G, and $e = (u, v)$ be an edge of its skeleton $\sigma^-(\nu)$. Let $G_e = G^-(\mu_e)$ for the child $\mu_e \in Ch(\nu)$ corresponding to e if e is a virtual edge, and G_e be the graph consisting of the real edge e otherwise. We say that the edge e is *r-colored* (resp., *b-colored* at an endpoint u of e if G_e contains at least one red (resp., blue) edge incident to u, and that the edge e is *bi-colored* at u if it is r- and b-colored at u. The edge e is *mono-colored* at u if it is not bi-colored at u. Any edge which is bi-colored at a vertex is a virtual edge.

We consider a disjunctive embedding ψ of the graph G_e, and define the color-pattern of an endpoint u (resp., v) of e in ψ to be the sequence of the colors of the edges incident to u (resp., v) in the clockwise order around u (resp., v) starting from the edge in the boundary $B_{u,v}(\psi)$ (resp., $B_{v,u}(\psi)$). For example, the graph G_2 in Fig. 3(c) has a nonroot R-node ν_c (see Fig. 12). In the embedding of $G^-(\nu_c)$ of the R-node ν_c in Fig. 3(c), we have the sequence (red, red, blue, red, red) of the colors of the edges around u, which is simply denoted by `rbr`. The graph G_1 in Fig. 3(a) has a P-node ν_c (see Fig. 11), and the embedding of $G^-(\nu_c)$ in Fig. 3(a) gives the sequence (red, blue, red, red, red, blue, blue) of the colors of the edges around u, which implies that the embedding is not disjunctive at u.

For any vertex v, there are six color-patterns `r,b,rb,br,rbr,brb` that can be part of a disjunctive sequence of the edge colors around v. For a color-pattern α, we denote by $\alpha^\top$ its reversal. For example, $(rb)^\top = br$. We define the *configuration* of ψ of the graph G_e to be the ordered pair of the color-patterns of the endpoints u and v of e. Let $\tau(u, v; G_e)$ be the set of all configurations of disjunctive embeddings of the graph G_e. For each real red (resp., blue) edge $e = (u, v)$, $\tau(u, v; G_e) = \{(\texttt{r},\texttt{r})\}$ (resp., $\tau(u, v; G_e) = \{(\texttt{b},\texttt{b})\}$). Note that $(\alpha, \beta) \in \tau(u, v; G_e)$ if and only if $(\beta, \alpha) \in \tau(v, u; G_e)$

For example, we consider the graph G_1 in Fig. 3(a). For the virtual edges $e_a, e_b, \ldots, e_k$ of the nodes in the SPQR tree of G_1 in Fig. 11, we have

$\tau(u_1, v; G_{e_k}) = \{(\texttt{r},\texttt{b})\}$, $\tau(u_1, v; G_{e_i}) = \{(\texttt{r},\texttt{rb}), (\texttt{r},\texttt{br})\}$, $\tau(u, u_1; G_{e_j}) = \{(\texttt{b},\texttt{b})\}$,
$\tau(u, u_1; G_{e_h}) = \{(\texttt{br},\texttt{rb}), (\texttt{rb},\texttt{br})\}$, $\tau(u, v; G_{e_e}) = \{(\texttt{rb},\texttt{br}), (\texttt{br},\texttt{br}), (\texttt{rb},\texttt{rb}), (\texttt{rb},\texttt{br})\}$,
$\tau(u, v; G_{e_d}) = \{(\texttt{r},\texttt{r})\}$, $\tau(u, v_1; G_{e_g}) = \{(\texttt{rb},\texttt{br}), (\texttt{br},\texttt{rb})\}$,
$\tau(u, v; G_{e_f}) = \{(\texttt{rb},\texttt{br}), (\texttt{brb},\texttt{br}), (\texttt{br},\texttt{rb}), (\texttt{brb},\texttt{rb})\}$, $\tau(u, v; G_{e_c}) = \{(\texttt{rbr},\texttt{rbr}), (\texttt{brb},\texttt{brb})\}$,
$\tau(x_1, v; G_{e_b}) = \{(\texttt{b},\texttt{brb})\}$, $\tau(x_1, x_3; G_{e_a}) = \{(\texttt{b},\texttt{r})\}$.

We show that the configuration sets $\tau(u, v; G_e)$, $e = (u, v) \in E(\sigma^-(\nu))$ for all nodes ν in $\mathcal{T}$ can be computed by a dynamic programming. The algorithm constructs the configuration set $\tau(u, v; G_e)$ for an edge e from the configuration sets $\tau(u', v'; G_{e'})$, $e' = (u', v') \in E(\sigma^-(\nu))$ of the node ν corresponding to the edge e, during which we may detect that G_e cannot have a disjunctive embedding.

Theorem 10 *Let $G = (V, E_1 \cup E_2)$ be a planar graph with $E_1 \cap E_2 = \emptyset$. Then whether G has a disjunctive plane embedding or not can be tested in linear time.*

We give a proof of the theorem.

Leaf nodes: For each virtual edge $e = (u, v)$ that corresponds to a leaf node ν of $\mathcal{T}$, the configuration set $\tau(u, v; G_e)$ can be computed easily as follows.

(1) R-nodes: If ν is a leaf R-node, then find a plane embedding ψ_ν of $G_e = G^-(\nu) = \sigma^-(\nu)$ in which the end vertices of the parent virtual edge (u, v) of ν appear on its boundary, and let (α_u, α_v) be the pair of color-patterns α_u of u and α_v of v. We terminate the computation if α_u or α_v does not belong to {`r`,`b`,`rb`,`br`,`rbr`,`brb`}, since G cannot have a disjunctive embedding. Otherwise we let $\tau(u, v; G_e) = \{(\alpha_u, \alpha_v), (\alpha_u^\top, \alpha_v^\top)\}$.

(2) S-nodes: For a leaf S-node ν, the skeleton $\sigma^-(\nu) = G^-(\nu)$ is a series of real edges $(u_1, u_2), (u_2, u_3), \ldots, (u_k, u_{k+1})$, where $u = u_1$ and $v = u_{k+1}$. Let $\alpha_u =$`r` (resp., `b`) and $\alpha_v =$`r` (resp., `b`) if (u_1, u_2) is a red edge (resp., blue edge) and (u_k, u_{k+1}) is a red edge (resp., blue edge). Then we have $\tau(u, v; G_e) = \{(\alpha_u, \alpha_v)\}$.

(3) P-nodes: In this case, $\sigma^-(\mu)$ is given as a pair of red and blue edges joining the vertices u and v. Then we have $\tau(u, v; G_e) = \{$(`rb`,`br`), (`br`,`rb`)$\}$.

Internal nodes: Next we consider the case where an virtual edge $e = (u, v)$ corresponds to an internal node ν in $\mathcal{T}$.

(1) S-nodes: We consider the case where ν is an S-node. Let $e_1 = (u_1, u_2), e_2 = (u_2, u_3), \ldots, e_p = (u_p, u_{p+1})$ be the edges in its skeleton $\sigma^-(\nu)$, and $\tau(u_i, u_{i+1}; G_{e_i})$, $i = 1, 2, \ldots, p$ be the configuration sets for the edges e_i in μ. We say that color-patterns $(\alpha_1, \alpha_2) \in \tau(u_i, u_{i+1}; G_{e_i})$ and $(\alpha_3, \alpha_4) \in \tau(u_{i+1}, u_{i+2}; G_{e_{i+1}})$ are *connectable* if the vertex u_{i+1} remains *disjunctive* for the edge color with α_2 and α_3. For example, $\alpha_2 =$`rb` and $\alpha_3 =$`rb` are not connectable. Then we can compute $\tau(u_1, u_{p+1}; G_e)$ as follows. After initializing $\tau(u_1, u_2) := \tau(u_1, u_2; G_{e_1})$, we compute for $k = 3, 4, \ldots, p + 1$,

$$\tau(u_1, u_k) := \{(\alpha_1, \alpha_4) \mid \text{connectable pairs of } (\alpha_1, \alpha_2) \in \tau(u_1, u_{k-1}) \text{ and } (\alpha_3, \alpha_4) \in \tau(u_{k-1}, u_k; G_{e_{k-1}})\}$$

and output $\tau(u_1, u_{p+1}; G_e) := \tau(u_1, u_{p+1})$.

