FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

MASTER THESIS

Bc. Martin Hora

The complexity of constrained graph drawing

Computer Science Institute of Charles University

Supervisor of the master thesis:	doc. RNDr. Vít Jelínek, Ph.D.
Study programme:	Computer Science
Study branch:	Theoretical Computer Science

Prague 2019

I declare that I carried out this master thesis independently, and only with the cited sources, literature and other professional sources.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In .... date .... signature of the author

I would like to thank my supervisor doc. RNDr. Vít Jelínek, Ph.D. for his guidance and valuable advice. He was always ready to listen to my ideas and his comments greatly improved the readability of this thesis.

Title: The complexity of constrained graph drawing

Author: Bc. Martin Hora

Institute: Computer Science Institute of Charles University

Supervisor: doc. RNDr. Vít Jelínek, Ph.D.,
Computer Science Institute of Charles University

Abstract: A labeled embedding of a planar graph G is a pair $(\mathcal{G}, g)$ consisting of a planar drawing $\mathcal{G}$ of G and a function g assigning labels (colors) to the faces of $\mathcal{G}$. We study the problem of Embedding Restriction Satisfiability (ERS) that investigates whether a given graph has a labeled embedding satisfying a provided set of conditions. ERS is a relatively new problem, so not much is known about it. Nevertheless, it has great potential. It generalizes several problems looking for a particular drawing of a planar graph, such as the problem of Partially Embedded Planarity. Therefore, ERS may become a focal point in the area of graph drawing. In this thesis, we examine the computational complexity of ERS. We show that ERS is NP-complete. After that, we look at the complexity of some specific classes of its instances. We try to locate the boundary between the NP-complete and the polynomial variants of the problem.

Keywords: planar graphs, partially embedded graphs, constrained planarity, computational complexity

Contents

1. Introduction

The planar graphs belong among the most studied classes of graphs. After all, one of the oldest challenges in graph theory is to recognize whether a given graph is planar. This problem was addressed in 1930 by Kuratowski [14], who proved that a graph is planar if and only if it does not contain a subdivision of the complete graph K_5 or the complete bipartite graph $K_{3,3}$. Kuratowski's characterization of the planar graphs is a very interesting result from the point of view of computational complexity. The forbidden subgraphs can be found in polynomial time, therefore planarity testing is in the complexity class P. In fact, there are even algorithms running in linear time. The first of them was proposed in 1974 by Hopcroft and Tarjan [10]. And in 2004 Boyer and Myrvold [2] published a linear algorithm, which not only tests planarity, but it also produces a certificate supporting its decision. It either generates a planar embedding, or it provides a subgraph which is a subdivision of K_5 or $K_{3,3}$.

Hence, we have an asymptotically optimal algorithm constructing a planar embedding of a planar graph. However, some graphs have a huge amount of planar embeddings. When we are drawing such a graph in practice, then we often do not want to take the first embedding we are able to construct, but we try to select one with good properties. For instance, imagine we are designing a CPU and we know that there are several components producing a lot of heat. Then, we would like to place these components far apart to avoid overheating the processor. In general, we look for an embedding satisfying some additional conditions. This type of problems is known as constrained planarity.

Arguably the most famous problem of this category is Clustered Planarity [5]. We are given a planar graph G and a hierarchy of clusters of vertices of G. The goal is to find a planar embedding of G where for each cluster C there is a topological cycle γ_C separating the vertices of C from the others and crossing every edge of G at most once. So for each cluster, its vertices are "close" to each other. Even though the problem has been intensively studied since its introduction in 1995, we still do not know how hard it is. It is one of the candidates for an NP-intermediate problem, i.e. a problem that is neither in the class P nor NP-complete.

Another example is the Partially Embedded Planarity problem [1]. Here, the input consists of a graph G, its subgraph H and a planar embedding $\mathcal{H}$ of H. The task is to extend $\mathcal{H}$ into an embedding of G. The problem was proposed by Angelini et al., who also published a linear-time algorithm solving it. However, their algorithm is quite complicated. I studied this problem in my bachelor thesis [11] and I found an alternative linear algorithm that is easier to implement.

In the bachelor thesis, I came up with an object I called a labeled embedding of a graph. It is formed by a planar embedding of the graph and a function assigning a label to each face of the embedding. Then, I defined a structure called embedding restriction that puts constraints on labeled embeddings. It allows us to select for each vertex a subset of its incident edges and fix their ordering in the embedding. Next, it enables us to prescribe labels for incident faces of each edge. And finally, we can mark some edges as transparent. A transparent edge must have the same label on both sides. This leads to a problem, called Embedding Restriction Satisfiability (ERS), deciding whether a graph has a labeled embedding satisfying a given embedding restriction.

Further, I showed that each instance of Partially Embedded Planarity can be reduced in linear time into an instance of the new problem. These very specific instances of ERS can be then solved in linear time. Nevertheless, this is so far the only known result about the computational complexity of the ERS problem.

The goal of the thesis is to rectify the deficiency. We thoroughly investigate the complexity of ERS. First, we show that the problem is NP-complete in general setting. Later, we derive several variants of ERS by placing extra conditions on the embedding restriction structure. Our intention is to find the boundaries where the problem stops being NP-hard and where it becomes polynomial. We want to narrow the gap between the polynomial and NP-complete instances as much as possible.

The thesis has three chapters. The second chapter contains the formal definition of Embedding Restriction Satisfiability including the reduction from the Partially Embedded Planarity. Also, several useful data structures such as the SPQR-trees are described there. The third chapter covers the NP-complete variants of the problem. Finally in the fourth chapter, we look at the instances that can be solved in polynomial time.

2. Preliminaries

In this chapter, we formally define the problem of Embedding Restriction Satisfiability. We present several of its variations and investigate their mutual relations. We also show a few interesting techniques allowing us to recognize some unsatisfiable instances. And finally, we describe the SPQR decomposition of biconnected graphs, which plays a crucial role in the polynomial algorithms of chapter 4.

Before we get into action, we need to establish some notations and agree on several conventions. If not mentioned otherwise, all the graphs in this thesis are undirected multigraphs without self-loops. Let G be a graph, then $V(G)$ denotes the set of vertices of G and $E(G)$ is the multiset of its edges. When we write $e = \{u, v\}$ for an edge $e \in E(G)$, then we mean that the edge e connects the vertices u and v, but there can still be more parallel edges between these two vertices.

Despite the fact that we work with undirected graphs, it is sometimes useful to consider the edges to be oriented. For instance, it enables us to easily distinguish the two incident faces of an edge in a planar embedding. We can refer to the left and right incident face. So for convenience, we presume that every edge has a fixed default orientation. And if necessary, the default direction can be overridden. For example, by putting $e = (u, v)$ we force the orientation from u to v.

2.1 Cyclic sequences

The cyclic sequences are arguably the most frequently used structure of the thesis. For example, we need them to describe the ordering of incident edges for each vertex in a planar embedding.

Definition 1. *Let X be a set and n be a non-negative integer. Further, let $\doteq$ be the equivalence on the sequences of length n containing elements of X such that $(x_0, x_1, \ldots, x_{n-1}) \doteq (x'_0, x'_1, \ldots, x'_{n-1})$ iff there exists $k \in \{0, 1, \ldots, k-1\}$ such that for each $i \in \{0, 1, \ldots, k-1\}$ $x_i = x'_{((i+k) \mod n)}$. The* cyclic sequences *of X of length n are the equivalence classes of $\doteq$. The members of an equivalence class φ of $\doteq$ are called the* rotations *of φ.*

Definition 2. *Let φ and ψ be two cyclic sequences. Then, ψ is a* cyclic subsequence *of φ if ψ can be derived from φ by removing some or no elements.*

For instance, (a, t, a, l) is a cyclic subsequence of (a, l, p, h, a, b, e, t), but (l, a, l) is not.

A typical problem we encounter is verifying that a cyclic sequence ψ is a subsequence of a cyclic sequence φ. One possible approach is to open φ into a linear sequence φ_{lin} and then test for each rotation of ψ whether it is a subsequence of φ_{lin}. The time complexity of this algorithm is $\mathcal{O}(\text{length}(\psi) \cdot \text{length}(\varphi))$. However, we can do better for some more specific cyclic sequences.

Definition 3. *Let φ be a cyclic sequence of elements of a set Z and let X, Y be non-empty disjoint subsets of Z. Then, φ is* (X, Y)-crossing *if there exist*

$x, x' \in X$ and $y, y' \in Y$ such that (x, y, x', y') is a subsequence of φ. If φ is not (X, Y)-crossing, then we say that φ is (X, Y)-non-crossing. *Further, if $X = \{x\}$ and $Y = \{y\}$, then we prefer the notation (x, y)-(non-)crossing.*

It is possible to check that a cyclic sequence φ is (x, y)-crossing in linear time with respect to the length of φ because the subsequence (x, y, x, y) has a constant length. If we want to determine whether φ is (X, Y)-crossing for some disjoint sets X and Y, then we can first make a copy φ' of φ replacing all the elements of X by x and the elements of Y by y for some $x \in X$, $y \in Y$, and then we just test that φ' is (x, y)-crossing.

We can even check in linear time that a cyclic sequence is (x, y)-non-crossing simultaneously for all possible pairs of distinct x and y.

Algorithm 1: An algorithm [1] verifying that a cyclic sequence is (x, y)-non-crossing for each pair of distinct elements x, y.

```
input : A cyclic sequence φ.
function test_non_crossing(φ):
    foreach x in φ : total[x] ← 0 ; visited[x] ← 0 ;
    foreach x in φ : total[x] += 1 ;
    stack S;
    foreach x in φ, starting arbitrarily, following the cyclic sequence φ :
        visited[x] += 1;
        if visited[x] = 1 : S.push(x);
        else if S.top() ≠ x : return false, x and S.top() are crossing;
        if visited[x] = total[x] : S.pop();
    return true;
```

Lemma 4 (Angelini et al. [1], Lemma 4.9)**.** *Let φ be a cyclic sequence containing elements of a set X. The function* `test_non_crossing(φ)` *verifies in linear time with respect to the length of φ that φ is (x, y)-non-crossing for each pair of distinct items $x, y \in X$.*

The order in which the elements of φ are popped from the stack S in Algorithm 1 has interesting properties. For example, it allows us to test subsequences in linear time.

Definition 5. *Let X be a set and φ a cyclic sequence of elements of X such that φ is (x, y)-non-crossing for each pair of distinct items $x, y \in X$. A sequence ψ of elements of X is an* elimination ordering *of φ if there exists a computation of the function* `test_non_crossing(φ)` *such that ψ is the order in which the items are popped from the stack S.*

Lemma 6. *Let ψ be an elimination ordering of a cyclic sequence φ and let x be an element of ψ. Next, let φ' be the cyclic sequence obtained from φ by removing all the elements preceding x in ψ. Then the occurrences of x in φ' form a continuous interval.*

Lemma 7. *If φ is a cyclic sequence that is (x, y)-non-crossing for each pair of distinct x, y, then we can test whether a cyclic sequence ψ is a subsequence of φ in time $\mathcal{O}(length(\varphi))$.*

Algorithm 2: An algorithm verifying that a cyclic sequence ψ is a subsequence of a cyclic sequence φ which is (x, y)-non-crossing for each pair of distinct elements x, y.

input : A cyclic sequence ψ, and a cyclic sequence φ which is (x, y)-non-crossing for each distinct x and y.

function `NC_subsequence`(ψ, φ):
- **if** ψ *is empty* **: return** true;
- **if** ψ *is longer than* φ **: return** false;
- **foreach** x *in* ψ **do** count[x] $\leftarrow 0$;
- **foreach** x *in* ψ **do** count[x] $+= 1$;
- $\beta \leftarrow$ an elimination ordering of φ;
- **foreach** x *in* β*, following the ordering* **do**
  - **if** *count[x]* ≥ 1 **:**
    - **if** *the elements x in* ψ *do not form one continuous interval* **: return** false;
    - $\varphi_{lin} \leftarrow$ open φ in the x-interval to a linear sequence while removing all the occurrences of x;
    - $\psi_{lin} \leftarrow$ open ψ in the x-interval to a linear sequence while removing all the occurrences of x;
    - **if** ψ_{lin} *is a subsequence of* φ_{lin} **: return** true;
    - **return** false;
  - **else** remove all the occurrences of x from φ ;
- **return** false;

Proof. The function `NC_subsequence`(ψ, φ) returns true iff ψ is a subsequence of φ. If ψ is a subsequence of φ, then the first element of the elimination ordering β appearing also in ψ must form one continuous interval in ψ. So we can use this element as a synchronizing point between φ and ψ and use the greedy linear algorithm for testing subsequences of normal (non-cyclic) sequences.

Assuming that we can afford the array `count` indexed by the elements of ψ, then the algorithm can be implemented in linear time w.r.t. the length of φ. We just need to represent the cyclic sequences φ and ψ as bidirectionally linked lists and remember for each x in φ (and ψ) a vector of pointers to the occurrences of x in φ (and ψ). $\square$

2.2 Drawings and embeddings

Let G be a graph. A *drawing* of G is a mapping assigning a unique point p_v of the plane $\mathbb{R}^2$ to each vertex $v \in V(G)$ and a continuous curve c_e to each edge $e \in E(G)$, $e = \{u, v\}$, such that p_u and p_v are the endpoints of c_e. A drawing of G is *planar* if for every edge $e \in E(G)$, $e = \{u, v\}$, the curve c_e avoids the points p_w for $w \in (V(G) \setminus \{u, v\})$ and the curves representing the edges do not intersect each other outside their endpoints. A graph G is *planar* if it has a planar drawing.

Even though the definition of planar drawings is quite intuitive, it has several disadvantages. It is hard to represent drawings in a discrete world, especially the continuous curves. Also, each planar graph has infinitely many planar drawings. We can just shift some vertices and edges by a little getting a new planar drawing that has the same structure as the original one. We resolve these issues by introducing an equivalence on the planar drawing of G.

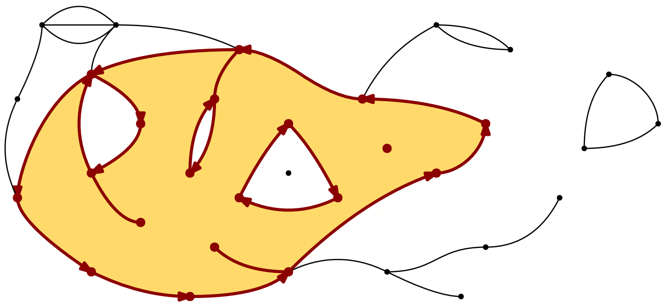


Figure 2.1: A planar drawing with a highlighted face and its directed boundary graph.

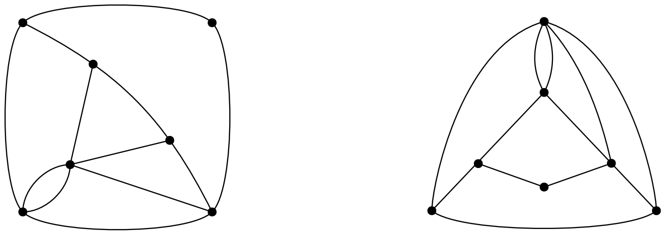


Figure 2.2: Two equivalent planar drawings of the same graph.

Let Γ be a planar drawing of G. The *rotation scheme* of a vertex $v \in V(G)$ in Γ is the cyclic counter-clockwise ordering of the curves representing the edges incident to v around the point p_v.

The planar drawing Γ of G divides the plane into connected regions, called the *faces* of Γ. The *boundary graph* of a face f is the subgraph B_f of G such that Γ projects the vertices and edges of B_f on the boundary of f. The edges of B_f lying inside the face f are not present in any other boundary graph. The remaining edges of B_f are contained in exactly two boundary graphs. By orienting the edges of B_f not lying inside f in such a way that the face f is located on the left side of them we produce the *directed boundary graph* of f.

Let $G = (V, E)$ be a graph and $\mathcal{G}$ its planar embedding. We let $F_{\mathcal{G}}$ denote the set of faces of $\mathcal{G}$, $l_{\mathcal{G}}(e)$ the left incident face of an edge $e \in E$, $r_{\mathcal{G}}(e)$ the right incident face of e, and $\sigma_{\mathcal{G}}(v)$ the rotation scheme of a vertex $v \in V$. The rotation schemes are oriented in the counter-clockwise direction.

We say that two planar drawings of G are *equivalent* if they have the same rotation schemes and the same directed boundary graphs. A class of this equivalence is called a *(planar) embedding* of G.

The planar embeddings are much easier to represent. Moreover, each (finite) graph has only finitely many planar embeddings.

Boyer and Myrvold [2] introduced an algorithm constructing an embedding of a graph G in time $\mathcal{O}(|V(G)| + |E(G)|)$. The embedding can be further converted to a planar drawing. For example, Chrobak and Payne [3] presented a linear time algorithm producing planar drawings with the vertices mapped to grid points and the edges realized by straight lines.

Let $\mathcal{G}$ be a planar embedding of G. Then we use the notation $F_{\mathcal{G}}$ for the set of faces of $\mathcal{G}$ and $\sigma_{\widehat{\mathcal{G}}}(v)$ for the rotation scheme of a vertex $v \in V(G)$ in $\mathcal{G}$. Next, $l_{\mathcal{G}}(e)$ and $r_{\mathcal{G}}(e)$ are respectively the left and the right incident face to an edge $e \in E(G)$. If not said otherwise, then we use the default orientation of the edge *e*.

Each planar embedding has a dual planar graph. The vertices of the dual are the faces of the embedding and there is a one-to-one correspondence between the edges of the dual and original.

Definition 8. *Let $\mathcal{G}$ be an embedding of a graph G. Then a multigraph $D_\mathcal{G}$ with self-loops is the* dual planar graph *to $\mathcal{G}$ if $V(D_\mathcal{G}) = F_\mathcal{G}$, and $\{f, f'\} \in E(G)$ iff there is an edge e' in G such that in one of its two orientations $l_\mathcal{G}(e') = f$, $r_\mathcal{G}(e') = f'$.*

The dual planar graph is always connected because the plane is connected.

In our problem, we look for embeddings that have their faces tagged by some labels. This object is formally characterized in the following definition.

Definition 9. *Let G be a graph and L a set of labels. The* labeled embedding *of G is a pair $\mathcal{G}_L = (\mathcal{G}, g)$ where $\mathcal{G}$ is a planar embedding of G and $g : F_\mathcal{G} \to L$ is a* face-labeling function. *The face-labeling function g is* connected *if for every $\ell \in L$ the set of faces $g^{-1}(\ell)$ induces a connected subgraph of the dual planar graph to $\mathcal{G}$. We say that a labeled embedding is* connected *if its face-labeling function is connected.*

2.3 Embedding restrictions

In this section, we formally define the Embedding Restriction Satisfiability problem. We start with a structure that puts some constraints on a labeled embedding.

Definition 10. *Let G be a graph and L a set of labels not containing a special token $\bigstar$. An* embedding restriction *of G is a quadruplet $\widehat{\mathcal{R}} = \left(\sigma_{\widehat{\mathcal{R}}}, l_{\widehat{\mathcal{R}}}, r_{\widehat{\mathcal{R}}}, T_{\widehat{\mathcal{R}}}\right)$ where*

(i) for each vertex $v \in V(G)$ $\sigma_{\widehat{\mathcal{R}}}(v)$ is a cyclic sequence of edges incident to v,

(ii) $l_{\widehat{\mathcal{R}}}, r_{\widehat{\mathcal{R}}} : E(G) \to L \cup \{\bigstar\}$ are functions prescribing labels for incident faces,

(iii) $T_{\widehat{\mathcal{R}}} \subseteq E(G)$ is a set of transparent *edges such that $l_{\widehat{\mathcal{R}}}(e) = r_{\widehat{\mathcal{R}}}(e)$ for each transparent edge $e \in T_{\widehat{\mathcal{R}}}$.*

Token $\bigstar$ acts as a "wildcard" that can stand for any label. If an edge is not transparent, then it is *opaque*.

Sometimes for illustration, we need to display an embedding restriction in a figure. In that case, we highlight the end sections of the edges anchored in rotation schemes in red. We use dashed lines for the transparent edges and we indicate the prescribed labels by capital letters.

Definition 11. *A labeled embedding $\mathcal{G}_L = (\mathcal{G}, g)$ of a graph G* satisfies *an embedding restriction $\widehat{\mathcal{R}} = \left(\sigma_{\widehat{\mathcal{R}}}, l_{\widehat{\mathcal{R}}}, r_{\widehat{\mathcal{R}}}, T_{\widehat{\mathcal{R}}}\right)$ if the following conditions hold:*

(i) The embedding $\mathcal{G}$ respects the rotation schemes $\sigma_{\widehat{\mathcal{R}}}$,
$(\forall v \in V(G))$ $\sigma_{\widehat{\mathcal{R}}}(v)$ is a subsequence of $\sigma_\mathcal{G}(v)$.

(ii) The labels assigned by g comply with the conditions prescribed by $l_{\widehat{\mathcal{R}}}$ and $r_{\widehat{\mathcal{R}}}$,
$(\forall e \in E(G))$ $l_{\widehat{\mathcal{R}}}(e) \in \{\bigstar, g(l_\mathcal{G}(e))\}$ and $r_{\widehat{\mathcal{R}}}(e) \in \{\bigstar, g(r_\mathcal{G}(e))\}$.

(iii) *The transparent edges have the same label on both sides,* $\left(\forall e \in T_{\widehat{\mathcal{R}}}\right)\ g(l_{\mathcal{G}}(e)) = g(r_{\mathcal{G}}(e))$.

We distinguish two basic problems that look for a labeled embedding satisfying an embedding restriction.

Problem 12 (Embedding Restriction Satisfiability (ERS))**.**
Input: *A graph G, an embedding restriction $\widehat{\mathcal{R}}$ of G.*
Question: *Is there a labeled embedding $\mathcal{G}_L$ of G which satisfies $\widehat{\mathcal{R}}$?*

Problem 13 (Embedding Restriction Connected Satisfiability (ERCS))**.**
Input: *A graph G, an embedding restriction $\widehat{\mathcal{R}}$ of G.*
Question: *Is there a connected labeled embedding $\mathcal{G}_L$ of G satisfying $\widehat{\mathcal{R}}$?*

Later in chapter 3, we show that both of them are NP-complete for general embedding restrictions. However, we can place some additional constraints on the embedding restrictions, that affects the complexity of the problems.

Definition 14. *Let G be a graph and $\widehat{\mathcal{R}}$ an embedding restriction of G. We say that $\widehat{\mathcal{R}}$ has*

(i) labeled opaque edges *if no opaque edge has prescribed the token $\bigstar$ for its incident edges,* $\left(\forall e \in (E(G) \setminus T_{\widehat{\mathcal{R}}})\right) l_{\widehat{\mathcal{R}}}(e) \neq \bigstar \neq r_{\widehat{\mathcal{R}}}(e)$,

(ii) anchored borders *if each edge $e \in E(G)$, $e = \{u, v\}$, such that $l_{\widehat{\mathcal{R}}}(u, v) \neq r_{\widehat{\mathcal{R}}}(u, v)$ appears in both $\sigma_{\widehat{\mathcal{R}}}(u)$ and $\sigma_{\widehat{\mathcal{R}}}(v)$.*

2.4 Partially embedded planarity

The Partial Embedded Planarity problem addresses the question of whether a planar embedding of a subgraph can be extended to an embedding of the entire graph. It was introduced by Angelini et al. [1], who also presented a linear time algorithm for the problem. However, the original algorithm was quite complicated. There is a bit simpler linear algorithm [11] that inspired the creation of the ERS problem. Actually, the second algorithm transforms its inputs into a specific type of ERCS instances and then it solves them.

Definition 15. *Let G be a graph and H its subgraph. Further, let $\mathcal{G}$ and $\mathcal{H}$ be embeddings of G and H respectively. We say that $\mathcal{G}$ is an* extension *of $\mathcal{H}$ if $\mathcal{H}$ is obtained from $\mathcal{G}$ by removing the vertices and edges that are not in H.*

Problem 16 (Partially Embedded Planarity (PEP) [1])**.**
Input: *A graph G, its subgraph H, a planar embedding $\mathcal{H}$ of H.*
Question: *Is there an embedding of G that is an extension of $\mathcal{H}$?*

Theorem 17 (Angelini et al. [1])**.** *PEP can be solved in linear time.*

Consider an instance PEP$(G, H, \mathcal{H})$ such that H has no isolated vertices. We describe how to construct an embedding restriction $\widehat{\mathcal{R}}$ of an equivalent ERCS instance. First, we employ the same rotation schemes as $\mathcal{H}$. For each vertex

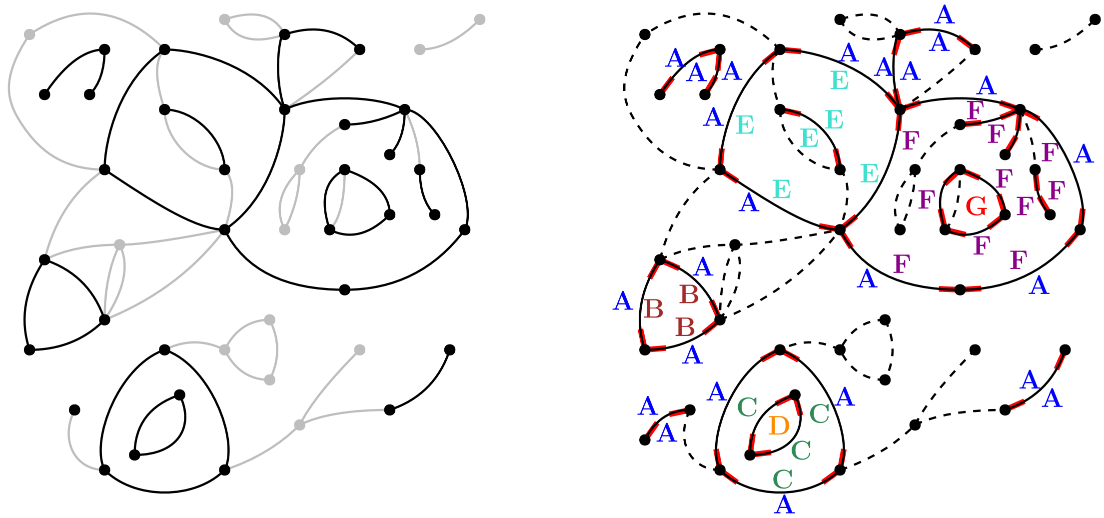


Figure 2.3: On the left, there is a PEP instance. The vertices and edges of H are black and the remaining vertices and edges of G are gray. On the right, there is an equivalent ERCS instance. Notice that the isolated vertex of H is replaced by a path of length one.

$v \in V(H)$ we put $\sigma_{\widehat{\mathcal{R}}}(v) = \sigma_{\mathcal{H}}(v)$ while leaving the rotation schemes of the vertices not in H empty. Next, we use the faces of $\mathcal{H}$ as labels. For each edge $e \in E(H)$ we prescribe $l_{\widehat{\mathcal{R}}}(e) = l_{\mathcal{H}}(e)$ and $r_{\widehat{\mathcal{R}}}(e) = r_{\mathcal{H}}(e)$. All the remaining edges are transparent and we do not specify their incident labels. So, $T_{\widehat{\mathcal{R}}} = (E(G) \setminus E(H))$ and $l_{\widehat{\mathcal{R}}}(e) = r_{\widehat{\mathcal{R}}}(e) = \bigstar$ for each edge $e \in T_{\widehat{\mathcal{R}}}$.

Lemma 18. *PEP*$(G, H, \mathcal{H})$ *accepts iff ERCS*$(G, \widehat{\mathcal{R}})$ *accepts.*

Proof. Let $\mathcal{G}$ be an embedding of G that is an extension of $\mathcal{H}$ and let g be the face-labeling function assigning to each face f of $\mathcal{G}$ the face of $\mathcal{H}$ in which f is located. Realize that $(\mathcal{G}, g)$ is a connected labeled embedding of G satisfying $\widehat{\mathcal{R}}$.

For the second implication, let us assume that $(\mathcal{G}, g)$ is a connected labeled embedding of G satisfying $\widehat{\mathcal{R}}$. We want to show that $\mathcal{G}$ is an extension of $\mathcal{H}$. So, let $\mathcal{H}'$ denote the embedding of H obtained from $\mathcal{G}$ by removing the vertices and edges not in H. Apparently, the rotation schemes of $\mathcal{H}'$ and $\mathcal{H}$ are identical, because $\sigma_{\widehat{\mathcal{R}}}(v) = \sigma_{\mathcal{H}}(v)$ for each $v \in V(H)$. Also, $\mathcal{H}'$ and $\mathcal{H}$ have the same number of faces. And since we removed only transparent edges while constructing $\mathcal{H}'$, then there is a bijection between the faces of $\mathcal{H}$ and $\mathcal{H}'$ assigning to a face f of $\mathcal{H}$ the face of $\mathcal{H}'$ tagged by f. The corresponding faces of $\mathcal{H}$ and $\mathcal{H}'$ have the same directed boundary graphs. Thus, $\mathcal{H}$ and $\mathcal{H}'$ must be identical embeddings, and so $\mathcal{G}$ is an extension of $\mathcal{H}$. □

If the graph H has some isolated vertices, then we can add a new incident edge to each such vertex, effectively replacing the isolated vertices in H by paths of length one. The modified PEP instance is equivalent to the original one and it allows us to apply Lemma 18.

Notice that the created embedding restriction $\widehat{\mathcal{R}}$ has labeled opaque edges and anchored borders.

2.5 Augmented embedding restrictions

An embedding restriction allows us to prescribe labels for incident faces of each edge. But what if we want to enforce that there is a face incident to a vertex v which is tagged by a label ℓ? We can achieve this by including labels in rotation schemes.

First, we define the augmented rotation scheme of a labeled embedding that also contains the labels of faces incident to a vertex.

Definition 19. *Let G be a graph, $\mathcal{G}_L = (\mathcal{G}, g)$ its labeled embedding and v a vertex of G. Further, let $\sigma_{\mathcal{G}}(v) = (e_1, e_2, \ldots, e_k)$ be the rotation scheme of v in $\mathcal{G}$. The* augmented rotation scheme *of v in $\mathcal{G}_L$ is the cyclic sequence $\rho_{\mathcal{G}_L}(v) = (e_1, g(l_{\mathcal{G}}(e_1)), e_2, g(l_{\mathcal{G}}(e_2)), \ldots, e_k, g(l_{\mathcal{G}}(e_k)))$.*

Notice that if $\mathcal{G}_L = (\mathcal{G}, g)$ is a connected labeled embedding, then the augmented rotation scheme $\rho_{\mathcal{G}_L}(v)$ of a vertex v is (x, y)-non-crossing for each pair of distinct x, y. The edges of G do not cause any problems, because each edge incident to v appears exactly once in $\rho_{\mathcal{G}_L}(v)$. And if there are two labels x, y such that $\rho_{\mathcal{G}_L}(v)$ contains the subsequence (x, y, x, y), then the sets of vertices $g^{-1}(x)$ and $g^{-1}(y)$ could not simultaneously induce connected subgraphs of the dual planar graph to $\mathcal{G}$.

The following definitions are just variations of Definitions 10, 11 and Problems 12, 13.

Definition 20. *Let G be a graph and L a set of labels not containing a special token $\bigstar$. An* augmented embedding restriction *of G is a quadruplet $\widehat{\mathcal{A}} = \left(\rho_{\widehat{\mathcal{A}}}, l_{\widehat{\mathcal{A}}}, r_{\widehat{\mathcal{A}}}, T_{\widehat{\mathcal{A}}}\right)$ where*

(i) for each vertex $v \in V(G)$ $\rho_{\widehat{\mathcal{A}}}(v)$ is a cyclic sequence of edges incident to v and labels of L such that there are not two consecutive occurrences of the same label,

(ii) $l_{\widehat{\mathcal{A}}}, r_{\widehat{\mathcal{A}}} : E(G) \to L \cup \{\bigstar\}$ are functions prescribing labels for incident faces,

(iii) $T_{\widehat{\mathcal{A}}} \subseteq E(G)$ is a set of transparent *edges such that $l_{\widehat{\mathcal{A}}}(e) = r_{\widehat{\mathcal{A}}}(e)$ for each transparent edge $e \in T_{\widehat{\mathcal{A}}}$.*

Definition 21. *A labeled embedding $\mathcal{G}_L = (\mathcal{G}, g)$ of a graph G* satisfies *an augmented embedding restriction $\widehat{\mathcal{A}}$ if the following conditions hold:*

(i) The labeled embedding $\mathcal{G}_L$ respects the rotation schemes $\rho_{\widehat{\mathcal{A}}}$:
$(\forall v \in V)$ $\rho_{\widehat{\mathcal{A}}}(v)$ is a cyclic subsequence of $\rho_{\mathcal{G}_L}(v)$.

(ii) The labels assigned by g comply with the conditions prescribed by $l_{\widehat{\mathcal{A}}}$ and $r_{\widehat{\mathcal{A}}}$,
$(\forall e \in E)$ $l_{\widehat{\mathcal{A}}}(e) \in \{\bigstar, g(l_{\mathcal{G}}(e))\}$ & $r_{\widehat{\mathcal{A}}}(e) \in \{\bigstar, g(r_{\mathcal{G}}(e))\}$.

(iii) The transparent edges have the same label on both sides,
$\left(\forall e \in T_{\widehat{\mathcal{A}}}\right)$ $g(l_{\mathcal{G}}(e)) = g(r_{\mathcal{G}}(e))$.

Problem 22 (Augmented ERS (AERS))**.**
Input: *A graph G, an augmented embedding restriction $\widehat{\mathcal{A}}$ of G.*
Question: *Is there a labeled embedding $\mathcal{G}_L$ of G satisfying $\widehat{\mathcal{A}}$?*

Problem 23 (Augmented ERCS (AERCS))**.**
Input: *A graph G, an augmented embedding restriction $\widehat{\mathcal{A}}$ of G.*
Question: *Is there a connected labeled embedding $\mathcal{G}_L$ of G satisfying $\widehat{\mathcal{A}}$?*

The ER(C)S problem is just a special case of AER(C)S. Therefore, ER(C)S instances can be straightforwardly reduced to AER(C)S instances. However, there is also a reduction in the opposite direction.