For the virtual edge e_b which corresponds to the S-node ν_b in Fig. 11, we can obtain $\tau(x_1, v; G_{e_b}) = \{$(`b`,`brb`)$\}$ from $\tau(x_1, u; G_e) = \{$(`b`,`b`)$\}$ for the real blue edge $e = (x_1, u)$ and $\tau(u, v; G_{e_c}) = \{$(`rbr`,`rbr`), (`brb`,`brb`)$\}$, where (`b`,`b`) $\in \tau(x_1, u; G_e)$ and (`rbr`,`rbr`) $\in \tau(u, v; G_{e_c})$ are not connectable.

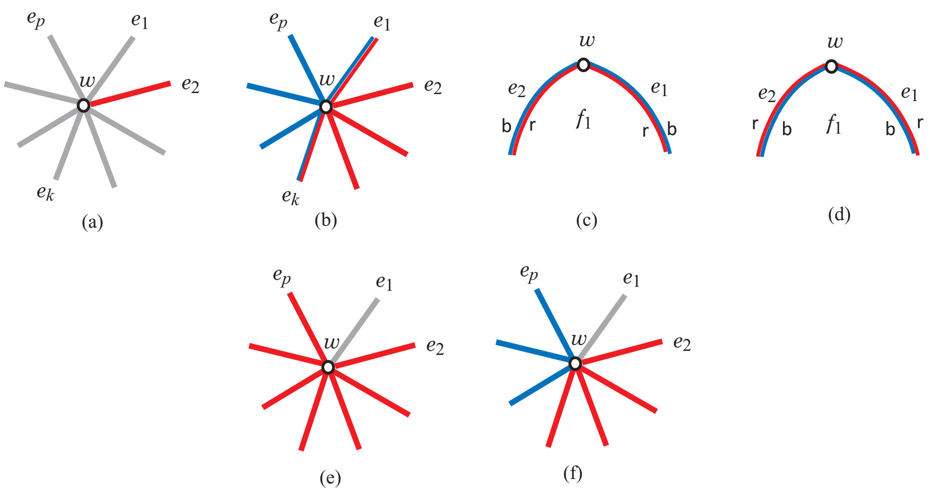

Figure 6: Illustration for the edges incident to a vertex w in the skeleton $H = \sigma^-(\nu)$ of an R-node ν. (a) $|E^{bi}(w)| = 2$, $deg(w; H) \geq 3$ and e_2 is r-colored at w; (b) $|E^{bi}(w)| = 2$, $deg(w; H) \geq 3$ and e_2 is r-colored at 2, and e_p is b-colored at w; (c) $|E^{bi}(w)| = |E(w; H)| = 2$ and $(\texttt{rb}, \beta) \in \tau(w, x_1; G_{e_1})$; (d) $|E^{bi}(w)| = |E(w; H)| = 2$ and $(\texttt{br}, \beta) \in \tau(w, x_1; G_{e_1})$; (e) $|E^{bi}(w)| = 1$, $e_1 \in E^{bi}(w)$, and e_2 and e_p are r-colored at w; (f) $|E^{bi}(w)| = 1$, $e_1 \in E^{bi}(w)$, e_2 is r-colored at w, and e_p is b-colored at w.

(2) R-nodes: We consider the case where ν is an R-node. Let ζ_ν be a plane embedding of $H = \sigma^-(\nu)$ in which the end vertices of the parent virtual edge (u, v) of ν appear on the boundary.

We check whether $G^{-}(\nu)$ has a disjunctive embedding or not by checking all possible combinations of the color-patterns around each vertex $w \in V(H)$ based on $\tau(w, x; G_{e'})$, $e' = (w, x) \in E(w; H)$. For each vertex $w \in V(H)$, let $E^{bi}(w)$ denote the set of edges of $\sigma^{-}(\nu)$ which are bi-colored at vertex w. Denote $E(w; H)$ by $\{e_i = (w, x_i) \mid i = 1, 2, \ldots, p\}$, where $e_1, e_2, \ldots, e_p$ appear around w in the clockwise order. We consider the following subcases for each vertex $w \in V(H)$.

(i) $|E^{bi}(w)| \geq 3$: In this case, the vertex w cannot be disjunctive for any plane embedding of $G^{-}(\nu)$.

(ii) $|E^{bi}(w)| = 2$ and $deg(w; H) \geq 3$: Let $e_1, e_k \in E^{bi}(w) \cap E(w; H)$, where $k \geq 3$ is assumed without loss of generality. Consider the case where e_2 is r-colored at w (the other case where e_2 is b-colored at w can be treated symmetrically). See Fig. 6(a). If there is an edge e_i ($2 \leq i < k$) which is b-colored at w (or e_j ($k < j \leq p$) which is r-colored), then we terminate the computation. See Fig. 6(b). Otherwise let $\tau'(w, x_1)$ (resp., $\tau'(w, x_k)$) be the set obtained from $\tau(w, x_1; G_{e_1})$ (resp., $\tau(w, x_k; G_{e_k})$) by deleting all the pairs of color-patterns (α, β) such that $\alpha \in \{\mathtt{rb}, \mathtt{brb}, \mathtt{rbr}\}$ (resp., $\alpha \in \{\mathtt{br}, \mathtt{brb}, \mathtt{rbr}\}$).

(iii) $|E^{bi}(w)| = |E(w; H)| = 2$: In this case, $w \in \{u, v\}$ holds, and for the inner face f_1 of H containing w, we assume without loss of generality that e_1, f_1, e_2 appear around w in the clockwise order. See Fig. 6(c) and (d). Let $\tau'(w, x_1)$ (resp., $\tau'(w, x_2)$) be the set obtained from $\tau(w, x_1; G_{e_1})$ (resp., $\tau(w, x_2; G_{e_2})$) by deleting all the pairs of color-patterns (α, β) such that $\alpha \in \{\mathtt{brb}, \mathtt{rbr}\}$ (resp., $\alpha \in \{\mathtt{brb}, \mathtt{rbr}\}$).

(iv) $|E^{bi}(w)| = 1$: Let $e_1 \in E^{bi}(w)$. Assume that the edges e_i which are r-colored (b-colored) at w appear consecutively (otherwise we can terminate the computation). If e_2 and e_p are r-colored (resp., b-colored) at w, then let $\tau'(w, x_1)$ be the set obtained from $\tau(w, x_1; G_{e_1})$ by deleting all the pairs of color-patterns (α, β) such that $\alpha = \mathtt{brb}$ (resp., $\alpha = \mathtt{rbr}$). See Fig. 6(e).

If e_2 is r-colored and e_p is b-colored (resp., e_2 is b-colored and e_p is r-colored) at w, then let $\tau'(w, x_1)$ be the set obtained from $\tau(w, x_1; G_{e_1})$ by deleting all the pairs of color-patterns (α, β) such that $\alpha \neq \mathtt{br}$ (resp., $\alpha \neq \mathtt{rb}$). See Fig. 6(f).

(v) $|E^{bi}(w)| = 0$: We check whether the edges which are r-colored at w and the edges which are b-colored at w appear consecutively around w or not. We terminate the computation if the color-pattern around w in H is not disjunctive.

For each edge $(x, y) \in E(H)$, we set $\tau'(x, y) := \tau'(x, y) \cap \{(\beta, \alpha) \mid (\alpha, \beta) \in \tau'(y, x)\}$ and $\tau'(y, x) := \tau'(y, x) \cap \{(\beta, \alpha) \mid (\alpha, \beta) \in \tau'(x, y)\}$. Finally, we compute the configuration set $\tau(u, v; G_e)$ based on the sets $\tau'(x, y)$, $(x, y) \in E(H)$. If there is an edge $(x, y) \in E(H)$ with $\tau'(x, y) = \emptyset$, then we terminate the computation. Note that H has no edge joining the vertices u and v. Hence $\tau(u, v; G_e)$ can be obtained by the set of all pairs of the color-patterns of u and v.

We compute the set $\kappa(u)$ of the color-patterns of the vertex u as follows (the set $\kappa(v)$ of the color-patterns of the vertex v can be computed symmetrically). Denote $E(u; H)$ by $\{e_i = (u, x_i) \mid i = 1, 2, \ldots, p\}$. If $|E^{bi}(u)| \geq 3$, then we terminate the computation. If $|E^{bi}(u)| = 0$, then we have a unique color-pattern of u in $G^{-}(\nu)$. If $1 \leq |E^{bi}(u)| \leq 2$, then we let $\kappa(u)$ be the set of color-patterns α of u which can be realized by choosing a color-pattern α' with $(\alpha', \beta) \in \tau'(u, x_i)$ for each $(u, x_i) \in E^{bi}(u)$, where we terminate the computation if no $\alpha \in \{\mathtt{r}, \mathtt{b}, \mathtt{rb}, \mathtt{br}, \mathtt{rbr}, \mathtt{brb}\}$ can be realized around u.

From the resulting sets $\kappa(u)$ and $\kappa(v)$, we have $\tau(u, v; G_e) = \{(\alpha, \beta), (\alpha^{\top}, \beta^{\top}) \mid \alpha \in \kappa(u),\ \beta \in \kappa(v)\}$, where we add the pairs $(\alpha^{\top}, \beta^{\top})$ to the set because H also has the other embedding which is obtained by flipping the current embedding ζ_{μ}.

(3) P-nodes: Now we consider the case where ν is a P-node. For $H = \sigma^{-}(\nu)$, e_i, $i = 1, 2, \ldots, p$ be the edges in H, which join u and v. For a vertex $w \in \{u, v\}$, if H has more than two edges which are bi-colored at w, then $G^{-}(\nu)$ cannot have a disjunctive embedding and we terminate the computation.

The configuration set $\tau(u, v; G_e)$ can be computed by investigating all essentially different ordering of edges in $E(H)$ around the vertex u. We see that $\tau(u, v; G_e)$ can be computed in $O(|E(H)|)$ time since the number of such orderings is constant.

In what follows, we show how to compute $\tau(u, v; G_e)$ in the case where the edges e_1 and e_2 are are bi-colored at u and the edges e_3 and e_4 are are bi-colored at v (the other cases can be treated in a similar manner).

We first assume that there are at most four edges which are mono-colored at u and at v by treating all edges which have the same colors at u and v as a single virtual edge (u, v).

The configuration set $\tau(u, v; G_e)$ can be computed as follows. We consider the case where there are four such edges (the other cases can be treated analogously). Denote these edges by $e_{rr}, e_{rb}, e_{br}, e_{bb}$, where $e_{cc'}$ is the edge which is c-colored at u and c'-colored at v. Hence, now we have eight edges joining u and v. Consider a cyclic ordering π of these edges around u without specifying the first and last edges, which will be chosen later to decide color-patterns of u and v. Without loss of generality we only need to consider a cyclic ordering π such that all edges which are r-colored at u appear after e_1 and before e_2, and all edges which are b-colored at u appear after e_2 and before e_1. Hence the ordering of the edges $e_1, e_2, e_{rr}, e_{rb}, e_{br}$ and e_{bb} in such a cyclic ordering π is given by one of $(e_1, e_{rr}, e_{rb}, e_2, e_{br}, e_{bb})$, $(e_1, e_{rb}, e_{rr}, e_2, e_{br}, e_{bb})$, $(e_1, e_{rr}, e_{rb}, e_2, e_{bb}, e_{br})$ and $(e_1, e_{rb}, e_{rr}, e_2, e_{bb}, e_{br})$. Among these, only $(e_1, e_{rb}, e_{rr}, e_2, e_{br}, e_{bb})$ and $(e_1, e_{rr}, e_{rb}, e_2, e_{bb}, e_{br})$ can give an embedding which is disjunctive at v too. We first consider the case of $(e_1, e_{rb}, e_{rr}, e_2, e_{br}, e_{bb})$. See Fig. 7(a). The pairs of color-patterns for e_1 and e_2 which admits a disjunctive embedding with π are only $(\texttt{br}, \texttt{b}) \in \tau(u, v; G_{e_1})$ and $(\texttt{rb}, \texttt{r}) \in \tau(u, v; G_{e_2})$ (if no such pairs exist, then we terminate the computation).

As shown in Fig. 7(b), in this case, one of e_3 and e_4 (say e_3) must appear between e_{rb} and e_{rr} and the other (e_4) must appear between e_{br} and e_{bb}, satisfying $(\texttt{r}, \texttt{rb}) \in \tau(u, v; G_{e_3})$ and $(\texttt{b}, \texttt{br}) \in \tau(u, v; G_{e_4})$ (otherwise, π cannot give an embedding disjunctive at u and v). If the above pairs of color-patterns exist, then from the current cyclic ordering π, we let $\tau(u, v; G_e)$ include $(\texttt{rbr}, \texttt{brb})$, $(\texttt{rbr}, \texttt{rbr})$, $(\texttt{brb}, \texttt{rbr})$, and $(\texttt{brb}, \texttt{brb})$.

We can handle the other case, i.e., the edges $e_1, e_2, e_{rr}, e_{rb}, e_{br}, e_{bb}$ appear in the order of $(e_1, e_{rr}, e_{rb}, e_2, e_{bb}, e_{br})$ in π symmetrically (see Fig. 7(c)). If there are pairs of color-patterns $(\texttt{br}, \texttt{r}) \in \tau(u, v; G_{e_1})$, $(\texttt{rb}, \texttt{b}) \in \tau(u, v; G_{e_2})$, $(\texttt{r}, \texttt{br}) \in \tau(u, v; G_{e_3})$ and $(\texttt{b}, \texttt{rb}) \in \tau(u, v; G_{e_4})$, then π is given by $(e_1, e_{rr}, e_3, e_{rb}, e_2, e_{bb}, e_4, e_{br})$ and we let $\tau(u, v; G_e)$ include $(\texttt{rbr}, \texttt{brb})$, $(\texttt{rbr}, \texttt{rbr})$, $(\texttt{brb}, \texttt{rbr})$, and $(\texttt{brb}, \texttt{brb})$.