Lemma 24. *An AER(C)S instance can be converted to an equivalent ER(C)S instance in polynomial time.*

Proof. (sketch) Let $\widehat{\mathcal{A}}$ be an augmented embedding restriction of a graph G. We construct a graph H and an embedding restriction $\widehat{\mathcal{S}}$ such that AER(C)S$(G, \widehat{\mathcal{R}})$ is satisfiable iff ER(C)S$(H, \widehat{\mathcal{S}})$ is satisfiable. We begin with H equal to G and for every vertex v we add a new edge for each occurrence of a label in $\rho_{\widehat{\mathcal{A}}}(v)$. A new substituting edge replacing label ℓ in $\rho_{\widehat{\mathcal{A}}}(v)$ is incident to v and its second vertex appears only in this edge. The embedding restriction $\widehat{\mathcal{S}}$ is derived from $\widehat{\mathcal{A}}$ by replacing the labels in rotation schemes by the substituting edges and prescribing corresponding labels for the incident faces of the substituting edges. □

The three subsequent lemmata are simple observations about the AERS and AERCS problems.

Lemma 25. *Let G be a graph and $\widehat{\mathcal{A}}$ an augmented embedding restriction of G. AERCS$(G, \widehat{\mathcal{A}})$ can be satisfied only if ERCS$(G, \widehat{\mathcal{R}})$ is satisfiable, where $\widehat{\mathcal{R}}$ is the embedding restriction derived from $\widehat{\mathcal{A}}$ by omitting labels from the rotation schemes.*

Lemma 26. *Let G be a graph and $\widehat{\mathcal{A}}$ an augmented embedding restriction of G. If there is a vertex $v \in V(G)$ and two distinct labels x, y such that (x, y, x, y) is a cyclic subsequence of $\rho_{\widehat{\mathcal{A}}}(v)$, then AERCS$(G, \widehat{\mathcal{A}})$ cannot be satisfied.*

Lemma 27. *Let G be a graph, v and w vertices of G, $e = (v, w)$ an edge of G, ℓ a label and $\widehat{\mathcal{A}}$ an augmented embedding restriction of G such that $l_{\widehat{\mathcal{A}}}(e) = \ell$ and $\rho_{\widehat{\mathcal{A}}}(v) = (e, \ell, \tau)$, where τ is a sequence of labels and edges incident to v. Further, let $\rho' = (e, \tau)$ be a cyclic sequence and $\widehat{\mathcal{A}}'$ is the augmented embedding restriction of G derived from $\widehat{\mathcal{A}}$ by replacing $\rho_{\widehat{\mathcal{A}}}(v)$ by ρ'. Then, a labeled embedding $\mathcal{G}_L$ of G satisfies $\widehat{\mathcal{A}}$ iff it satisfies $\widehat{\mathcal{A}}'$.*

Proof. The label ℓ is already enforced by the edge e, so it is not necessary to put it in the rotation scheme of vertex v. □

Remark. An analogous observation can be made for $r_{\widehat{\mathcal{A}}}(e) = \ell$, $\rho_{\widehat{\mathcal{A}}}(v) = (\ell, e, \tau)$ and $\rho' = (e, \tau)$.

2.6 Connectivity and SPQR-trees

Some algorithms of chapter 4 are based on the SPQR decomposition of biconnected graphs. The decomposition is described by a data structure called SPQR-tree. These trees were first introduced by Di Battista and Tamassia [4], who utilized them to represent the set of all planar embeddings of a biconnected graph.

Since then, SPQR-trees were applied in many algorithms processing biconnected graphs. We define the SPQR-trees using the same notation as Gutwenger and Mutzel [8], who showed how to construct SPQR-trees in linear time.

A graph G is *connected* if for every pair of vertices $u, v \in V(G)$ there exists a path between u and v. A maximal connected subgraph of G is called a *component* of G. A connected graph is *biconnected* if for each triplet x, y, z of distinct vertices there is a path from x to y avoiding z.

Let G be a biconnected graph and $u, v \in V(G)$ a pair of its vertices. We can decompose E into equivalence classes $E_1, \dots, E_k$ such that two edges are in the same class iff they lie on a common trail not containing vertices u and v except as endpoints. The first vertex of the trail might be equal to the last one. The classes $E_1, \dots, E_k$ are called the *separation classes* of G with respect to $\{u, v\}$. If there are at least two separation classes neither of them being a single edge, or if there are at least three separation classes and one of them is not a single edge, then we say that $\{u, v\}$ is a *separation pair* of G.[1] The graph G is *triconnected* if it contains no separation pair.

We further define four special types of biconnected graphs.

Definition 28. *Let G be a graph. We say that G is*

(S) *an* S-skeleton *if it is a cycle,*

(P) *a* P-skeleton *if it is a dipole graph, i.e. a graph with 2 vertices and at least 3 edges between them,*

(Q) *a* Q-skeleton *if it is a graph with 2 vertices and at most 2 edges between them,*

(R) *an* R-skeleton *if it is a triconnected graph without multiple edges and it is neither an S-skeleton nor a Q-skeleton.*

When we want to cover more than one type of skeletons, we put all the relevant types inside a pair of brackets. For example, a graph is an [SR]-skeleton, if it is either an S-skeleton or an R-skeleton.

Lemma 29. *A biconnected graph has a separation pair iff it is not a [PQR]-skeleton or C_3 (i.e. a cycle with 3 vertices).*

Proof. [PQR]-skeletons and C_3 do not have a separation pair.

Let G be a graph that is biconnected and it is not a [PQR]-skeleton or C_3. G is not a [PQ]-skeleton, so it has at least 3 vertices. If G has multiple edges between a pair of vertices u and v, then $\{u, v\}$ is a separation pair of G. Otherwise, G is a biconnected graph without multiple edges and it is not C_3, so it has at least 4 vertices. Since G is not an R-skeleton, then G is not triconnected. It means G must have a separation pair. □

Let $\{u, v\}$ be a separation pair of G and let $E_1, \dots, E_k$ be separation classes of G w.r.t. $\{u, v\}$. Further, let $I \subset \{1, \dots, k\}$ be an index set. We partition the edges of G according to I into two multisets $E_I = \bigcup_{i \in I} E_i$ and $E'_I = (E \setminus E_I)$. If

[1] Gutwenger and Mutzel [8] also consider $\{u, v\}$ to be a separation pair if there are at least 4 separation classes and all of them consists of a single edge.

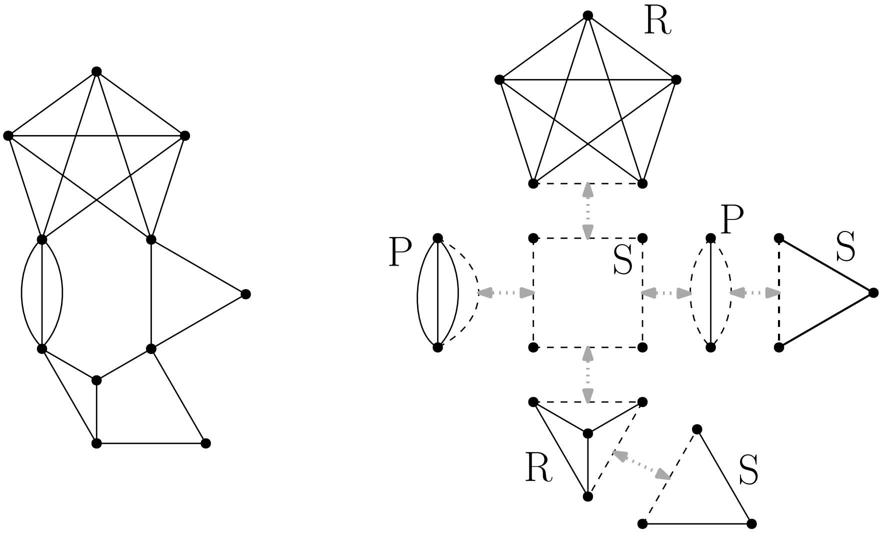


Figure 2.4: An SPQR-tree of a biconnected graph. The virtual edges are dashed and each pair of dual virtual edges is linked together by a gray arrow.

both E_I and E'_I contain at least two edges, then we say that the pair of graphs $H = (V(E_I), E_I \cup (u,v))$ and $H' = (V(E'_I), E'_I \cup (u,v))$ is a *pair of split graphs* of G with respect to $\{u,v\}$. The extra copies of edge (u,v) added to the H and H' are called *virtual edges*. Let $e_G^{virt}(H)$ denote the virtual edge of H and $e_G^{virt}(H')$ the virtual edge of H'. We say that $e_G^{virt}(H')$ is *the dual virtual edge* to $e_G^{virt}(H)$ and vice versa. Notice that the edges $e_G^{virt}(H)$ and $e_G^{virt}(H')$ have the same default orientation and that both H and H' are biconnected. If E_I or E'_I consists of just one separation class, then we say that the pair of split graphs (H, H') is *simple*. It can be easily observed that there is at least one simple pair of split graphs for each separation pair.

We can continue recursively splitting H and H' until they have no separation pairs. This process yields an SPQR-tree of the graph G.

Formally, we start with the set of graphs $N = \{G\}$ containing only the graph G. While there is a graph G_* in N that has a separation pair, we find a pair of its split graphs (H_*, H'_*) and replace G_* in N by H_* and H'_*. From Lemma 29, it follows that when this algorithm stops, all the graphs in N are [SPQR]-skeletons. (And all the S-skeletons in N are isomorphic to C_3.) The skeletons in N are linked together by pairs of virtual edges created in the same splitting operation. These links form a tree structure. Let $A \subseteq \binom{N}{2}$ be a set such that $\{G_1, G_2\} \in A$ iff there is a virtual edge e such that e is an edge of G_1 and G_2 contains the dual virtual edge to e. The graph $\tau = (N, A)$ is an *SPQR-tree* of the graph G. In order to easily differentiate between the vertices and edges of a graph and its SPQR-tree, we refer to the vertices of an SPQR-tree as *nodes* and to the edges as *arcs*.

The opposite operation to the splitting of a graph to a pair of split graphs is called *merging*. If G_1 and G_2 are two nodes of an SPQR-tree $\tau = (N, A)$ such that $\{G_1, G_2\} \in A$, then we can unite the graphs G_1 and G_2 leaving out the two dual virtual edges while removing the arc $\{G_1, G_2\}$ from A. It is a common practice to merge P-skeletons with the same set of vertices and S-skeletons sharing a pair of dual virtual edges.

Theorem 30 (Gutwenger and Mutzel [8]). *Each biconnected graph has an SPQR-tree and it can be constructed in linear time.*

Lemma 31. *For every biconnected graph G that is not an [SPQR]-skeleton, there exists a separation pair $\{u, v\}$ and a simple pair of split graphs (H, H') w.r.t. $\{u, v\}$ such that H is an [SPR]-skeleton.*

Proof. Let G be a biconnected graph that is not an [SPQR]-skeleton and let τ be an SPQR-tree of G. If τ has a leaf μ which is an [SR]-skeleton, then we choose $\{u, v\}$ as the vertices of the virtual edge of μ, H as μ and H' as the merge of the remaining nodes. (H consists of one separation class and the virtual edge.) Otherwise, all the leaves of τ are P-skeletons. Let us root τ in an arbitrary node. G is not a P-skeleton, so there must be a node ξ of τ that is not a P-skeleton and all of its descendants are P-skeletons. Let e be a virtual edge of ξ that is not the virtual edge connecting ξ to its parent node. We choose $\{u, v\}$ as the vertices of e, H as the merge of the nodes of the sub-tree connected to ξ via the virtual edge e and H' as the merge of the remaining nodes. Then H is a P-skeleton and $\{u, v\}$ is not a separation pair of H', so H' consists of one separation class and the virtual edge e. □

2.7 Eulerian Circuits

Using several known results about Eulerian circuits, we can sometimes construct a certificate demonstrating that an embedding restriction with labeled opaque edges is unsatisfiable. An Eulerian circuit of a graph is a walk starting and ending in the same vertex $v \in V(G)$ that visits each edge exactly once. There is a well-known criterion for recognizing graphs with an Eulerian circuit. A directed graph G has an Eulerian circuit iff for each vertex v the number of edges coming to v is equal to the number of edges leaving v and G has at most one component that is not an isolated vertex.

Let $\widehat{\mathcal{R}}$ be an embedding restriction with labeled opaque edges of a graph G and let $\mathcal{G}_L$ be a labeled embedding of G satisfying $\widehat{\mathcal{R}}$. If the two faces of $\mathcal{G}_L$ incident to an edge $e \in E(G)$ are tagged by different labels, then these labels must be prescribed by $l_{\widehat{\mathcal{R}}}(e)$ and $r_{\widehat{\mathcal{R}}}(e)$. Therefore, the restriction $\widehat{\mathcal{R}}$ determines which labels neighbor with each other. For a vertex $v \in V(G)$ we can even create a directed graph characterizing the arrangement of labels tagging the faces incident to v.

Definition 32. *Let u be a vertex of the graph G and let*

$$E_u = \{(u, v) \in E(G) \mid l_{\widehat{\mathcal{R}}}((u, v)) \neq r_{\widehat{\mathcal{R}}}((u, w))\}$$

denote the set of border edges in $\widehat{\mathcal{R}}$ incident to u. The edges of E_u are oriented from u outwards. We further define the set of labels incident to u and the multiset of ordered pairs of adjacent labels as follows:

$$V_L^u = \{l_{\widehat{\mathcal{R}}}(e) \mid e \in E_u\} \cup \{r_{\widehat{\mathcal{R}}}(e) \mid e \in E_u\},$$
$$E_L^u = \{(l_{\widehat{\mathcal{R}}}(e), r_{\widehat{\mathcal{R}}}(e)) \mid e \in E_u\}.$$

The directed graph $G_L^u = (V_L^u, E_L^u)$ is called the label ordering graph *of u in $\widehat{\mathcal{R}}$.*

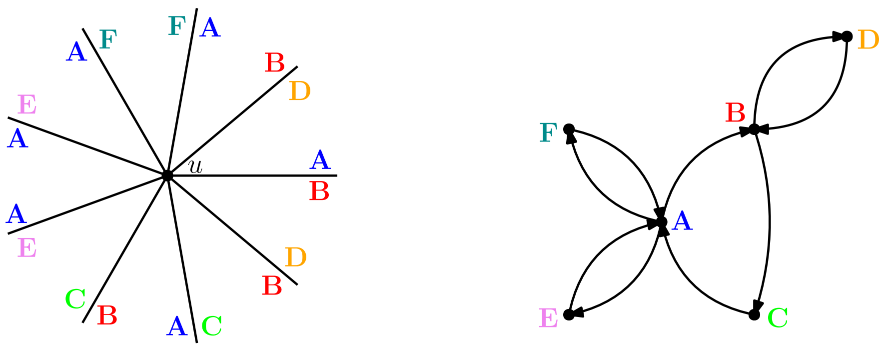


Figure 2.5: The border edges incident to a vertex u in an embedding restriction $\widehat{\mathcal{R}}$ and the corresponding label ordering graph of u in $\widehat{\mathcal{R}}$.

Lemma 33. *If G has a labeled embedding satisfying $\widehat{\mathcal{R}}$, then for each vertex $u \in V(G)$ the label ordering graph of u in $\widehat{\mathcal{R}}$ has an Eulerian circuit.*

Proof. Let $(\mathcal{G}, g)$ be a labeled embedding of G satisfying $\widehat{\mathcal{R}}$. There is a natural bijection between the edges of E_u and the edges of the label ordering graph G_L^u. The order of the edges of E_u in the rotation scheme $\sigma_{\widehat{\mathcal{R}}}(u)$ yields an Eulerian circuit in G_L^u. □

Further, if we look only for connected labeled embeddings, then we get a stronger condition. There must be an Eulerian circuit ε such that no two labels alternate in ε. It is forbidden that ε first visits a label ℓ_1, then it walks through another label ℓ_2, it returns to ℓ_1, and finally, it comes back to ℓ_2.

Definition 34. *We say that an Eulerian circuit ε of G has a* crossing *if there exist two vertices $u, v \in V(G)$ such that (u, v, u, v) is a subsequence of ε.*

Lemma 35. *If G has a connected labeled embedding satisfying $\widehat{\mathcal{R}}$, then for each vertex $u \in V(G)$ the label ordering graph of u in $\widehat{\mathcal{R}}$ has an Eulerian circuit without crossings.*

Proof. We use the same construction as in Lemma 33. If there was a crossing of labels ℓ_1 and ℓ_2, then at least one of these labels cannot from a connected region in the satisfying embedding. □

Surprisingly, it is not hard to test whether a graph has an Eulerian circuit without crossings. We can take an arbitrary Eulerian circuit ε and check it for crossings. The following lemma shows that ε has a crossing iff every Eulerian circuit of G has a crossing.

Lemma 36. *Let G be a directed graph with an Eulerian circuit. Then the following statements are equivalent.*

(i) *G has an Eulerian circuit with a crossing.*

(ii) *There exist two vertices $u, v \in V(G)$, $u \neq v$, such that there are two directed edge-disjoint paths from u to v in G.*

(iii) *Every Eulerian circuit of G has a crossing.*

Proof. The implication from (iii) to (i) is trivial.