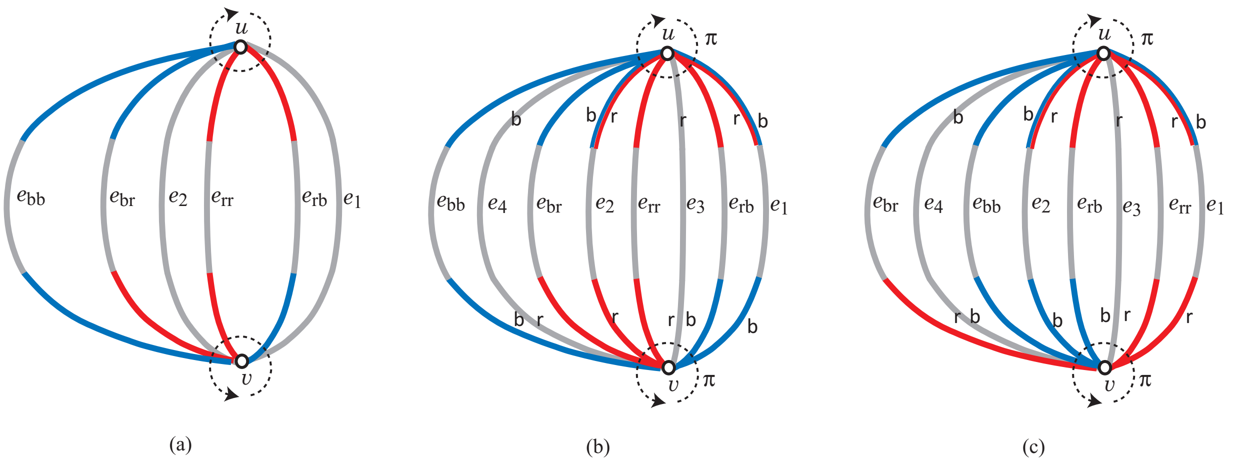

Figure 7: Illustration for the edges in the skeleton $H = \sigma^{-}(\nu)$ of a P-node ν. (a) an ordering of edges $e_1, e_2, e_{rr}, e_{rb}, e_{br}, e_{bb}$; (b) a cyclic ordering $\pi = (e_1, e_{rb}, e_3, e_{rr}, e_2, e_{br}, e_4, e_{bb})$; (c) a cyclic ordering $\pi = (e_1, e_{rr}, e_3, e_{rb}, e_2, e_{bb}, e_4, e_{br})$.

Root nodes: Finally consider the root ν^* of $\mathcal{T}$. In this case, we can use the arguments for internal R- and P-nodes.

(1) R-nodes: Let ν^* be the root R-node, and $H = \sigma(\nu^*)$. Note that $deg(w; H) \geq 3$ for all vertices $w \in V(H)$. In this case, we regard ν^* as μ in the internal R-node case, and apply the same procedure for the cases (i),(ii),(iv) and (v) to each vertex $w \in V(H)$, where case (iii) does not occur. If there is an edge $(w,x) \in E(H)$ such that $\tau'(w,x) = \emptyset$ holds after applying the procedure, then we can conclude that G has no disjunctive embedding. Otherwise, G has a disjunctive embedding.

(2) P-nodes: Let ν^* be the root P-node, $H = \sigma(\nu^*)$, and $\{u,v\} = V(H)$.

In this case, we regard ν^* as ν in the internal P-node case, and apply the same procedure for computing all cyclic orders π of the edges in $E(u;H)$ that give rise to an embedding of $G^-(\nu) = G$ which is disjunctive at u and v. If there is no such cyclic order π of the edges in $E(u;H)$, then we can conclude that G has no disjunctive embedding. Otherwise, G has a disjunctive embedding.

7 Detecting Disjunctive and Splitter-free Embeddings

This section shows how to test whether a give graph G admits a disjunctive and splitter-free embedding. By Theorem 9, we can test whether G admits a disjunctive embedding in linear time, and by Theorem 10, we can test whether G admits a splitter-free embedding in linear time. Recall that there is an instance G that admits a disjunctive embedding and a splitter-free embedding, but no disjunctive and splitter-free embedding, as we have observed in Fig. 3(a)-(b). This suggests that we have to check whether G has disjunctive and splitter-free embedding at the same time. We show that the algorithm for finding disjunctive embeddings can be modified so that the existence of disjunctive and splitter-free embeddings can be tested in linear time.

For this, we apply the following algorithm.

Step 1. Apply the algorithm in Section 4 to a given graph $G = (V, E_1 \cup E_2)$ to check whether G has a rigid splitter at an R-node. If G has a rigid splitter, then we terminate the computation concluding that G has no 2-page book embedding. If G has no rigid splitter, go to Step 2.

Step 2. Apply the algorithm in Section 5 to check whether G has a splitter-free embedding. If G has no splitter-free embedding, then we terminate the computation concluding that G has no 2-page book embedding. Consider the case where G has a splitter-free embedding. In this case, the conditions (i)-(ii) of Lemma 8 hold for all P-nodes, and a splitter-free embedding ψ_G of G is obtained by computing an embedding ζ_ν of each P-node ν under the constraints 1-4. For each P-node ν, we store the constraint used to construct such an embedding ζ_ν. Go to Step 3.

Step 3. We apply the algorithm in Section 6 to G imposing the constraint used to choose an embedding ζ_ν for a virtual edge $e = (u,v)$ that corresponds to a P-node ν when we update $\tau(u,v;G_e)$. Each of the constraints 1-4 requires the edges in $E^r(\sigma(\nu)) \cup E^{br}(\sigma(\nu)) \cup \{e_r\}$ (resp., $E^b(\sigma(\nu)) \cup E^{br}(\sigma(\nu)) \cup \{e_b\}$) appear consecutively, and the embeddings of G_e of these edges e are placed so that the r-rimmed paths (resp., the b-rimmed paths) meet each other. Whenever we compute $\tau(u,v;G_e)$ for a virtual edge $e = (u,v)$ that corresponds to a P-node ν by the procedure in Section 6, we eliminate all the pairs of color-patterns $(\alpha,\beta) \in \tau(u,v;G_e)$ that do not satisfy the constraint used for the P-node ν. Since $|\tau(u,v;G_e)| = O(1)$, we can update the set τ with the same time complexity. If $\tau(u,v;G_e)$ becomes empty for some $(u,v) \in E(\sigma(\nu))$ during the execution of this algorithm, then we terminate the computation concluding that G has no disjunctive and splitter-free embedding. Otherwise if we can construct a disjunctive embedding ψ at the root ν^*, then ψ is also splitter-free.

For example, we obtain $\tau(u,v;G_{e_c}) = \{(\texttt{rbr},\texttt{rbr}), (\texttt{brb},\texttt{brb})\}$ for the virtual edge e_c which corresponds to the P-node ν_c in Fig. 11, before we delete the pairs of color-patterns that do not satisfy the constraint 1 which requires that the edges e_e and e_f appear consecutively in the embedding ζ of H, and the r-rimmed paths of the embeddings $\psi_{G_{e_e}}$ and $\psi_{G_{e_f}}$ of the graphs G_{e_e} and G_{e_f} need to meet each other. In this case, none of the pairs of color-patterns in $\tau(u,v;G_{e_c})$ satisfies the constraint, and $\tau(u,v;G_{e_c})$ becomes empty, indicating that the graph G_1 in Fig. 3(a) has no disjunctive and splitter-free embedding.

Thus we have the next result.

Theorem 11 *Let $G = (V, E_1 \cup E_2)$ be a planar graph with $E_1 \cap E_2 = \emptyset$. Then whether G has a 2-page book embedding with the partition E_1 and E_2 or not can be tested in linear time.* □

8 Clustered Graph Planarity

A *clustered graph* $C = (G, T)$ consists of an undirected graph $G = (V, E)$ (called the *underlying graph*) and a rooted tree $T = (\mathcal{V}, \mathcal{A})$ (called the *inclusion tree* of C), where the leaves of T are exactly the vertices of G [12]. Each node ν of T represents a *cluster* $V(\nu)$, a subset of the vertices of G that are leaves of the subtree rooted at ν. Let $G(\nu)$ denote the subgraph induced from G by $V(\nu)$. A cluster corresponding to a leaf node (resp., an internal node and the root) in T is called a *leaf cluster* (resp., an *internal cluster* and the *root cluster*). An edge e is called *inter-cluster* edge if the end vertices of e belong to different internal clusters. The clustering hierarchy is called *flat* if all internal clusters are at the same level.

In a *drawing* of a clustered graph $C = (G, T)$, graph G is drawn as points for vertices and curves for edges in the plane as usual. For each node ν of T, the cluster is drawn as a simple closed region $R(\nu)$ enclosed by a simple closed curve such that the drawing of $G(\nu)$ is completely contained in the interior of $R(\nu)$, the regions for all sub-clusters of ν are completely contained in the interior of $R(\nu)$, and the regions for all other clusters are completely contained in the exterior of $R(\nu)$. A clustered graph is *compound planar* (*c-planar*) if it admits a c-planar drawing without edge crossings or *edge-region crossings* (i.e., the drawing of e crosses the boundary of region R more than once).