The implication from (i) to (ii) is also easy. If an Eulerian circuit has a crossing of vertices x and y, then there are two edge-disjoint directed paths from x to y in the circuit.

It remains to show that (ii) yields (iii). Let us suppose that there are two directed edge-disjoint paths p_1 and p_2 from u to v and let ε be an Eulerian circuit of G. We show that ε has a crossing.

The circuit ε must visit both u and v at least twice. If ε has a crossing of u and v then we are done. Otherwise, there is an occurrence of the vertex u in ε such that starting from its position ε leaves u, then it does all its visits of v while avoiding u, and then it returns to u. Without loss of generality, let us assume that after leaving u the circuit ε uses the last edge of p_1 sooner than the last edge of p_2. We partition the circuit ε into three trails α, β and γ the following way. α starts in u, does not return there and it ends by the first visit of v. β starts with the first visit of v and it ends with arriving in v through the last edge of p_2. γ covers the rest. Let w be the vertex on p_2 such that all the edges of p_2 beyond w are on the trail β and the edge of p_2 leading to w is not on β. The vertex w is well defined, since the last edge of p_2 is in β and the first edge of p_2 is not there (β avoids u). Then, the circuit ε contains a crossing of v and w. □

3. NP-complete problems

This chapter contains the hardness results about ERS and some related problems. We prove that both ERS and ERCS in general settings are NP-complete. Later, we inspect ERS for embedding restrictions with labeled opaque edges. Finally, we show that even if we get a planar embedding $\mathcal{G}$ satisfying the rotation schemes of an embedding restriction $\widehat{\mathcal{R}}$, it is still hard to find a connected face-labeling function g such that the labeled embedding $(\mathcal{G}, g)$ satisfies $\widehat{\mathcal{R}}$.

But first, we present three NP-complete problems that are used to prove the NP-hardness of our problems. Two of them are variants of the Planar 3-SAT problem and the last one is a planar version of the Vertex-disjoint paths problem.

Definition 37. *Let φ be a Boolean formula in 3-CNF, C the set of clauses of φ and $X = \{x_1, \ldots, x_n\}$ the set of variables of φ. Further, let $E_C = \{\{x, c\} \mid x \in c \text{ or } \neg x \in c, \text{ for } x \in X, c \in C\}$ be the set of edges linking each clause to its variables, and let $E_X = \{\{x_1, x_2\}, \{x_2, x_3\}, \ldots, \{x_n, x_1\}\}$ be the cycle on the set of variables. Then, the graph $G_s(\varphi) = (C \cup X, E_C)$ is called the* simple associated graph *of φ and the graph $G(\varphi) = (C \cup X, E_C \cup E_X)$ is the* associated graph *of φ.*

Problem 38 (Simple Planar 3-SAT)**.**
Input: *A formula φ in 3-CNF, a planar embedding of its simple associated graph $G_s(\varphi)$.*
Question: *Is φ satisfiable?*

Problem 39 (Separable Planar 3-SAT [15])**.**
Input: *A formula φ in 3-CNF, a planar embedding $\mathcal{G}_\varphi$ of its associated graph $G(\varphi)$ such that for each variable x the edges representing occurrences of the positive literal x are separated from the edges representing the negative literal $\neg x$ by the edges of the cycle E_X.*
Question: *Is φ satisfiable?*

Theorem 40 (Lichtenstein [15])**.** *Simple Planar 3-SAT and Separable Planar 3-SAT are NP-complete.*

Problem 41 (Planar vertex-disjoint paths problem)**.**
Input: *A planar graph $G = (V, S \cup D)$.*
Question: *Are there vertex-disjoint circuits $C_1, \ldots, C_{|D|}$ such that $|E(C_i) \cap D| = 1$ for each $i \in \{1, \ldots, |D|\}$?*

Theorem 42 (Middendorf and Pfeiffer [16])**.** *The planar vertex-disjoint paths problem is NP-complete.*

3.1 ERS and ERCS

Theorem 43. *Both ERS and ERCS are NP-complete.*

Proof. Both ERS and ERCS are in NP. A labeled embedding of a graph G has polynomial size w.r.t. G, so we can generate a satisfying embedding using a non-deterministic algorithm, and verify its validity in polynomial time.

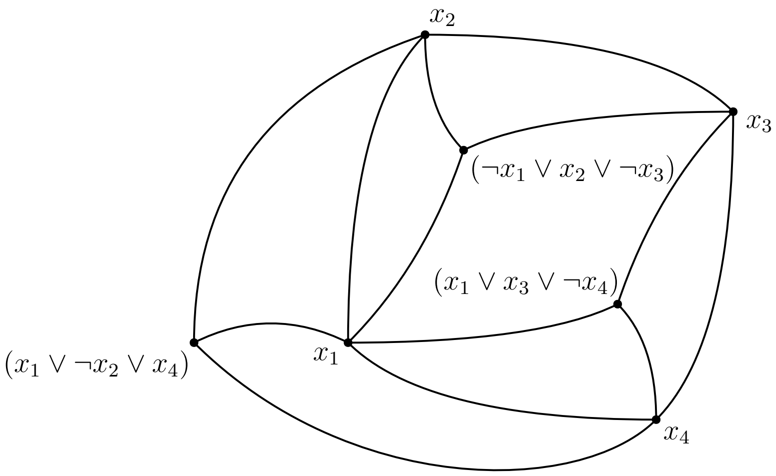


Figure 3.1: The associated graph of the Boolean formula
$\varphi(x_1, x_2, x_3, x_4) = (x_1 \vee \neg x_2 \vee x_4) \wedge (\neg x_1 \vee x_2 \vee \neg x_3) \wedge (x_1 \vee x_3 \vee \neg x_4)$.

It remains to prove that ERS and ERCS are NP-hard. We show a reduction of Separable Planar 3-SAT (Problem 39) working simultaneously for both ERS and ERCS.

Let φ be a formula in 3-CNF with variables $\{x_1, \ldots, x_n\}$ and $\mathcal{G}_\varphi$ an embedding of its associated graph $G(\varphi)$ such that for each variable the edges to positive literals are separated from edges to negative literals by the edges of the cycle $E_X = \{\{x_1, x_2\}, \ldots, \{x_n, x_1\}\}$. Without loss of generality, we suppose that each clause of φ has at least two literals. We describe how to construct a graph H and an embedding restriction $\widehat{\mathcal{R}}$ such that H has a (connected) labeled embedding satisfying $\widehat{\mathcal{R}}$ iff φ is satisfiable.

Let T be a label representing the value true and let $F_1, \ldots, F_n$ be labels representing the value false.

The idea behind the reduction is that we widen the edges of $\mathcal{G}_\varphi$. We replace vertices in $\mathcal{G}_\varphi$ by polygons where the number of sides of each polygon is equal to the degree of the corresponding vertex. The edges of $\mathcal{G}_\varphi$ are then realized as corridors connecting these polygons. For every $i \in \{1, \ldots n\}$ the polygon V_i representing the variable x_i is further split into two faces P_i, N_i in such a way that P_i is adjacent to the corridors leading to the clauses containing the positive literal x_i and N_i is adjacent to the corridors to the clauses containing the negative literal $\neg x_i$. The partition between the faces P_i and N_i connects the corridors of the edges of the cycle E_X. The embedding restriction $\widehat{\mathcal{R}}$ then enforces that one of the faces is assigned the label T and the second face is tagged by F_i. However, it does not specify which of the two faces has to be labeled by T. Both choices are possible. The labeling of P_i by T corresponds to the assignment $x_i = 1$ and the tagging of N_i by T means $x_i = 0$. Furthermore, the edges of V_i are transparent in $\widehat{\mathcal{R}}$, so that the labels assigned to the faces of V_i travels through the corridors towards the clauses. The embedding restriction $\widehat{\mathcal{R}}$ then requires that for each polygon representing a clause at least one of the incoming corridors is labeled by T. It guarantees that the assignment given by the labeling of P_i, N_i for each $i \in \{1, \ldots, n\}$ satisfies all the clauses.

Now, let us describe the reduction in more detail. The construction is illus-

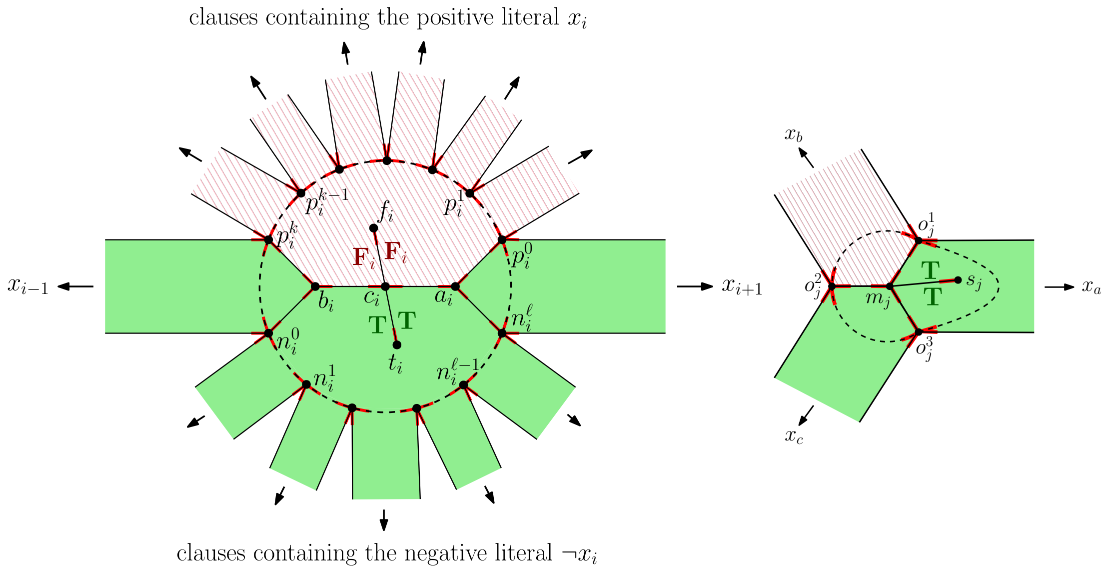


Figure 3.2: On the left, there is the structure representing a variable x_i. In this setting x_i is assigned the value false. The structure on the right implements a clause $c_j = (x_a \vee x_b \vee x_c)$. In this setting the clause c_j is satisfied by x_a.

trated in Figure 3.2. Unless otherwise noted, the edges of H are opaque in $\widehat{\mathcal{R}}$, they have prescribed the token $\bigstar$ for their incident faces and they are anchored in the rotation schemes of their vertices. The rotation schemes of $\widehat{\mathcal{R}}$ are not explicitly stated, however, they can be inferred from Figure 3.2.

Let x_i be a variable that appears in k clauses as the positive literal x_i and in ℓ clauses as the negative literal $\neg x_i$. Without loss of generality, let us assume that the edges in the rotation scheme $\sigma_{\mathcal{G}_\varphi}(x_i)$ are listed in the following order: the edge to x_{i+1}, the edges to clauses $c_1, \dots, c_k$ where x_i appears as the positive literal, the edge to x_{i-1}, the edges to clauses $c'_1, \dots, c'_\ell$ where x_i appears as the negative literal. In the graph H we represent x_i as the $(k+\ell+2)$-gon V_i with vertices $p_i^0, p_i^1, \dots, p_i^k, n_i^0, n_i^1, \dots, n_i^\ell$. The edges $\{p_i^0, p_i^1\}, \dots, \{p_i^k, n_i^0\}, \dots \{n_i^\ell, p_i^0\}$ are transparent in $\widehat{\mathcal{R}}$.

V_i is connected to V_{i+1} by a pair of edges. One of the edges is incident to n_i^ℓ and the second is incident to p_i^0. For $j \in \{1, \dots, k\}$, the connection to the clause c_j is realized by a pair of edges incident to p_i^{j-1} and p_i^j. Similarly, for $j \in \{1, \dots, \ell\}$ the connection to c'_j is done by two edges incident to n_i^{j-1} and n_i^j.

The partition of V_i into faces P_i, N_i is accomplished by three vertices a_i, b_i, c_i and the edges $\{n_i^0, b_i\}, \{p_i^k, b_i\}, \{b_i, c_i\}, \{c_i, a_i\}, \{a_i, p_i^0\}, \{a_i, n_i^\ell\}$. P_i is the face containing vertices $p_i^0, \dots, p_i^k, b_i, c_i, a_i$ and N_i is the face containing $n_i^0, \dots, n_i^\ell, a_i, c_i, b_i$. The assignment of the labels T and F_i to the the faces P_i, N_i is done by the addition of two vertices t_i, f_i and two edges $e_i^t = \{c_i, t_i\}, e_i^f = \{c_i, f_i\}$ such that e_i^t, e_i^f do not appear in the rotation scheme $\sigma_{\widehat{\mathcal{R}}}(c_i)$ and they have prescribed the labels T and F_i for their incident faces, respectively. (I.e. $l_{\widehat{\mathcal{R}}}(e_i^t) = r_{\widehat{\mathcal{R}}}(e_i^t) = T$, $l_{\widehat{\mathcal{R}}}(e_i^f) = r_{\widehat{\mathcal{R}}}(e_i^f) = F_i$.) Notice that V_i alone has two possible labeled embeddings satisfying $\widehat{\mathcal{R}}$ that differs only in the placement of the edges e_i^t, e_i^f.

Let c_j be a clause of φ with h literals. Further, let $x_j^1, \dots, x_j^h$ denote the literals of c_j in the order in which they appear in the rotation scheme $\sigma_{\mathcal{G}_\varphi}(c_j)$.

The clause c_j is then represented in H as an h-gon C_j with vertices $o_j^1, \dots, o_j^h$. The edges $\{o_j^1, o_j^2\}, \dots, \{o_j^h, o_j^1\}$ are transparent in $\widehat{\mathcal{R}}$.

For each $i \in \{1, \dots, h\}$, C_j is connected to the polygon representing the variable of the literal x_j^i by a pair of edges incident to vertices o_j^{i-1} and o_j^i. Next, C_j contains a central vertex m_j that is adjacent to $o_j^1, \dots, o_j^k$. In order to ensure that the vertex m_j is incident to a face labeled by T, we add one vertex s_j and one edge $e_j^s = \{m_j, s_j\}$ such that e_j^s does not appear in $\sigma_{\widehat{\mathcal{R}}}(m_j)$ and $l_{\widehat{\mathcal{R}}}(e_j) = r_{\widehat{\mathcal{R}}}(e_j^s) = T$.

If φ has a satisfying assignment ξ then we can construct a connected labeled embedding of H satisfying $\widehat{\mathcal{R}}$. For $i \in \{1, \dots, n\}$, if ξ assigns $x_i = 1$ then we put the edge e_i^t inside the face P_i, e_i^f inside N_i, plus we label P_i by T and N_i by F_i. Else if $x_i = 0$ in ξ then we put e_i^t in N_i, e_i^f in P_i, labeling N_i by T and P_i by F_i. Since ξ is a satisfying assignment, then for each clause c_j there is at least one literal satisfying c_j. Therefore, we can embed the edge e_j^s inside a face belonging to a satisfying literal. Further, we label the faces a_i, n_i^ℓ, p_i^0 and b_i, p_i^k, n_i^0 by T. We spread the assigned labels through the transparent edges. Finally, we assign a new unique label to each face that is still unlabeled. The created labeled embedding is connected and it satisfies $\widehat{\mathcal{R}}$.

If H has a labeled embedding satisfying $\widehat{\mathcal{R}}$, then for every $i \in \{1, \dots, n\}$ it labels one of the faces P_i, N_i by T and the other by F_i. From these labelings, we derive an assignment ξ of variables of φ. We set $x_i = 1$ in ξ iff P_i is labeled by T. The labels of P_i and N_i travel through the transparent edges towards the clauses and for each clause c_j the edge e_j^s is embedded in a face tagged by T. Therefore, the assignment ξ satisfies all the clauses of φ.

It is easy to observe that if ERCS$(H, \widehat{\mathcal{R}})$ is satisfiable, then ERS$(H, \widehat{\mathcal{R}})$ is also satisfiable. Therefore, we showed that φ has a satisfying assignment if and only if ERCS$(H, \widehat{\mathcal{R}})$ is satisfiable and it happens if and only if ERS$(H, \widehat{\mathcal{R}})$ satisfiable. Moreover, the graph H and the embedding restriction $\widehat{\mathcal{R}}$ can be constructed in polynomial time w.r.t. the length of φ, so both ERS and ERCS are NP-hard. □

Notice that the construction of H and $\widehat{\mathcal{R}}$ would work even if we omitted the transparent edges. In addition, the embedding restriction $\widehat{\mathcal{R}}$ has anchored borders.

Corollary 44. *ERS and ERCS are NP-complete even for embedding restrictions without transparent edges and with anchored borders.*

3.2 ERS with labeled opaque edges

Theorem 45. *ERS for embedding restrictions with labeled opaque edges is NP-complete.*

Proof. The problem is in NP. We can use the same argument as in Theorem 43.

We describe a reduction from Simple Planar 3-SAT (Problem 38) to ERS with labeled opaque edges. Let φ be a formula in 3-CNF and $\mathcal{G}_\varphi$ an embedding of its simple associated graph $G_s(\varphi)$. We construct a graph H and an embedding restriction $\widehat{\mathcal{R}}$ with labeled opaque edges such that φ is satisfiable iff H has a labeled embedding satisfying $\widehat{\mathcal{R}}$.