The computational complexity status of the c-planarity testing for clustered graphs is one of the main open problems in Graph Drawing (see [4] for a brief survey). Polynomial-time testing algorithms have been found only for several classes of clustered graphs in which the structures on the underlying graphs and cluster structures are both restricted. The followings are such classes:

- c-connected clustered graphs, in which each cluster induces a connected subgraph of the underlying graph [6, 12].
- Completely connected clustered graphs, that are c-connected clustered graphs such that the complement of each cluster also induces a connected subgraph of the underlying graph [3].
- Almost connected clustered graphs, in which either (i) all nodes corresponding to non-connected clusters are on the same path in the cluster hierarchy, or (ii) for each non-connected cluster its parent and all its siblings are connected [14].
- Cycles of clusters, in which the hierarchy is flat, the underlying graph is a cycle, and the clusters are arranged in a cycle [6].
- Clustered cycles, that are clustered graphs in which the hierarchy is flat, the underlying graph is a simple cycle, and the clusters are arranged into an embedded plane graph [5].
- Flat clustered graphs with small faces, that are flat clustered graphs with a fixed plane embedding of the underlying graph and each face of the embedding has at most five vertices [7].
- Clustered graphs with small faces, that are clustered graphs such that the underlying graph is a triconnected planar graph, each face contains at most four vertices, and every two internal clusters are disjoint [16].

However, even for a flat clustered graph with exactly two internal clusters, which is the simplest nontrivial case of cluster structures, the complexity status is not known. We prove that this fundamental case of the c-planarity testing problem with arbitrary underlying graphs can be solved in linear time by reducing the problem to the 2-page book embedding problem with partitioned edge sets. For example, Figure 8(a) and (b) show a c-planar clustered graph C_1 with two internal clusters $\nu_1 = \{v_1, v_2\}$ and $\nu_2 = \{v_3, v_4, v_5\}$ and its c-planar drawing, while Figure 8(c) shows a non-c-planar clustered graph C_2 with two internal clusters $\nu_1 = \{v_1, v_2\}$ and $\nu_2 = \{v_3, v_4, v_5\}$.

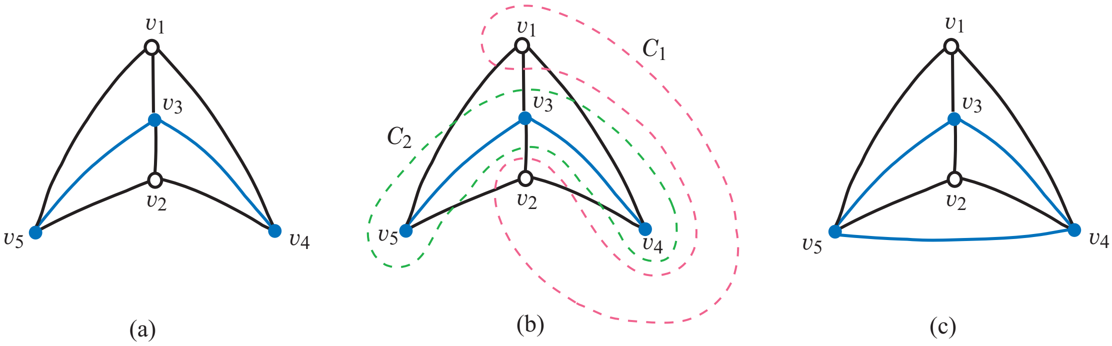

Figure 8: (a) A clustered graph C_1 with internal clusters $\nu_1 = \{v_1, v_2\}$ and $\nu_2 = \{v_3, v_4, v_5\}$; (b) A c-planar drawing of C_1; (c) A non-c-planar clustered graph C_2 with internal clusters $\nu_1 = \{v_1, v_2\}$ and $\nu_2 = \{v_3, v_4, v_5\}$.

8.1 Reduction from 2-page Book Embedding Problem

We first observe that the 2-page book embedding problem with partitioned edge sets can be reduced to the c-planarity testing problem with flat clustered graphs with two internal clusters. Let $G = (V, E_1 \cup E_2)$ be a graph with a partitioned edge set E_1 and E_2. We replace each vertex $v \in V$ with two vertices v_1 and v_2 changing the end vertices of each edge $(u, v) \in E_i$ to (u_i, v_i), and join the two new vertices v_1 and v_2 of each vertex $v \in V$ with a new edge $e_v = (v_1, v_2)$. Let $G' = (V_1 \cup V_2, E_1 \cup E_2 \cup E_V)$ be the resulting graph, where V_i denotes the set of vertices v_i, $v \in V$, and E_V denotes the set of the new edges e_v, $v \in V$. By defining internal clusters ν_i with $V(\nu_i) = V_i$, $i = 1, 2$, we obtain a flat clustered graph $C = (G', T)$ with two internal clusters.

Lemma 12 *A graph $G = (V, E_1 \cup E_2)$ with a partitioned edge set admits a 2-page book embedding if and only if the flat clustered graph C defined above is c-planar.*

Proof. Only if part: Assume that G has a 2-page book embedding π, i.e., a planar embedding ψ with a separating curve λ. Then it is easy to see that the transformation from G to G' preserves the planarity of ψ and λ still separates the edges in E_1 from those in E_2 in the resulting plane embedding of G'. Thus, C admits a c-planar embedding without edge crossings or edge-region crossings.

If part: Assume that C is c-planar; i.e., C admits a c-planar embedding ψ without edge crossings or edge-region crossings. Let λ be the boundary of the cluster region $R(\nu_1)$, and ψ_G be the embedding of G obtained from the plane embedding for G' by contracting each edge $e_v \in E_V$. We see that ψ_G remains planar and λ visits all the vertices in V. Since a plane embedding ψ_G of G has a separating curve λ, G has a 2-page book embedding. □

8.2 Reduction to 2-page Book Embedding Problem

We now show that the c-planarity testing problem with flat clustered graphs with two internal clusters can be reduced to the 2-page book embedding problem with partitioned edge sets. Let $C = (G, T)$ be a clustered graph such that the underlying graph G is a planar graph and the inclusion tree T is flat.

For each internal cluster ν, a subgraph H of $G(\nu)$ is called *irrelevant* if G has a cut vertex $w \in V(H)$ such that $H - \{w\}$ is a connected component of $G - \{w\}$. The vertices in $H - \{w\}$ are called *irrelevant*. For example, the subgraphs H_1, H_2, H_3 and H_4 of the clustered graph C in Fig. 9(b) are irrelevant. We can identify all irrelevant subgraphs in linear time by the depth-first search on G.

Let $C_1 = (G_1, T_1)$ be the clustered graph obtained from C by removing the vertices together with the incident edges. See Fig. 9(b) for the clustered graph C_1 of the clustered graph C in Fig. 9(a).

Lemma 13 *A clustered graph C is c-planar if and only if the clustered graph C_1 is c-planar.*

Proof. The only if part is trivial. To show the if part, we assume that C_1 admits a c-planar drawing ψ_{C_1} without edge crossings or edge-region crossings. Since each irrelevant subgraph H of a graph $G(\nu)$ has a vertex w such that the graph $H - \{w\}$ is adjacent to w, we can add a planar drawing ψ_H of each H to ψ_{C_1} so that the resulting drawing remains a c-planar drawing without edge crossings or edge-region crossings. □

Let $\nu_1, \nu_2, \ldots, \nu_r$ be the internal clusters of C_1. For each ν_i, let G_1^i be the graph obtained from G_1 by contracting the set of vertices not in ν_i into a single vertex z_i, and ψ_i be a plane embedding of G_1^i such that z_i appears on the boundary of ψ_i (if G_1^i is not planar then C_1 is not c-planar). Let Γ^i be the embedding induced from G_1^i by the vertices in ν_i, and call a vertex or an edge not on the outer boundary of Γ^i *enclosed*. Note that each enclosed vertex v is in the interior of a cycle f of some edges in the outer boundary of Γ^i. Hence the embedding obtained from Γ^i by removing the enclosed vertices in ν_i is an outerplane embedding, and it will be an outerplane embedding of a cactus after removing all enclosed edges. We call a block in the cactus a *leaf-block* if it has at most one vertex shared with other blocks in the cactus.

Let $C_2 = (G_2, T_2)$ be the clustered graph obtained from C_1 by removing the enclosed vertices/edges in C_1 over all ν_i. Hence the vertices in each $V(\nu_i)$ induce a cactus H_i, where each leaf-block in the cactus H_i contains a vertex that is adjacent to a vertex in $V(\nu_j)$ with some other cluster ν_j ($j \neq i$), since otherwise $G(\nu_i)$ in C_1 would have an irrelevant subgraph.

See Fig. 9(b), the vertices x_1 and x_2. Note that each internal cluster ν now induces an outerplanar graph $G(\nu)$ in C_2.

Lemma 14 *The clustered graph C_1 is c-planar if and only if the clustered graph C_2 is c-planar.*

Proof. The only if part is trivial. Assume that C_1 has at least one inter-cluster edge (otherwise C_1 and C_2 are c-planar if G is planar). To show the if part, we assume that C_2 admits a c-planar drawing ψ_{C_2} without edge crossings or edge-region crossings, where without loss of generality that the boundary of G_2 in ψ_{C_2} contains at least one inter-cluster edge. Consider how the cactus $G(\nu)$ in G_2 is embedded in c-planar drawing ψ_{C_2}. Since the boundary of G_2 contains an inter-cluster edge and the cactus $G(\nu)$ in G_2 has no irrelevant subgraph, teach cycle of the cactus $G(\nu)$ appears as a facial cycle without including any edge/vertex in its interior (otherwise the interior contains a leaf-block of $G(\nu)$ and an edge joining the block and some other cluster ν' would create a crossing with an edge in the cycle). Hence we can put back all vertices/edges enclosed by f to ψ_{C_2} without creating edge crossings or edge-region crossings. Hence we can obtain a c-planar drawing ψ_{C_1} of C_1 without edge crossings or edge-region crossings. □

Assume that C_2 has exactly two internal clusters, say ν_1 and ν_2, where $V(G_2) = V(\nu_1) \cup V(\nu_2)$ holds. We subdivide each inter-cluster edge $e = (u, v)$ with a new vertex w; i.e., $e = (u, v)$ is replaced with two edges (u, w) and (w, v). Let W be the set of the new vertices introduced to subdivide all inter-cluster edges in C_2, and $G_3 = (V_3 = V(\nu_1) \cup V(\nu_2) \cup W, E_1 \cup E_2)$, where E_i denotes the set of edges incident to vertices in $V(\nu_i)$. See Fig. 10(a).

Lemma 15 *Let $C = (G, T)$ be a clustered graph such that the underlying graph G is a planar graph and the inclusion tree T is flat and has exactly two internal clusters ν_1 and ν_2. Then C is c-planar if and only if the graph $G_3 = (V_3 = V(\nu_1) \cup V(\nu_2) \cup W, E_1 \cup E_2)$ defined above admits a 2-page book embedding with the partition E_1 and E_2 of $E(G_3)$.*

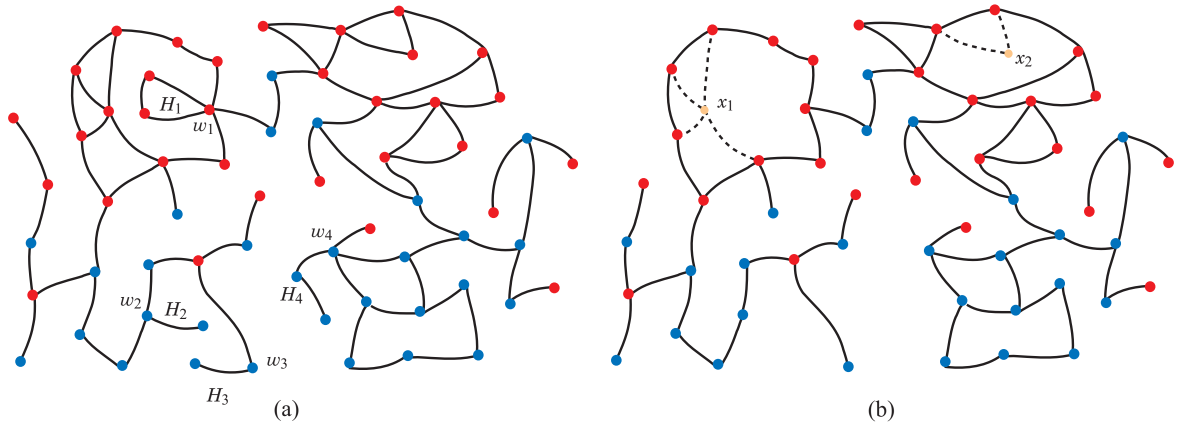

Figure 9: (a) An example of a flat clustered graph $C = (G, T)$; (b) a plane embedding of the flat clustered graph $C_1 = (G_1, T_1)$ obtained by removing the irrelevant blocks in C, where the clustered graph from C_1 by removing the enclosed vertices x_1 and x_2 gives $C_2 = (G_2, T_2)$.

Proof. By Lemmas 13 and 14, it suffices to show that C_2 is c-planar if and only if the graph $G_3 = (V_3 = V(\nu_1) \cup V(\nu_2) \cup W, E_1 \cup E_2)$ admits a 2-page book embedding.

Only if part: Assume that C_2 admits a c-planar drawing ψ_{C_2} without edge crossings or edge-region crossings, where without loss of generality that the boundary of G in ψ_{C_2} contains at least one inter-cluster edge and all vertices in each outerplanar graph $G(\nu_i)$ appear the outer boundary of the drawing of $G(\nu_i)$. See Fig. 10(a).

Let ψ_{G_3} be the plane embedding of G_3 obtained from the plane embedding of G_2 in ψ_{C_2}, by replacing each inter-cluster edge (u, v) with (u, w) and (w, v).

In ψ_{C_2}, $V(\nu_1)$ is contained in a region $R(\nu_1)$. Let B be the boundary of $R(\nu_1)$, which separates the plane into two regions $R_1 = R(\nu_1)$ and R_2. Since B separates $V(\nu_1)$ and $V(\nu_2)$ and B intersects each inter-cluster edge of C_2 exactly once, we can assume without loss of generality that B visits each of all vertices in W exactly once, enclosing all red edges in its region R_1. Since all vertices of each outerplanar graph $G(\nu_i)$ appear on its outer boundary, we can draw a curve L_v from each vertex $v \in V(\nu_1) \cup V(\nu_2)$ to a point p_v on B so that no two L_v and $L_{v'}$ intersect each other (if they intersect, we can switch p_v and $p_{v'}$ to eliminate the intersection). See Fig. 10(b). This implies that we can modify B into a simple closed curve λ that visits each of all vertices in $V_3 = V(\nu_1) \cup V(\nu_2) \cup W$ exactly once while separating the red edges from the blue edges. As observed, such separating curve λ gives a 2-page book embedding of G_3.