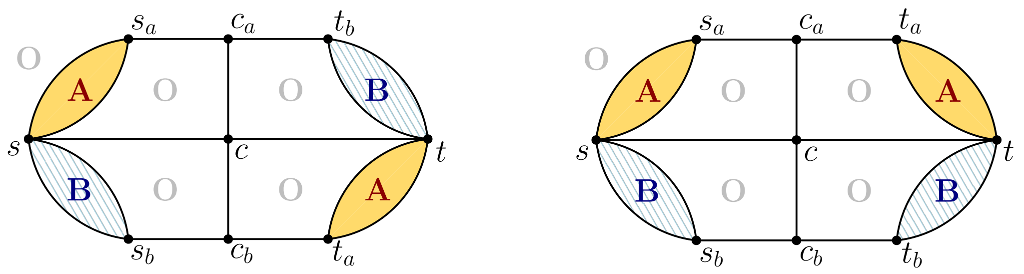


Figure 3.3: On the left, there is the negative embedding of the transporter graph. On the right, there is the negative embedding of the negator.

The embedding restriction $\widehat{\mathcal{R}}$ uses three labels A, B and O. All the edges of H are opaque in $\widehat{\mathcal{R}}$ and they have prescribed labels for their incident faces. Thus, $\widehat{\mathcal{R}}$ has labeled opaque edges. Unless otherwise noted, the edges of H are not listed in rotation schemes and they have prescribed the label O for their incident faces.

Let ℓ be a label. Then an *ℓ-pocket* is a graph with two vertices u, v and two edges $e_l = (u, v)$, $e_r = (u, v)$ such that $l_{\widehat{\mathcal{R}}}(e_l) = r_{\widehat{\mathcal{R}}}(e_r) = \ell$ and $r_{\widehat{\mathcal{R}}}(e_l) = l_{\widehat{\mathcal{R}}}(e_r) = O$.

The basic building block of the construction is the *transporter* graph which is illustrated in Figure 3.3. Its purpose is to transport the assignment of a variable to a clause containing the variable. It has 2 important vertices the *source* s, that corresponds to the variable, and the *target* t corresponding to the clause. Except for s and t, the transporter has 7 additional vertices c, s_a, s_b, t_a, t_b, c_a, c_b and it consists of:

(i) the *spine* formed by edges $\{s, c\}$, $\{c, t\}$,

(ii) two A-pockets on vertices s, s_a and t, t_a,

(iii) two B-pockets on vertices s, s_b and t, t_b,

(iv) two *ribs* $\{\{c_a, s_a\}, \{c_a, t_b\}, \{c_a, c\}\}$ and $\{\{c_b, s_b\}, \{c_b, t_a\}, \{c_b, c\}\}$.

The transporter has two possible labeled embeddings satisfying $\widehat{\mathcal{R}}$. Both of them are characterized by the rotation scheme $\sigma(s)$ of the vertex s. The labeled embedding where $\sigma(s)$ contains the A-pocket, the spine and the B-pocket in this order is called the *positive* embedding. The other embedding ($\sigma(s) =$ (the B-pocket, the spine, the A-pocket)) is called the *negative* embedding.

The *negator* is a variation of the transporter that contains the edges $\{c_a, t_a\}$, $\{c_b, t_b\}$ instead of $\{c_a, t_b\}$, $\{c_b, t_a\}$.

The idea of the reduction is that we take the embedding $\mathcal{G}_\varphi$ and we replace its edges by transporters. An edge corresponding to an occurrence of a positive literal is replaced by the transporter and the edge signifying a negative literal is replaced by the negator. We place the transporters in such a way that the variable vertex is always the source. Next, we ensure that for each variable vertex either all of the incident transporters have the positive embedding, or all of them have the negative one. And for each clause, we require that at least one of the transporters brings a satisfying assignment. I.e. for each clause c there is a transporter T such that in the rotation scheme of c the A-pocket of T, the spine of T and the B-pocket of T are listed in this order.

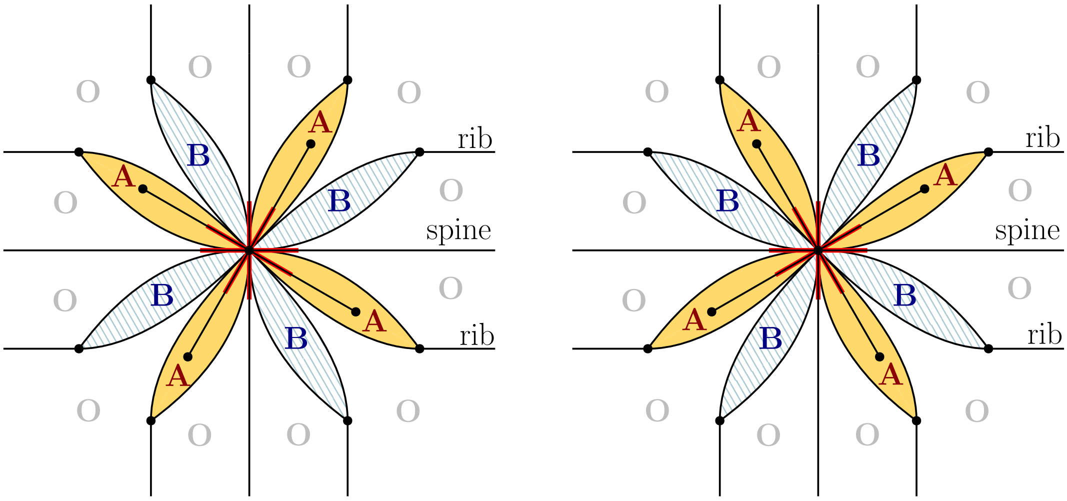


Figure 3.4: The surroundings of a vertex representing a variable x that appears in 4 clauses. On the left, there is the positive labeled embedding corresponding to $x = 1$. On the right, there is the negative embedding corresponding to $x = 0$.

We have to slightly modify the graph $G_s(\varphi)$ and its embedding $\mathcal{G}_\varphi$. Into each face of $\mathcal{G}_\varphi$ we add a new *ground* vertex. And for each clause c' of φ with k' literals we add k' edges from c' to the ground vertices of the incident faces. We add the edges in such a way that the edges to variables of c' and the edges to the ground vertices alternate regularly in the rotation scheme of c'. (It is possible that multiple edges lead from c' to the same ground vertex.) Let $G_s^*(\varphi)$ be the new graph containing the ground vertices and their connections to clauses and let $\mathcal{G}_\varphi^*$ be its embedding.

Now, we can replace the edges between variables and clauses by transporters in $G_s^*(\varphi)$, as mentioned earlier. We do not substitute for the edges leading to the ground vertices. The edges to the ground vertices ensure that in every labeled embedding satisfying $\widehat{\mathcal{R}}$ there is no pair of transporters such that one is nested inside the other one. (I.e. in the rotation scheme of a variable or a clause any two transporters T_1, T_2 are ordered as follows: a pocket of T_1, the spine of T_1, the second pocket of T_1, a pocket of T_2, the spine of T_2, the second pocket of T_2.)

Let x be a variable of φ that appears in k clauses. Let $c_1, \ldots, c_k$ be the order of the clauses containing x in the rotation scheme $\sigma_{\mathcal{G}_\varphi^*}(x)$. To synchronize the embeddings of the transporters incident to x, we add k new edges $e_1 = (x, y_1), \ldots, e_k = (x, y_k)$ such that $l_{\widehat{\mathcal{R}}}(e_j) = r_{\widehat{\mathcal{R}}}(e_j) = A$ for each $j \in \{1, \ldots, k\}$. In addition, we prescribe the rotation scheme $\sigma_{\widehat{\mathcal{R}}}(x) = (s_1, e_1, s_2, e_2, \ldots, s_k, e_k)$ where s_j is the spine of the transporter connecting x and c_j for each $j \in \{1, \ldots, k\}$.

Notice that if the transporter to the clause c_j has the positive embedding, then the transporter to c_{j+1} must have the positive embedding as well. And if the transporter to c_j has the negative embedding, then the transporter to c_{j-1} has also the negative embedding. From these observations, it follows that there are two possible labeled embeddings of the edges incident to x satisfying $\widehat{\mathcal{R}}$. In the first one, all the incident transporters have the positive embedding. This embedding corresponds to the assignment $x = 1$. In the second one, all the transporters have the negative embedding and it represents the assignment $x = 0$. Both the situations are portrayed in Figure 3.4.

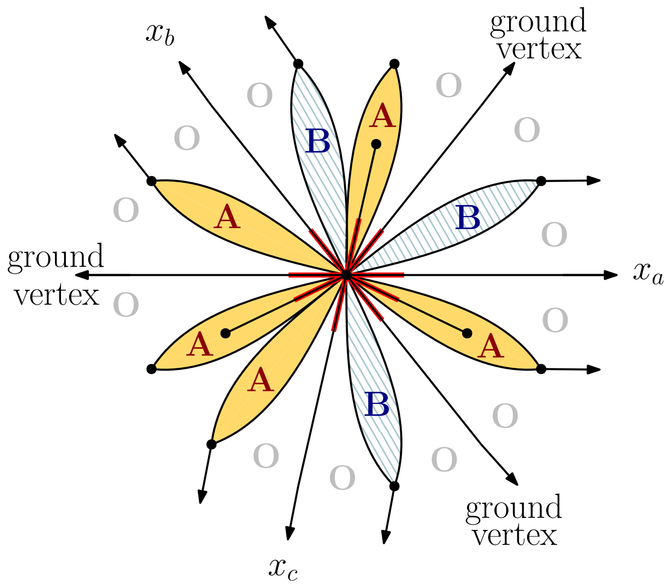


Figure 3.5: The surroundings of a vertex representing the clause $(x_a \vee x_b \vee \neg x_c)$. In this labeled embedding the clause was satisfied by literal x_a.

Let d be a clause containing ℓ literals and let $(x_d^1, g_1, x_d^2, g_2, \ldots x_d^\ell, g_\ell)$ be the order of vertices adjacent to d in the rotation scheme $\sigma_{\mathcal{G}_\varphi^*}(d)$ where $g_1, \ldots g_\ell$ are ground vertices. We add ℓ new edges $e_1' = (d, z_1), \ldots, e_\ell' = (d, z_\ell)$ such that $l_{\widehat{\mathcal{R}}}(e_i') = r_{\widehat{\mathcal{R}}}(e_i') = A$ for each $i \in \{1, \ldots, \ell\}$. Further, we add $(\ell - 1)$ new A-pockets with vertices $\{d, w_1\}, \ldots, \{d, w_{\ell-1}\}$. And finally, we set the rotation scheme $\sigma_{\widehat{\mathcal{R}}}(d) = (e_1', s_1', g_1', e_2', s_2', g_2', \ldots, e_\ell', s_\ell', g_\ell')$ where for each $i \in \{1, \ldots, \ell\}$ s_i' is the spine of the transporter to x_d^i and g_i' is the ground edge to g_i. The situation is illustrated in Figure 3.5.

Observe that in a labeled embedding satisfying $\widehat{\mathcal{R}}$ the edge e_i' must be either in the A-pocket of the transporter from variable x_d^i to d or in one of the $(\ell - 1)$ newly added A-pockets. However, there are only $(\ell - 1)$ of these new pockets, so at least one of the edges must end up in the A-pocket of a transporter. And since the A-pocket precedes the spine of the transporter in the rotation scheme of d then the transporter brings a satisfying assignment to d.

If the graph H has a labeled embedding satisfying $\widehat{\mathcal{R}}$, then following Figure 3.4 we can interpret the embeddings of edges incident to the variable vertices as an assignment ξ of φ. And from the previous observation, we get that each clause of φ is satisfied by ξ.

On the other side, if φ has a satisfying assignment ξ then we can embed the variables according to Figure 3.4. And since ξ is satisfying, then for every clause c we have a literal x_c^* satisfying c. Therefore, the A pocket of the transporter from x_c^* to c precedes its spine in the rotation scheme of c. We put the edge e_*' corresponding to the literal x_c^* in c into this A-pocket and we embed the remaining e' edges into the newly added A-pockets. The resulting labeled embedding of H satisfies $\widehat{\mathcal{R}}$. □

The construction of Theorem 45 produces ERS instances with connected graphs, but not biconnected.

Open problem 46. *Is ERS for instances with labeled opaque edges and bicon-*

nected graphs also NP-complete?

3.3 Existence of a connected face-labeling

Theorem 47. *Let $\mathcal{G}$ be a planar embedding of a graph G and let $\widehat{\mathcal{R}}$ be an embedding restriction of G. Then, it is NP-complete to decide whether there exists a connected face-labeling function g such that the connected labeled embedding $(\mathcal{G}, g)$ satisfies $\widehat{\mathcal{R}}$.*

Proof. The problem is a member of the NP complexity class. Let $F_{\mathcal{G}}$ be the set of faces of $\mathcal{G}$. Apparently $|F_{\mathcal{G}}|$ is polynomial w.r.t. the size of G. It makes sense to consider only face-labeling functions assigning labels from the set A, where A consists of the labels of $\widehat{\mathcal{R}}$ and $|F_{\mathcal{G}}|$ new labels not appearing in $\widehat{\mathcal{R}}$. Thus in polynomial time, we can generate a satisfying face-labeling function g using a non-deterministic algorithm, and then verify that $(\mathcal{G}, g)$ satisfies $\widehat{\mathcal{R}}$.

We describe a reduction from the planar vertex-disjoint paths problem (Problem 41). Let $H = (V_H, S \cup D)$ be an instance of the planar vertex-disjoint paths problem where D is the set of pairs of vertices that we want to connect by vertex-disjoint paths in the graph $H' = (V_H, S)$. We construct a graph G, an embedding $\mathcal{G}$ of G and an embedding restriction $\widehat{\mathcal{R}}$ such that the instance H is satisfiable iff there exists a connected face-labeling function g such that the connected labeled embedding $(\mathcal{G}, g)$ satisfies $\widehat{\mathcal{R}}$.

H' is a planar graph because it is a subgraph of H. Let $\mathcal{H}'$ be a planar embedding of H' and let $(\mathcal{H}', h)$ be a labeled embedding of H' that assigns a unique label to each face of $\mathcal{H}'$. Further, let $d_1, \ldots, d_{|D|}$ be the members of the set D and let $V(D)$ denote the set of vertices of the edges of D. We can assume that the edges of D are disjoint (i.e. there are not two edges in D sharing a vertex). Otherwise the instance H is trivially unsatisfiable. We introduce $|D|$ new distinct labels $\ell_1, \ldots, \ell_{|D|}$ that are also distinct from the range of the face-labeling function h.

We create the graph G and its embedding $\mathcal{G}$ from $\mathcal{H}'$ by widening its edges. We replace every vertex of H' by a polygon such that the number of sides of the polygon is equal to the degree of the replaced vertex. The edges are then replaced by corridors connecting the corresponding polygons in such a way that for each vertex $v \in V_H$ the cyclic order of the corridors leaving the polygon of v is the same as the order of their corresponding edges in the rotation scheme $\sigma_{\mathcal{H}'}(v)$.

Next, we construct the embedding restriction $\widehat{\mathcal{R}}$. All the edges of G are opaque in $\widehat{\mathcal{R}}$ and they are listed in the rotation schemes of both of their vertices. We just need to determine the labels of the incident faces for the edges of G. For each face f of $\mathcal{G}$ we prescribe the same label to all the edges incident to f. If the face f is incident to an edge e from the left side, then we set $l_{\widehat{\mathcal{R}}}(e)$, and if it is incident from the right side, then we set $r_{\widehat{\mathcal{R}}}(e)$. Observe that the face f corresponds either to a face of $\mathcal{H}'$, or to a vertex of H', or to an edge of H'.

(a) If f corresponds to a face f_h of $\mathcal{H}'$, then we prescribe for f the label $h(f_h)$.

(b) If f corresponds to an edge of H', then we use the token $\bigstar$.

(c) If f corresponds to a vertex $v \in V(H')$ and $v \notin V(D)$, then we again prescribe the token $\bigstar$.

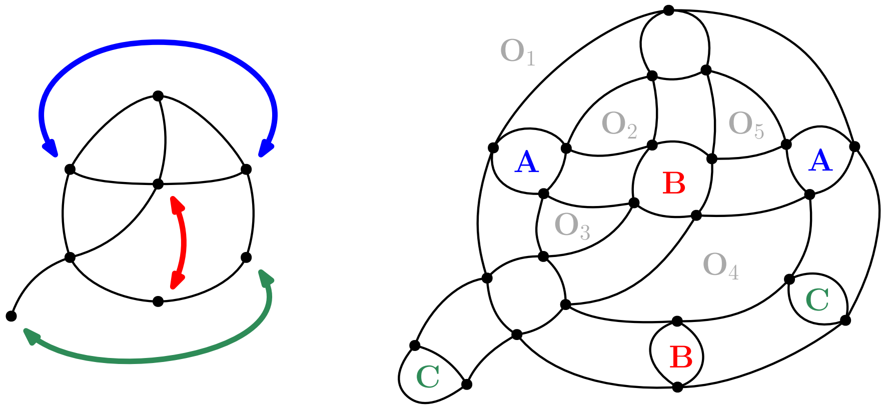


Figure 3.6: An instance of the planar vertex-disjoint paths problem and an equivalent instance of the connected face-labeling problem. The pairs of vertices that should be connected are linked together by color arrows.