If part: Assume that G_3 has a 2-page book embedding in which all vertices are placed on a spine and the red edges (resp., the blue edges) appear above (resp., below) the spine. We enclose all the red edges by a closed curve λ obtained from the spine and a curve joining the first and last vertices. Let R_1 be the region enclosed by λ. We can modify R_1 into a region R so that all vertices in $V(\nu_1)$ are included in R while all the vertices in W are on the boundary of R. This is possible because only red edges are incident to each vertex $v \in V(\nu_1)$ and all the red edges are contained in R_1. Hence R is a cluster region for ν_1, and we can obtain a cluster region R' for ν_2 in the complement area of R. Therefore, C_2 admits a c-planar drawing without edge crossings or edge-region crossings. □

The following theorem summarizes the main result of this Section.

Theorem 16 *Let $C = (G, T)$ be a clustered graph such that the underlying graph G is a planar graph and the inclusion tree T is flat and has exactly two internal clusters. Then whether C is c-planar or not can be tested in linear time.* □

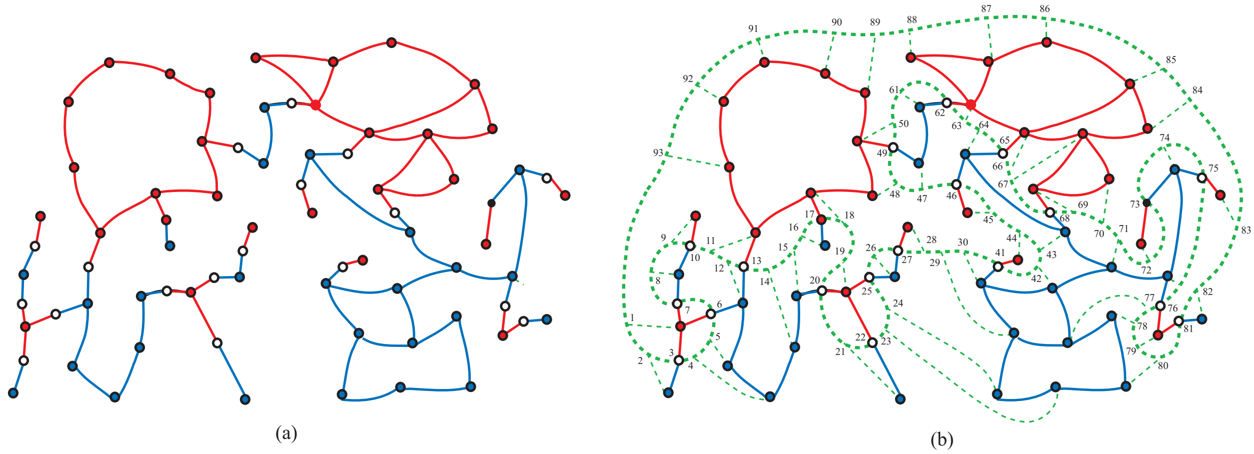

Figure 10: (a) The graph G_3 obtained from $C_2 = (G_2, T_2)$ in Fig. 9; (b) A simple closed curve B that separates the edges in E_1 from the edges in E_2.

References

[1] F. Bernhart and P.C. Kainen, The book thickness of a graph, J. Combin. Theory Ser. B 27(3), 320-331, 1979.

[2] F. R. K. Chung, F. T. Leighton, and A. L. Rosenberg, Embedding graphs in books: A graph layout problem with applications to VLSI design, SIAM J. Algebraic Discrete Methods, 8, pp. 33-58, 1986.

[3] S. Cornelsen and D. Wagner, Completely connected clustered graphs. Journal of Discrete Algorithms, 4(2), pp. 313-323, 2006.

[4] P. F. Cortese and G. Di Battista, Clustered planarity. In Symposium on Computational Geometry, pp. 32-34, 2005.

[5] P. F. Cortese, G. Di Battista, M. Patrignani, and M. Pizzonia, On embedding a cycle in a plane graph. In Graph Drawing, pp. 49-60, 2005.

[6] P. F. Cortese, G. Di Battista, M. Patrignani, and M. Pizzonia, Clustering cycles into cycles of clusters. Journal of Graph Algorithms and Applications, 9(3), pp. 391-413, 2005.

[7] G. Di Battista and F. Frati, Efficient c-planarity testing for embedded flat clustered graphs with small faces, Graph Drawing 2007, Lecture Notes in Computer Science 4875, pp. 291-302, 2007.

[8] G. Di Battista and R. Tamassia, On-line planarity testing, SIAM J. on Comput. 25(5), pp. 956-997, 1996.

[9] G. Di Battista and R. Tamassia, On-line maintenance of triconnected components with SPQR-trees, Algorithmica, 15, pp. 302-318, 1996.

[10] V. Dujmović and D. R. Wood, On linear layouts of graphs Discrete Math. Theor. Comput. Sci, 6, pp. 339-358, 2004.

[11] V. Dujmović and D. R. Wood, Stacks, queues and tracks: Layouts of graph subdivisions, Discrete Math. Theor. Comput. Sci, 7, pp. 155-202, 2005.

[12] P. Eades, Q. Feng, X. Lin and H. Nagamochi, Straight-line drawing algorithms for hierarchical graphs and clustered graphs, Algorithmica 44, pp. 1-32, 2006.

[13] M. T. Goodrich, G. S. Lueker, and J. Z. Sun, C-planarity of extrovert clustered graphs. In Graph Drawing, pp. 211-222, 2005.

[14] C. Gutwenger, M. Jünger, S. Leipert, P. Mutzel, M. Percan, and R. Weiskircher, Advances in c-planarity testing of clustered graphs. In Graph Drawing, pp. 220-235, 2002.

[15] J. E. Hopcroft and R. E. Tarjan, Dividing a graph into triconnected components, SIAM J. on Comput., 2, pp. 135-158, 1973.

[16] E. Jelínková, J. Kára, J. Kratochvíl, M. Pergel, O. Suchý, and T. Vyskocil, Clustered planarity: small clusters in eulerian graphs, Graph Drawing 2007, Lecture Notes in Computer Science 4875, pp. 303-314, 2007.

[17] A. L. Rosenberg, The Diogenes approach to testable fault-tolerant arrays of processors, IEEE Trans. Comput. C-32, pp. 902-910, 1983.

[18] W. T. Tutte, Graph Theory, Encyclopedia of Mathematics and Its Applications, Vol. 21, Addison-Wesley, Reading, MA, 1984.

[19] M. Yannakakis, Embedding planar graphs in four pages, J. Comput. System Sci. 38, pp. 36-67, 1989.

[20] A. Wigderson, The complexity of the Hamiltonian circuit problem for maximal planar graphs, Technical Report 298, EECS Department, Princeton University, 1982.

Appendix 1: SPQR tree

We review the definition of *triconnected components* [15] (or *3-blocks*) and a variation of the *SPQR tree* [8, 9] of a biconnected graph.

First we review the definition of triconnected components [15]. If G is triconnected, then G itself is the unique triconnected component of G. Otherwise, let u, v be a cut-pair of G. We split the edges of G into two disjoint subsets E_1 and E_2, such that $|E_1| > 1$, $|E_2| > 1$, and the subgraphs G_1 and G_2 induced by E_1 and E_2 only have vertices u and v in common. Form the graph G'_1 from G_1 by adding an edge (called a *virtual edge*) between u and v that represents the existence of the other subgraph G_2; similarly form G'_2. We continue the splitting process recursively on G'_1 and G'_2. The process stops when each resulting graph reaches one of three forms: a triconnected simple graph, a set of three multiple edges (a triple bond), or a cycle of length three (a triangle).

The triconnected components of G are obtained from these resulting graphs:

- a triconnected simple graph;
- a *bond*, formed by merging the triple bonds into a maximal set of multiple edges;
- a *polygon*, formed by merging the triangles into a maximal simple cycle.

The triconnected components of G are unique. See [15] for further details.

One can define a tree structure, sometimes called as the *3-block tree*, using triconnected components as follows. The nodes of the 3-block tree are the triconnected components of G. The edges of the 3-block tree are defined by the virtual edges, that is, if two triconnected components have a virtual edge in common, then the nodes that represent the two triconnected components in the 3-block tree are joined by an edge that represents the virtual edge.