(d) If f corresponds to a vertex v such that $v \in d_i$ for some i, then we use the label ℓ_i.

If the instance H is satisfiable, then we can take the set of vertex-disjoint paths $\{P_1, \ldots, P_{|D|}\}$ connecting the pairs of vertices $d_1, \ldots, d_{|D|}$ and for each $i \in \{1, \ldots, |D|\}$ we can label the faces of $\mathcal{G}$ corresponding to P_i by the label ℓ_i. Each of the remaining faces of $\mathcal{G}$ that are not prescribed in $\widehat{\mathcal{R}}$ can be labeled by a new unique label. The described face-labeling is connected and it satisfies $\widehat{\mathcal{R}}$.

If there exists a connected face-labeling function g such that $(\mathcal{G}, g)$ satisfies $\widehat{\mathcal{R}}$, then for each $i \in \{1, \ldots, |D|\}$ we can construct a path P_i connecting the vertices of d_i from the faces of $\mathcal{G}$ labeled by the label ℓ_i. The constructed paths $P_1, \ldots, P_{|D|}$ are vertex-disjoint.

The described construction can be done in polynomial time with respect to the size of the instance H, so the problem is NP-hard. □

4. Polynomial algorithms

In this chapter, we present a polynomial algorithm solving the ERCS instances with labeled opaque edges. In addition, we improve the algorithm to run in linear time for the instances that have not only labeled opaque edges but also anchored borders.

The algorithm is derived in several stages. First, we focus on the SPQR-skeletons and biconnected graphs. After that, we look at the AERCS problem for biconnected graphs. And finally, we show how to process connected and disconnected graphs.

As we proved in the previous chapter, it is NP-hard to find a satisfying connected face-labeling function for a given planar embedding and an embedding restriction. However, this is not the case for the instances with labeled opaque edges.

Lemma 48. *Let $\mathcal{G}$ be an embedding of a graph G and let $\widehat{\mathcal{R}}$ be an embedding restriction of G. Then the following statements hold.*

(i) We can decide in time $\mathcal{O}(|V(G)| + |E(G)|)$ whether there exists a face-labeling function g such that the labeled embedding $\mathcal{G}_L = (\mathcal{G}, g)$ satisfies $\widehat{\mathcal{R}}$.

(ii) If $\widehat{\mathcal{R}}$ has labeled opaque edges then we can decide in time $\mathcal{O}(|V(G)|+|E(G)|)$ whether there exists a connected face-labeling function g such that the labeled embedding $\mathcal{G}_L = (\mathcal{G}, g)$ satisfies $\widehat{\mathcal{R}}$.

Proof. (i) The condition (i) of Definition 11 depends only on the embedding $\mathcal{G}$ and it can be checked in linear time. Next, let $G^o = \left(V(G), E(G) \setminus T_{\widehat{\mathcal{R}}}\right)$ be the subgraph of G containing all the opaque edges and let $\mathcal{G}^o$ be the planar embedding of G^o obtained by restricting $\mathcal{G}$ to G^o. Each face of $\mathcal{G}^o$ consists of several faces of $\mathcal{G}$ inducing a connected subgraph of the dual planar graph to $\mathcal{G}$. A face-labeling function g satisfies the condition (iii) of Definition 11 iff for each face f of $\mathcal{G}^o$ it assigns the same label to all the faces of $\mathcal{G}$ that are contained in f. Therefore, it is enough to check that there is no face of $\mathcal{G}^o$ that has prescribed two different labels in $\widehat{\mathcal{R}}$. And this can be done in linear time.

(ii) If $\widehat{\mathcal{R}}$ has labeled opaque edges and $E(G) \neq T_{\widehat{\mathcal{R}}}$, then every face of $\mathcal{G}^o$ has at least one prescribed label in $\widehat{\mathcal{R}}$. Thus, there exists at most one face-labeling function g fulfilling conditions (ii), (iii) of Definition 11. If the face-labeling function g exists, we must verify that it is connected. The verification can be also done in linear time. For example we can construct the graph H by taking the dual planar graph to $\mathcal{G}$ and removing all its edges $\{f_1, f_2\}$ such that $g(f_1) \neq g(f_2)$. Then, the face-labeling function g is connected iff the number of connected components of H is equal to the size of the range of g.

In case of $E(G) = T_{\widehat{\mathcal{R}}}$ the embedding $\mathcal{G}^o$ has only one face. It means that every labeled embedding satisfying $\widehat{\mathcal{R}}$ is connected. □

4.1 SPQR skeletons

[SQR]-skeletons have only a constant number of planar embeddings. Therefore, it is relatively easy to solve ERCS instances for them. We can even solve the ERS problem for [SQR]-skeletons in linear time.

Lemma 49. *ERCS instances for [SQR]-skeletons and embedding restrictions with labeled opaque edges can be solved in linear time.*

Proof. [SQ]-skeletons have just one planar embedding and R-skeletons either are not planar, or they have two embeddings that are mirror images of each other [6, 17]. Therefore, we can generate all possible embeddings of an [SQR]-skeleton in linear time. Furthermore, following Lemma 48 we can test each of these embeddings for a connected face-labeling function satisfying the embedding restriction. □

ERCS instances with labeled opaque edges for P-skeletons can be also solved in polynomial time. However, the algorithm is more complicated. First, we need to define an additional problem to which we reduce the ERCS instances for P-skeletons.

Problem 50 (Cyclic Suborders Extension (CSE))**.**
Input: *A set D, a set $\mathcal{C}$ of cyclic orders of subsets of D.*
Question: *Is there a cyclic order δ of elements of D such that γ is a subsequence of δ for every $\gamma \in \mathcal{C}$?*

The CSE problem is a generalization of the cyclic ordering problem [7]. We just do not demand that the cyclic orders of the set $\mathcal{C}$ have exactly three elements. Moreover, Galil and Megiddo [7] proved that the cyclic ordering is NP-complete, so the CSE problem is also NP-complete. But we are still able to solve the CSE instances with $|\mathcal{C}| \in \mathcal{O}(1)$ in polynomial time.

Lemma 51. *Let D be a set and $\mathcal{C}$ a set of cyclic orders of subsets of D. Then, CSE$(D, \mathcal{C})$ can be solved in time $\mathcal{O}(|\mathcal{C}| \cdot |D|^{|\mathcal{C}|})$.*

Proof. For a cyclic order β we define C_β as the set of directed edges such that $(b_1, b_2) \in C_\beta$ iff b_2 follows immediately after b_1 in β.

We show that there exists a cyclic order δ of D containing each cyclic order in $\mathcal{C}$ as a subsequence iff for each $\gamma \in \mathcal{C}$ we can select an edge $e_\gamma \in C_\gamma$ such that the graph $G = (D, \bigcup_{\gamma \in \mathcal{C}}(C_\gamma \setminus \{e_\gamma\}))$ is a directed acyclic graph (DAG).

If there exists a cyclic ordering δ of D containing all the suborders in $\mathcal{C}$ as a subsequence then we can remove an arbitrary edge e from C_δ getting a linear ordering δ_{lin} of D. Since δ contains γ as a cyclic subsequence for every $\gamma \in \mathcal{C}$, then there is exactly one edge $e_\gamma \in C_\gamma$ such that δ_{lin} respects all edges of $(C_\gamma \setminus \{e_\gamma\})$. The graph $G = \left(D, \bigcup_{\gamma \in \mathcal{C}}(C_\gamma \setminus \{e_\gamma\})\right)$ has a topological ordering, so G is a DAG.

If for each $\gamma \in \mathcal{C}$ there is an edge $e_\gamma \in C_\gamma$ such that the directed graph $G = (D, \bigcup_{\gamma \in \mathcal{C}}(C_\gamma \setminus \{e_\gamma\}))$ is a DAG, then G has a topological ordering δ_{lin}. We can take the ordering δ_{lin} and make it cyclic by putting its first element right behind the last one. The resulting cyclic order contains all suborders in $\mathcal{C}$ as a subsequence.

Based on this characterization of the satisfiable instances, we derive an algorithm running in time $\mathcal{O}(|\mathcal{C}| \cdot |D|^{|\mathcal{C}|+1})$. There are at most $\mathcal{O}(|D|^{|\mathcal{C}|}$ possible choices of the edges e_γ. The algorithm tries all of them and for each selection it constructs the graph G and it uses Kahn's algorithm [13] to test whether G is a DAG in time $\mathcal{O}(|\mathcal{C}| \cdot |D|)$.

Realize that it is not necessary to try all the possible choices of the edges e_γ. We can take a cyclic order $\gamma^* \in \mathcal{C}$ and remove an arbitrary edge $e_{\gamma^*} \in C_{\gamma^*}$. After that, we look for the edges e_γ only in the remaining $(|\mathcal{C}| - 1)$ cyclic orders. This improve the time complexity of the algorithm to $\mathcal{O}(|\mathcal{C}| \cdot |D|^{|\mathcal{C}|})$. □

There are better algorithms solving the CSE problem than the one described in Lemma 51. For example, if $|\mathcal{C}| \leq 2$, then CSE$(D, \mathcal{C})$ can be decided in time $\mathcal{O}(|D|)$. We present the idea of the linear algorithm without the technical details. Let us assume that $\mathcal{C} = \{\gamma_1, \gamma_2\}$ and that no element of D is repeated in neither γ_1 nor γ_2. If there is no item of D appearing in both the cyclic orders γ_1 and γ_2, then we can just concat γ_1 and γ_2 and arbitrarily fill in the other elements of D. And if there is a non-empty intersection D_I of γ_1 and γ_2, then we verify that γ_1 restricted to the elements of D_I is equal to the restriction of γ_2 to D_I. If γ_1 and γ_2 pass the test, then $\gamma_1 = (d_1, \varphi_1, d_2, \varphi_2, \ldots, d_{|D_I|}, \varphi_{|D_I|})$ and $\gamma_2 = (d_1, \psi_1, \ldots, d_{|D_I|}, \psi_{|D_I|})$, where $d_1, \ldots, d_{|D_I|}$ are the elements of D_I and $\varphi_1, \ldots, \varphi_{|D_I|}$, $\psi_1, \ldots, \psi_{|D_I|}$ are possibly empty sequences of items of the set $(D \backslash D_I)$. The cyclic order $(d_1, \varphi_1, \psi_1, d_2, \varphi_2, \psi_2, \ldots d_{|D_I|}, \varphi_{|D_I|}, \psi_{|D_I|})$ contains both γ_1 and γ_2 as a subsequence and the remaining elements of D can be arbitrarily filled in.

This leads to the question of whether a similar approach can be applied to CSE instances with more than two cyclic suborders to improve the time complexity. Ideally, we would like to solve the CSE problem in linear time with respect to the size of D.

Open problem 52. *Is there a function $f : \mathbb{N} \to \mathbb{N}$ such that CSE$(D, \mathcal{C})$ can be solved in time $\mathcal{O}\left(f(|\mathcal{C}|) \cdot |D|\right)$?*

The P-skeletons are the bottleneck of the entire polynomial algorithm for ERCS. And since we reduce the ERCS instances for P-skeletons to the CSE problem, then any improvement here would also decrease the total time complexity of the algorithm.

Lemma 53. *ERCS instances for P-skeletons and embedding restrictions with labeled opaque edges can be solved in polynomial time.*

Proof. An embedding of a P-skeleton is characterized by the rotation scheme around one of its vertices. Let $G = (\{u, v\}, E)$ be a P-skeleton, $\widehat{\mathcal{R}}$ an embedding restriction of G with labeled opaque edges and L a set of labels appearing in $\widehat{\mathcal{R}}$. Without loss of generality, we assume that all the edges in E are directed from u to v. First, we reduce the instance ERCS$(G, \widehat{\mathcal{R}})$ into a smaller one.

We say that an edge $e \in E$ is a *right-border* edge of a label $\ell \in L$ if $r_{\widehat{\mathcal{R}}}(e) = \ell$ and $l_{\widehat{\mathcal{R}}}(e) \neq \ell$. Similarly, e is a *left-border* edge of ℓ if $l_{\widehat{\mathcal{R}}}(e) = \ell$ and $r_{\widehat{\mathcal{R}}}(e) \neq \ell$. Notice that each right-border edge is a left-border edge of a different label and vice versa. So, it makes sense to use just the term *border* edge without the prefix.

Let E' be the subset of E containing the border edges of $\widehat{\mathcal{R}}$ and the edges anchored in $\sigma_{\widehat{\mathcal{R}}}(u)$ or in $\sigma_{\widehat{\mathcal{R}}}(v)$. Next, let $G' = (\{u, v\}, E')$ be the subgraph of G and $\widehat{\mathcal{R}}'$ the restriction of $\widehat{\mathcal{R}}$ to edges in E'. Then, G has a connected labeled embedding satisfying $\widehat{\mathcal{R}}$ iff the two following conditions hold:

(i) G' has a connected labeled embedding satisfying $\widehat{\mathcal{R}}'$,

(ii) if $|L| \geq 2$ then there is no label appearing in $\widehat{\mathcal{R}}$ and not in $\widehat{\mathcal{R}'}$.

The left-to-right implication is straightforward, we just have to remove edges not in E'. If $|L| \geq 2$ and there is a label in $\widehat{\mathcal{R}}$ not appearing in $\widehat{\mathcal{R}'}$ then this label does not have a right-border edge, so there is no labeled embedding of G satisfying $\widehat{\mathcal{R}}$.

For the second implication we show how to insert edges from $(E \setminus E')$ into a connected labeled embedding $\mathcal{G}'$ of G' satisfying $\widehat{\mathcal{R}'}$. We take edges from $(E \setminus E')$ one by one. Every edge $e \in (E \setminus E')$ satisfies that $l_{\widehat{\mathcal{R}}}(e) = r_{\widehat{\mathcal{R}}}(e)$. If $|L| \geq 2$ and $l_{\widehat{\mathcal{R}}}(e) \neq \bigstar$, then we insert e right in front of the right-border edge of label $l_{\widehat{\mathcal{R}}}(e)$. If $|L| \leq 1$ or $l_{\widehat{\mathcal{R}}}(e) = \bigstar$, then we insert e at an arbitrarily position. In both cases we assign the label of the original divided face to the two newly created faces. This process yields a connected labeled embedding of G satisfying $\widehat{\mathcal{R}}$.

After checking that every label from L appears also in $\widehat{\mathcal{R}'}$, we can continue to solve the instance ERCS$(G', \widehat{\mathcal{R}'})$. We show how to reduce in polynomial time the instance ERCS$(G', \widehat{\mathcal{R}'})$ into a set of CSE instances with a constant number of suborders such that ERCS$(G', \widehat{\mathcal{R}'})$ accepts iff at least one of the CSE instances accepts. The CSE instances with a constant number of suborders can be solved in polynomial time (Lemma 51), so we can decide ERCS$(G', \widehat{\mathcal{R}'})$ also in polynomial time.

If $|L| \leq 1$ then it is enough to find an embedding $\mathcal{G}'$ of G' that respects the rotation schemes $\sigma_{\widehat{\mathcal{R}'}}(u)$ and $\sigma_{\widehat{\mathcal{R}'}}(v)$. That is equivalent to solving the instance CSE$(E', \{\sigma_{\widehat{\mathcal{R}'}}(u), \sigma_{\widehat{\mathcal{R}'}}^{reversed}(v)\})$. If we find $\mathcal{G}'$, then let g' be a face-labeling function that assigns the same label to all the faces of $\mathcal{G}'$. In case of $L \neq \emptyset$, we choose the label from L. The labeled embedding $(\mathcal{G}', g')$ is connected and it satisfies $\widehat{\mathcal{R}'}$.

Further, we assume that $|L| \geq 2$. Then G' has a connected labeled embedding satisfying $\widehat{\mathcal{R}'}$, only if for each label $\ell \in L$ there is exactly one right-border edge of ℓ in E'. Moreover, every right-border edge $e \in E'$ is a left-border edge of a different label ℓ'. Therefore, in every connected labeled embedding of G' satisfying $\widehat{\mathcal{R}'}$ the first right-border edge following e must be the right-border of ℓ'. These observations imply that if G' has a connected labeled embedding satisfying $\widehat{\mathcal{R}'}$, then there exists a cyclic order β containing all the right-border edges of $\widehat{\mathcal{R}'}$ such that an edge e_1 is immediately followed by an edge e_2 iff $l_{\widehat{\mathcal{R}'}}(e_1) = r_{\widehat{\mathcal{R}'}}(e_2)$. Furthermore, β must be a subsequence of $\sigma_{\mathcal{G}'}(u)$ for every connected labeled embedding $\mathcal{G}'$ satisfying $\widehat{\mathcal{R}'}$, because β represents the only connected face-labeling of G' that can possibly satisfy $\widehat{\mathcal{R}'}$.

Next, we want to ensure that for every $\ell \in L$ and every $e \in E'$ such that $l_{\widehat{\mathcal{R}'}}(e) = r_{\widehat{\mathcal{R}'}}(e) = \ell$ the edge e is located between the left-border and the right-border edge of ℓ. Let E_u be the set containing all the edges e anchored in $\sigma_{\widehat{\mathcal{R}'}}(u)$ such that $l_{\widehat{\mathcal{R}'}}(e) \neq \bigstar$. And let τ_u be a cyclic order that is the restriction of $\sigma_{\widehat{\mathcal{R}'}}(u)$ to E_u. We define a set $\mathcal{B}_u$ of cyclic orders of some edges from E' such that at least one of them is a subsequence of $\sigma_{\mathcal{G}'}(u)$ for every connected labeled embedding $\mathcal{G}'$ satisfying $\widehat{\mathcal{R}'}$. If $E_u = \emptyset$ then we put $\mathcal{B}_u = \{\beta\}$.

If $\widehat{\mathcal{R}'}$ prescribes at least two different labels for the incident faces of the edges of E_u, we put $\mathcal{B}_u = \{\beta_u\}$, where the cyclic order β_u is based on τ_u, but for every two consecutive edges e_1, e_2 in τ_u such that $l_{\widehat{\mathcal{R}}}(e_1) \neq r_{\widehat{\mathcal{R}}}(e_2)$ we insert the right-border edge of label $l_{\widehat{\mathcal{R}}}(e_1)$ and the left-border edge of $r_{\widehat{\mathcal{R}}}(e_2)$ between them. (The right-border of $l_{\widehat{\mathcal{R}}}(e_1)$ and the left-border of $r_{\widehat{\mathcal{R}}}(e_2)$ might be the same edge. In that case, we insert only one copy of the edge.) There might be an edge that appears twice in β_u, but in that case, G' has no connected labeled embedding satisfying $\widehat{\mathcal{R}'}$.

Otherwise, $\widehat{\mathcal{R}'}$ prescribes just one unique label ℓ for the edges of E_u. In this case we put $\mathcal{B}_u = \{\beta_u^1, \beta_u^2, \ldots, \beta_u^{|E_u|}\}$, where β_u^i is the cyclic order τ_u in which we insert the right-border and the left-border edge of ℓ just behind the i-th edge of τ_u for a fixed numbering of the edges of τ_u.

Similarly, starting from $\sigma_{\widehat{\mathcal{R}'}}^{reversed}(v)$ we construct the set E_v, the cyclic order τ_v and the set of cyclic orders $\mathcal{B}_v$.

The graph G' has a connected labeled embedding satisfying $\widehat{\mathcal{R}'}$ iff there is a cyclic order σ of edges from E' such that $\sigma_{\widehat{\mathcal{R}'}}(u)$, $\sigma_{\widehat{\mathcal{R}'}}^{reversed}(v)$, β are subsequences of σ and there exist $\beta_u \in \mathcal{B}_u$ and $\beta_v \in \mathcal{B}_v$ that are also subsequences of σ. The left-to-right implication is straightforward. Just plug in the rotation scheme around vertex u for σ. For the second implication, let $\mathcal{G}'$ be the embedding of G' such that $\sigma = \sigma_{\mathcal{G}'}(u)$. Next, let g' be the face-labeling function that assigns the label ℓ to the faces between the left-border and the right-border edges of ℓ in the counter-clockwise direction around u. g' is connected and the labeled embedding $\mathcal{G}'_L = (\mathcal{G}', g')$ fulfills the conditions of Definition 11.

Therefore, ERCS$(G', \widehat{\mathcal{R}'})$ accepts iff there exists $\beta_u \in \mathcal{B}_u$, $\beta_v \in \mathcal{B}_v$ such that CSE$(E', \{\sigma_{\widehat{\mathcal{R}'}}(u), \sigma_{\widehat{\mathcal{R}'}}^{reversed}(v), \beta, \beta_u, \beta_v\})$ accepts. The cyclic orders $\sigma_{\widehat{\mathcal{R}'}}(u)$, $\sigma_{\widehat{\mathcal{R}'}}^{reversed}(v)$, β and the sets $\mathcal{B}_u$, $\mathcal{B}_v$ can be constructed in time $\mathcal{O}(|E|^2)$. There are $\mathcal{O}(|E|^2)$ combinations how to select the cyclic orders β_u, β_v and Lemma 51 yields that the CSE instance with five cyclic suborders are solvable in time $\mathcal{O}(|E|^5)$. Therefore, the total time complexity of the ERCS problem for P-skeletons and embedding restrictions with labeled opaque edges is $\mathcal{O}(|E|^7)$.

The upper bound of the time complexity can be further improved to $\mathcal{O}(|E|^6)$, if in the construction of the sets $\mathcal{B}_u$ and $\mathcal{B}_v$ we do not insert just the left-border and the right-border of the relevant labels, but we also add the entire interval of β clamped between these borders. Then each cyclic sequence in $\mathcal{B}_u$ and $\mathcal{B}_v$ contains β as a subsequence, so it is unnecessary to include β in the queries for the CSE oracle. However, this innovation may cause that the sequences of $\mathcal{B}_u$ and $\mathcal{B}_v$ are too long. But it can be prevented by counting the number of edges we added to τ_u and τ_v. If it is more than the length of β, then there must be a repetitive edge, so we can reject automatically. □

The time complexity $\mathcal{O}(|E|^6)$ of the algorithm for the P-skeletons in the proof of Lemma 53 is quite large. However, we reach this complexity only in some marginal cases when $|\mathcal{B}_u|, |\mathcal{B}_v| \in \Theta(|E|)$. So it is very likely that there is a more efficient algorithm. It is even possible that there is a linear time algorithm.

Open problem 54. *Can the ERCS instances for P-skeletons and embedding restrictions with labeled opaque edges be solved in linear time?*

If the embedding restriction also has anchored borders, then we can upgrade the algorithm from Lemma 53 to run in linear time.

Lemma 55. *ERCS instances for P-skeletons and embedding restrictions with labeled opaque edges and anchored borders can be solved in linear time.*

Proof. If the embedding restriction has anchored borders, then in the construction of Lemma 53 each of the sets $\mathcal{B}_u$ and $\mathcal{B}_v$ contains only one element. Moreover, all the elements of the sequence in $\mathcal{B}_u$ are also in $\sigma_{\widehat{\mathcal{R}}}(u)$ and analogously for $\mathcal{B}_v$ and $\sigma_{\widehat{\mathcal{R}}}^{reversed}(v)$. Similarly, β contains only the items that are both in $\sigma_{\widehat{\mathcal{R}}}(u)$ and $\sigma_{\widehat{\mathcal{R}}}^{reversed}(v)$. So we can in linear time verify that β is a subsequence of both $\sigma_{\widehat{\mathcal{R}}}(u)$ and $\sigma_{\widehat{\mathcal{R}}}^{reversed}(v)$, the sequence in $\mathcal{B}_u$ is a subsequence of $\sigma_{\widehat{\mathcal{R}}}(u)$ and the sequence in $\mathcal{B}_v$ is a subsequence of $\sigma_{\widehat{\mathcal{R}}}^{reversed}(v)$. If one of these tests fails, then we reject. Otherwise, we produce the same answer as $\mathrm{CSE}(E', \{\sigma_{\widehat{\mathcal{R}'}}(u), \sigma_{\widehat{\mathcal{R}'}}^{reversed}(v)\})$, which can be evaluated in linear time. □

4.2 ERCS of biconnected graphs

In this section, we introduce a polynomial algorithm solving ERCS instances with labeled opaque edges and biconnected graphs. The algorithm employs the divide-and-conquer paradigm. It finds a simple pair of split graphs and then it continues to processes each of the split graphs independently.

Lemma 56. *Let G be a biconnected graph, $\{u, v\}$ a separation pair of G and (H, H') a simple pair of split graphs of G with respect to $\{u, v\}$. Then in every embedding $\mathcal{G}$ of G the edges of H form a circular interval in $\sigma_{\mathcal{G}}(u)$ and in $\sigma_{\mathcal{G}}(v)$.*

Proof. Without loss of generality, we assume that H consists of one separation class of G and the virtual edge $e_G^{virt}(H)$. For the sake of contradiction, suppose that G has an embedding $\mathcal{G}$ and there exist edges $e_1, e_2 \neq e_G^{virt}(H)$ of H incident to u and edges $e_3, e_4 \neq e_G^{virt}(H')$ of H' also incident to u such that (e_1, e_3, e_2, e_4) is a subsequence of $\sigma_{\mathcal{G}}(u)$. There is a cycle C' in H' containing edges e_3, e_4, because H' is biconnected. And since H consists of one separation class, there is a cycle C in H containing edges e_1, e_2 and avoiding the vertex v. The cycles C and C' cross each other at the vertex u in the embedding $\mathcal{G}$, so there must be at least one more intersection of C and C'. The split graphs H and H' share only two vertices, so the only candidate for the second intersection is the vertex v. However, C avoids v. □

The condition of Lemma 56 is equivalent to the statement that both $\sigma_{\mathcal{G}}(u)$ and $\sigma_{\mathcal{G}}(v)$ are $(E(H), E(H'))$-non-crossing. Based on this criterion we can recognize some unsatisfiable instances of the ERCS problem.

Definition 57. *Let (H, H') be a pair of split graphs of a graph G with respect to a separation pair $\{u, v\}$ and let $\widehat{\mathcal{R}}$ be an embedding restriction of G. We say that the pair (H, H') is* $\widehat{\mathcal{R}}$-non-crossing *if both $\sigma_{\widehat{\mathcal{R}}}(u)$ and $\sigma_{\widehat{\mathcal{R}}}(v)$ are $(E(H), E(H'))$-non-crossing.*

Corollary 58. *Let G be a biconnected graph, $\widehat{\mathcal{R}}$ an embedding restriction of G, $\{u, v\}$ a separation pair of G and (H, H') a simple pair of split graphs of G with respect to $\{u, v\}$. Then G has a labeled embedding satisfying $\widehat{\mathcal{R}}$, only if (H, H') is $\widehat{\mathcal{R}}$-non-crossing.*

A non-crossing embedding restriction can be divided between the two split graphs. The division process described in the following definition is quite intuitive.

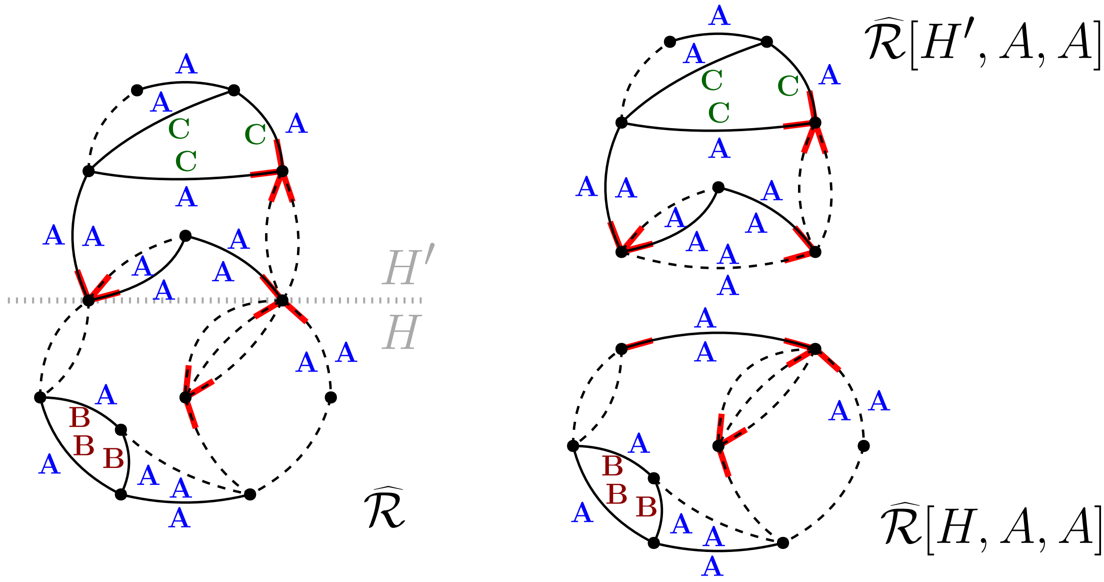


Figure 4.1: An example of a reduction of an embedding restriction $\widehat{\mathcal{R}}$ to split graphs H and H' with parental labels (A, A).

Definition 59. *Let (H, H') be a pair of split graphs of a graph G with respect to a separation pair $\{u, v\}$ and let $\widehat{\mathcal{R}}$ be an embedding restriction of G such that (H, H') are $\widehat{\mathcal{R}}$-non-crossing. Further, let p_l, p_r be labels. Then, the* reduction *of $\widehat{\mathcal{R}}$ to H with* parental labels *(p_l, p_r) is defined as the embedding restriction $\widehat{\mathcal{S}} = \left(\sigma_{\widehat{\mathcal{S}}}, l_{\widehat{\mathcal{S}}}, r_{\widehat{\mathcal{S}}}, T_{\widehat{\mathcal{S}}}\right)$ of H such that:*

(i) *$\widehat{\mathcal{S}}$ inherits the rotation schemes of $\widehat{\mathcal{R}}$ for all the vertices except for u and v, $(\forall w \in (V(H) \setminus \{u, v\}))\ \sigma_{\widehat{\mathcal{S}}}(w) = \sigma_{\widehat{\mathcal{R}}}(w)$.*

(ii) *For $w \in \{u, v\}$ if $\sigma_{\widehat{\mathcal{R}}}(w)$ does not contain any edge of $E(H')$ then $\sigma_{\widehat{\mathcal{S}}}(w) = \sigma_{\widehat{\mathcal{R}}}(w)$, otherwise the interval of edges of H' is replaced in $\sigma_{\widehat{\mathcal{S}}}(w)$ by one occurrence of $e_G^{virt}(H)$.*

(iii) *$\widehat{\mathcal{S}}$ inherits the functions $l_{\widehat{\mathcal{R}}}$ and $r_{\widehat{\mathcal{R}}}$, $(\forall e \in (E(H) \setminus \{e_G^{virt}(H)\}))\ l_{\widehat{\mathcal{S}}}(e) = l_{\widehat{\mathcal{R}}}(e)\ \&\ r_{\widehat{\mathcal{S}}}(e) = r_{\widehat{\mathcal{R}}}(e)$.*

(iv) *$l_{\widehat{\mathcal{S}}}(e_G^{virt}(H)) = p_l$, $r_{\widehat{\mathcal{S}}}(e_G^{virt}(H)) = p_r$.*

(v) *$\widehat{\mathcal{S}}$ also inherits the set of transparent edges of $\widehat{\mathcal{R}}$. In addition, the virtual edge $e_G^{virt}(H)$ is transparent iff there is no path between u and v in G using only the edges of $(E(H') \setminus T_{\widehat{\mathcal{R}}})$, i.e. the opaque edges of H'.*

We use the notation $\widehat{\mathcal{R}}[H, p_l, p_r]$ for the reduction of $\widehat{\mathcal{R}}$ to H with parental labels (p_l, p_r). $\widehat{\mathcal{R}}[H]$ is an abbreviation for $\widehat{\mathcal{R}}[H, \bigstar, \bigstar]$.

For the rest of this section, let G be a biconnected graph, $\widehat{\mathcal{R}}$ an embedding restriction of G with labeled opaque edges, $\{u, v\}$ a separation pair of G and (H, H') a simple $\widehat{\mathcal{R}}$-non-crossing pair of split graphs of G with respect to $\{u, v\}$.

If we get labeled embeddings $\mathcal{H}_L$ and $\mathcal{H}'_L$ of the two split graphs, we can merge them together forming a labeled embedding of G, provided that $\mathcal{H}_L$ and $\mathcal{H}'_L$ have assigned consistent labels around their virtual edges. Furthermore, if $\mathcal{H}_L$

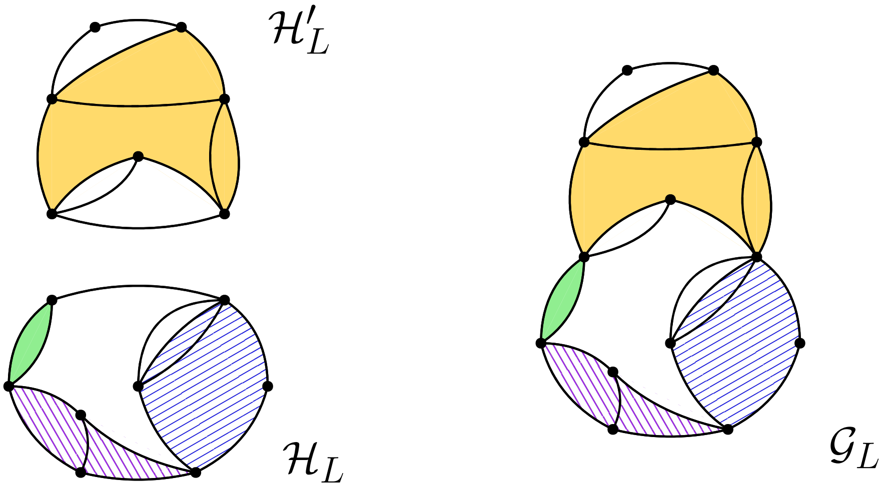


Figure 4.2: Connected labeled embeddings $\mathcal{H}_L$, $\mathcal{H}'_L$ of the two split graphs and the labeled embedding $\mathcal{G}_L$ produced by merging $\mathcal{H}_L$ and $\mathcal{H}'_L$. Notice that $\mathcal{G}_L$ is not connected.

satisfies $\widehat{\mathcal{R}}[H]$ and $\mathcal{H}'_L$ satisfies $\widehat{\mathcal{R}}[H']$, then the labeled embedding of G satisfies $\widehat{\mathcal{R}}$.