There are many variants of the 3-block tree in the literature; the first was defined by Tutte [18]. In this paper, we use the terminology of the *SPQR tree*, as defined by di Battista and Tamassia [8, 9]. We now briefly review this terminology.

Each node ν in the SPQR tree is associated with a graph $G = (V, E)$ called the *skeleton* of ν, denoted by $\sigma(\nu) = (V_\nu, E_\nu)$ ($V_\nu \subseteq V$), which corresponds to a triconnected component. There are four types of nodes in the SPQR tree. The node types and their skeletons are:

1. Q-node: the skeleton consists of two vertices connected by two multiple edges. Each Q-node corresponds to an edge of the original graph.

2. S-node: the skeleton is a simple cycle with at least three vertices (this corresponds to a polygon triconnected component).

3. P-node: the skeleton consists of two vertices connected by at least three edges (this corresponds to a bond triconnected component).

4. R-node: the skeleton is a triconnected graph with at least four vertices.

The SPQR tree as developed by di Battista and Tamassia is a data structure with efficient operations. In this paper, we use the SPQR tree only as a convenient way to traverse the 3-blocks of a biconnected graph. In fact, we use a slight modification of the SPQR tree: we omit the Q-nodes and we root the tree as described below. We will refer the (modified) SPQR tree as the SPQR tree throughout this paper.

In this paper, a given graph G can have multiple edges only when they are a pair of red and blue edges with the same end vertices, and we do not explicitly use Q-nodes. A pair of red and blue edges with the same end vertices u and v will be treated as a virtual edge (resp., as two real edges) when both u and v appear as vertices of a skeleton $\sigma(\nu)$ of an R- or S-node ν (resp., a P-node ν).

The SPQR tree is unique [8, 9]. We treat the SPQR tree of a graph G as a rooted tree $\mathcal{T}$ by choosing an arbitrary node ν^* as its root.

Some further notation for the SPQR tree is required. Suppose that G is a biconnected (but not triconnected) planar graph, and $\mathcal{T}$ is the rooted SPQR tree of G. Let ν be a nonroot node in $\mathcal{T}$, and μ be the parent of ν. The graph $\sigma(\mu)$ has one virtual edge e in common with $\sigma(\nu)$. The edge e is the *parent virtual edge* in $\sigma(\nu)$, and it is a *child virtual edge* in $\sigma(\mu)$. For each nonroot node v of the SPQR tree, $\sigma(\nu)$ has precisely one parent virtual edge, and for each non-leaf node ν', $\sigma(\nu')$ has at least one child virtual edge. A node ν which is neither the root or a leaf node is called an *internal* node.

We denote the graph formed from $\sigma(\nu)$ by deleting its parent virtual edge as $\sigma^-(\nu)$, if ν is not the root of $\mathcal{T}$. If $\sigma(\nu)$ is a nonroot R-node, then $\sigma^-(\nu)$ is internally triconnected. The union of the graphs $\sigma^-(\mu)$ for all descendants μ of ν, including ν itself, is denoted by $G^-(\nu)$; i.e., $G(\nu)$ is the graph obtained from G by inducing the vertex set $\cup\{V_\mu \mid \text{ descendants } \mu \text{ of } \nu\}$.

For example, Figs. 11 and 12 illustrate the SPQR trees of the graphs G_1 and G_2 in Fig. 3(a) and (c), respectively.

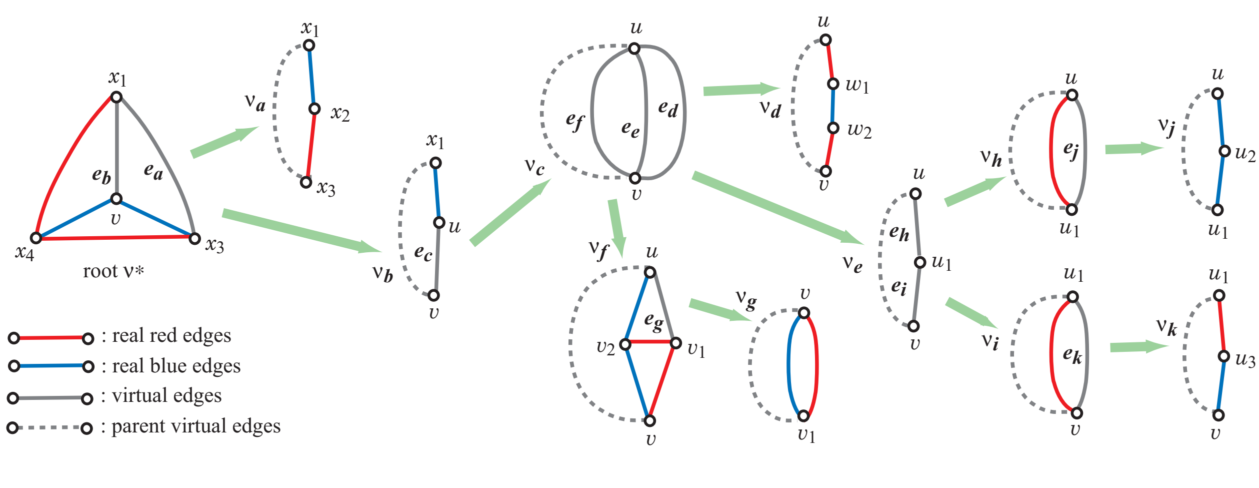

Figure 11: The SPQR tree $\mathcal{T}$ of the graph G_1 in Fig. 3(a), where ν^* is the root R-node, $\nu_a, \nu_b, \nu_d, \nu_g, \nu_e, \nu_j$ and ν_k are the S-nodes, ν_c, ν_g, ν_h and ν_i are the P-nodes, and ν_f is the non-root R-node.

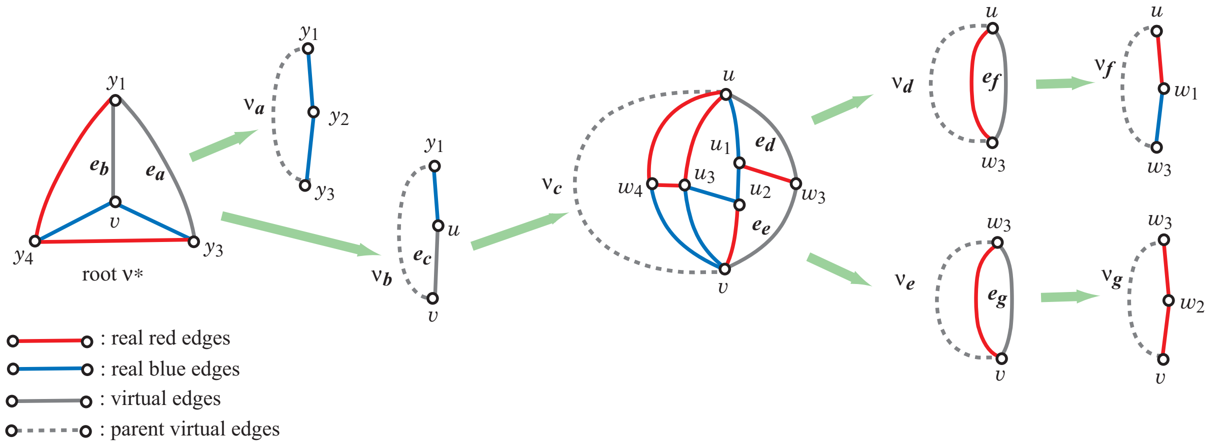

Figure 12: The SPQR tree $\mathcal{T}$ of the graph G_2 in Fig. 3(c), where ν^* is the root R-node, ν_a, ν_b, ν_f and ν_g are the S-nodes, ν_d and ν_e are the P-nodes, and ν_c is the nonroot R-node.