Lemma 60. *G has a labeled embedding satisfying $\widehat{\mathcal{R}}$ iff H and H' have labeled embeddings $(\mathcal{H}, h)$, $(\mathcal{H}', h')$ satisfying respectively $\widehat{\mathcal{R}}[H]$ and $\widehat{\mathcal{R}}[H']$ such that*

$$h(l_{\mathcal{H}}(e_G^{virt}(H))) = h'(r_{\mathcal{H}'}(e_G^{virt}(H'))),$$
$$h(r_{\mathcal{H}}(e_G^{virt}(H))) = h'(l_{\mathcal{H}'}(e_G^{virt}(H'))).$$

Proof. (sketch) If H, H' have suitable labeled embeddings $(\mathcal{H}, h)$, $(\mathcal{H}', h')$, we can replace the edge $e_G^{virt}(H)$ in $\mathcal{H}$ by the embedding $\mathcal{H}'$ without the edge $e_G^{virt}(H')$ getting an embedding $\mathcal{G}$ of G. The extra conditions for the face-labeling functions h, h' ensure that there exists a face-labeling function g of $\mathcal{G}$ such that $(\mathcal{G}, g)$ satisfies $\widehat{\mathcal{R}}$.

If G has a labeled embedding $\mathcal{G}_L = (G, g)$ satisfying $\widehat{\mathcal{R}}$ then according to Lemma 56 the edges of H' form a circular interval in $\sigma_{\mathcal{G}}(u)$ and in $\sigma_{\mathcal{G}}(v)$. We can replace the edges of H' in $\mathcal{G}_L$ by $e_G^{virt}(H)$ getting a labeled embedding $\mathcal{H}_L = (\mathcal{H}, h)$ of H. Similarly, replacing the edges of H in $\mathcal{G}_L$ by $e_G^{virt}(H')$ we obtain a labeled embedding $\mathcal{H}'_L = (\mathcal{H}', h')$ of H'. Then, $\mathcal{H}_L$ satisfies $\widehat{\mathcal{R}}[H]$, $\mathcal{H}'_L$ satisfies $\widehat{\mathcal{R}}[H']$ and the conditions for the labels of the virtual edges also hold. □

If we want to create a connected labeled embedding of G, then it is not enough to take connected labeled embeddings of H and H'. We must also ensure, that the labels around the virtual edges remain connected. For example, consider the situation depicted in Figure 4.2. This problem is addressed by the following definition.

Definition 61. *Let $\widehat{\mathcal{S}}$ be an embedding restriction of the split graph H and let ℓ be a label. We say that H is ℓ-*passable *in $\widehat{\mathcal{S}}$ if for each path p in H connecting u and v there exists an edge e of p such that $l_{\widehat{\mathcal{S}}}(e) \in \{\ell, \bigstar\}$ and $r_{\widehat{\mathcal{S}}}(e) \in \{\ell, \bigstar\}$.*

If $\mathcal{H}_L$ and $\mathcal{H}'_L$ are connected labeled embeddings of H and H' satisfying $\widehat{\mathcal{R}}[H]$ and $\widehat{\mathcal{R}}[H']$ such that all the faces incident to the virtual edges $e_G^{virt}(H)$, $e_G^{virt}(H')$

are tagged by a label ℓ and both H and H' are not ℓ-passable, then the labeled embedding created by merging $\mathcal{H}_L$ and $\mathcal{H}'_L$ together is not connected. The faces with label ℓ form two regions.

We are able to check that a split graph H is ℓ-passable in linear time with respect to the size of H. For example we can use the depth first search to test whether there is a path from u to v using only edges e with $l_{\widehat{\mathcal{S}}}(e) \notin \{\ell, \bigstar\}$ or $r_{\widehat{\mathcal{S}}}(e) \notin \{\ell, \bigstar\}$.

Furthermore, if there is a label ℓ present in both $\mathcal{H}_L$ and $\mathcal{H}'_L$, then ℓ must be incident to the virtual edges. Otherwise, the regions tagged by ℓ in $\mathcal{H}_L$ and $\mathcal{H}'_L$ would not connect together. So, let L, L' be the set of labels appearing respectively in $\widehat{\mathcal{R}}[H]$ and $\widehat{\mathcal{R}}[H']$ not including the token $\bigstar$. We differentiate five cases based on the intersection $L \cap L'$.

Lemma 62. *(i) Let $L = \emptyset$ or $L' = \emptyset$. Then G has a (connected) labeled embedding satisfying $\widehat{\mathcal{R}}$ iff respectively H, H' have (connected) labeled embeddings satisfying $\widehat{\mathcal{R}}[H]$, $\widehat{\mathcal{R}}[H']$.*

(ii) If $L, L' \neq \emptyset$, $L \cap L' = \emptyset$, then G has no labeled embedding satisfying $\widehat{\mathcal{R}}$.

(iii) Let $L \cap L' = \{\ell\}$. Then, G has a connected labeled embedding satisfying $\widehat{\mathcal{R}}$ iff respectively H, H' have connected labeled embeddings satisfying restrictions $\widehat{\mathcal{R}}[H, \ell, \ell]$, $\widehat{\mathcal{R}}[H', \ell, \ell]$ and at least one of the split graphs H, H' is ℓ-passable in the corresponding reduction of $\widehat{\mathcal{R}}$.

(iv) Let $L \cap L' = \{\ell_1, \ell_2\}$. Then G has a connected labeled embedding satisfying $\widehat{\mathcal{R}}$ iff H, H' have connected labeled embeddings satisfying either $\widehat{\mathcal{R}}[H, \ell_1, \ell_2]$, $\widehat{\mathcal{R}}[H', \ell_2, \ell_1]$ or $\widehat{\mathcal{R}}[H, \ell_2, \ell_1]$, $\widehat{\mathcal{R}}[H', \ell_1, \ell_2]$.

(v) If $|L \cap L'| \geq 3$ then G has no connected labeled embedding satisfying $\widehat{\mathcal{R}}$.

Proof. (sketch) If G has a connected labeled embedding $\mathcal{G}_L = (\mathcal{G}, g)$, then as in Lemma 60 we can replace the edges of H' in $\mathcal{G}$ by the virtual edge $e_G^{virt}(H)$ getting a connected labeled embedding $\mathcal{H}_L = (\mathcal{H}, h)$ of H satisfying $\widehat{\mathcal{R}}[H]$. Also, the replacement of the edges of H in $\mathcal{G}$ by $e_G^{virt}(H')$ produces a connected labeled embedding $\mathcal{H}'_L = (\mathcal{H}', h')$ of H' satisfying $\widehat{\mathcal{R}}[H']$. Furthermore, the conditions $h(l_{\mathcal{H}}(e_G^{virt}(H))) = h'(r_{\underline{\mathcal{H}'}}(e_G^{virt}(H')))$ and $h(r_{\mathcal{H}}(e_G^{virt}(H))) = h'(l_{\mathcal{H}'}(e_G^{virt}(H')))$ are met. It means that if $\widehat{\mathcal{R}}$ has labeled opaque edges and $L, L' \neq \emptyset$, then the labels assigned to the faces incident to the virtual edges must be present in both L and L'. Next, we observe that if a label x appears in both L and L', then it also has to be assigned to a face incident to the virtual edge $e_G^{virt}(H)$. Otherwise, x cannot form a connected region in $\mathcal{G}_L$. Thus under the mentioned conditions, the set of labels assigned for the faces incident to the virtual edges $e_G^{virt}(H)$ and $e_G^{virt}(H')$ must be identical to the set of labels in the intersection $L \cap L'$. And since there are only two faces incident to $e_G^{virt}(H)$, then $|L \cap L'| \leq 2$.

Moreover, if $L \cap L' = \{\ell\}$ and both H and H' are not ℓ-passable then there exist a path p in H and a path p' in H', forbidding the label ℓ to pass through. These two paths form a cycle c in G separating the two faces of $\mathcal{G}_L$ that are simultaneously incident to the edges of H and H'. We have already observed that these two faces must be tagged by the label ℓ in $\mathcal{G}_L$. However, the label ℓ could not pass through c, so in this case, labeled embedding $\mathcal{G}_L$ cannot be connected.

(iii) If H, H' have connected labeled embeddings $\mathcal{H}_L$, $\mathcal{H}'_L$ satisfying $\widehat{\mathcal{R}}[H,\ell,\ell]$, $\widehat{\mathcal{R}}[H',\ell,\ell]$, then as in Lemma 60 we can join $\mathcal{H}_L$ and $\mathcal{H}'_L$ while omitting the virtual edges in order to get a labeled embedding $\mathcal{G}_L$ of G satisfying $\widehat{\mathcal{R}}$. We need to show that the labeled embedding $\mathcal{G}_L$ is connected. For each label $x \neq \ell$ the faces tagged by x form a connected region in $\mathcal{G}_L$, because x is only present in one of the split graphs and both $\mathcal{H}_L$ and $\mathcal{H}'_L$ are connected. Now let us look at the label ℓ. Although the assignment for the incident faces of the virtual edges $e_G^{virt}(H)$ and $e_G^{virt}(H')$ ensures that the regions tagged by ℓ in $\mathcal{H}_L$ and $\mathcal{H}'_L$ merge together, there can still be two regions tagged by ℓ in $\mathcal{G}_L$. However, it is not the case if at least one of the split graphs in ℓ-passable. If there were two regions tagged by ℓ in $\mathcal{G}_L$, then the border of one of these regions would contain a path p in H and a path p' in H', both of them connecting u and v, and contradicting the ℓ-passable property of H and H'.

(iv) The proof of the second implication is very similar to the case (iii).

(i) Without loss of generality, $L = \emptyset$. Assume that H, H' have (connected) labeled embeddings $\mathcal{H}_L = (\mathcal{H}, h)$, $\mathcal{H}'_L = (\mathcal{H}', h')$ satisfying $\widehat{\mathcal{R}}[H]$, $\widehat{\mathcal{R}}[H']$. Since $\widehat{\mathcal{R}}$ has labeled opaque edges, then there is no opaque path between u and v in H, and so $e_G^{virt}(H')$ is transparent in $\widehat{\mathcal{R}}[H']$. It means that $h'(l_{\mathcal{H}'}(e_G^{virt}(H'))) = h'(r_{\mathcal{H}'}(e_G^{virt}(H')))$. We can define a new face-labeling function h^* of $\mathcal{H}$ assigning the label $h'(l_{\mathcal{H}'}(e_G^{virt}(H')))$ to all its faces. The labeled embedding $(\mathcal{H}, h^*)$ satisfies $\widehat{\mathcal{R}}[H]$ and it is connected. In addition, we can merge it with $\mathcal{H}'_L$ as in case (iii) getting a (connected) labeled embedding of G satisfying $\widehat{\mathcal{R}}$. □

The Lemma 62 encourages us to design a divide-and-conquer style algorithm that distinguishes three non-trivial situations based on the intersection $L \cap L'$. But first, we must deal with the case $|L \cap L'|$, which offers two pairs of subproblems. Solving both the pairs would hurt the time complexity. Luckily, it is unnecessary. We show that at most one option is viable for each pair of simple split graphs.

Lemma 63. *If $L \cap L' = \{\ell_1, \ell_2\}$ and $\widehat{\mathcal{R}}$ has labeled opaque edges, then at most one of the embedding restrictions $\widehat{\mathcal{R}}[H,\ell_1,\ell_2]$ and $\widehat{\mathcal{R}}[H,\ell_2,\ell_1]$ has a satisfying connected labeled embedding.*

Proof. We utilize our observations about the relation of the Eulerian circuits and the ERS problem.

Let $\widehat{\mathcal{S}}$ be an embedding restriction of H with labeled opaque edges and let E_u be the set of the border edges of $\widehat{\mathcal{S}}$ incident to u.

$$E_u = \{\{u, w\} \in E(H) \mid l_{\widehat{\mathcal{S}}}(u, w) \neq r_{\widehat{\mathcal{S}}}(u, w)\}.$$

Further, let us assume that all the edges of E_u are oriented from u outwards.

The edges of E_u are the only edges of $\widehat{\mathcal{S}}$ that are allowed to have different labels assigned to their incident faces. Therefore, H has a labeled embedding satisfying $\widehat{\mathcal{S}}$, only if for each label ℓ

$$\left|\{e \in E_u \mid l_{\widehat{\mathcal{S}}}(e) = \ell\}\right| = \left|\{e \in E_u \mid r_{\widehat{\mathcal{S}}}(e) = \ell\}\right|.$$

Now we consider the restrictions $\widehat{\mathcal{R}}[H,\ell_1,\ell_2]$ and $\widehat{\mathcal{R}}[H,\ell_2,\ell_1]$. These two restrictions differ only in the labels of the edge $e_G^{virt}(H)$. So let us count the border edges of the label ℓ_1.

$$\left|\{e \in E_u \mid l_{\widehat{\mathcal{R}}[H,\ell_1,\ell_2]}(e) = \ell_1\}\right| = \left|\{e \in E_u \mid l_{\widehat{\mathcal{R}}[H,\ell_2,\ell_1]}(e) = \ell_1\}\right| + 1,$$
$$\left|\{e \in E_u \mid r_{\widehat{\mathcal{R}}[H,\ell_1,\ell_2]}(e) = \ell_1\}\right| = \left|\{e \in E_u \mid r_{\widehat{\mathcal{R}}[H,\ell_2,\ell_1]}(e) = \ell_1\}\right| - 1.$$

Evidently, at most one of $\widehat{\mathcal{R}}[H,\ell_1,\ell_2]$ and $\widehat{\mathcal{R}}[H,\ell_2,\ell_1]$ can have the same number of the left and the right borders of ℓ_1. So at most one of them can have a satisfying labeled embedding. □

Algorithm 3: A test that the number of the left-border edges of a label ℓ is equal to the number of the right-border edges.

input : A graph H, an embedding restriction $\widehat{\mathcal{S}}$ of H with labeled opaque edges, a vertex $u \in V(H)$, a label ℓ.

```
function local_Euler_test(H, Ŝ, u, ℓ):
    E_u ← {{u,w} ∈ E(H) | l_Ŝ(u,w) ≠ r_Ŝ(u,w)};
    orient the edges of E_u from u outwards;
    return ( |{e ∈ E_u | l_Ŝ(e) = ℓ}| = |{e ∈ E_u | r_Ŝ(e) = ℓ}| );
```

Moreover, it is possible to recognize which of the embedding restrictions $\widehat{\mathcal{R}}[H,\ell_1,\ell_2]$ and $\widehat{\mathcal{R}}[H,\ell_2,\ell_1]$ is unsatisfiable in linear time with respect to the size of H. For instance, we can count the border edges of the label ℓ_1 incident to the vertex u as illustrated in Algorithm 3.

If the function `local_Euler_test`$(H, \widehat{\mathcal{R}}[H,\ell_1,\ell_2], u, \ell_1)$ returns false, then $\widehat{\mathcal{R}}[H,\ell_1,\ell_2]$ cannot be satisfied. And we have already observed in Lemma 63 that `local_Euler_test` fails at least one of $\widehat{\mathcal{R}}[H,\ell_1,\ell_2]$, $\widehat{\mathcal{R}}[H,\ell_2,\ell_1]$.

Therefore, we can go ahead with the divide-and-conquer notion. Algorithm 4 presents one possible implementation of this idea. It checks whether the input graph is an [SPQR]-skeleton. If so, it uses a specialized polynomial algorithm for the skeletons. Otherwise, it finds a simple pair of split graphs and applies Lemma 62.

The function `ERCS_biconnected`$(G, \widehat{\mathcal{R}})$ just answers whether G has a connected labeled embedding satisfying $\widehat{\mathcal{R}}$. It does not produce satisfying embedding. Nevertheless, it can be easily extended to do so. If G is an [SPQR]-skeleton, then we already know how to find a satisfying embedding. Otherwise, we just merge the embeddings of the split graphs H and H' obtained from the recursive calls like in Lemma 60.

Theorem 64. *Let G be a biconnected graph and $\widehat{\mathcal{R}}$ an embedding restriction of G with labeled opaque edges. Then ERCS$(G, \widehat{\mathcal{R}})$ can be solved in polynomial time.*

Proof. We show that the function `ERCS_biconnected`$(G, \widehat{\mathcal{R}})$ runs in polynomial time with respect to the size of H. The correctness of Algorithm 4 follows immediately from Lemma 62. The only thing worth noting is that in case of $L = \emptyset$ in the instance ERCS$(H, \widehat{\mathcal{R}}[H])$ the edge $e_G^{virt}(H)$ might be opaque and unlabeled. However, this is not an issue, because it is the only opaque edge of H, so we can make it transparent without consequences.

Algorithm 4: A polynomial algorithm for ERCS of biconnected graphs.

input : A biconnected graph G, an embedding restriction $\widehat{\mathcal{R}}$ of G with labeled opaque edges.

function `ERCS_biconnected`$(G, \widehat{\mathcal{R}})$:
  if *G is an [SPQR]-skeleton* **: return** `ERCS_skeleton`$(G, \widehat{\mathcal{R}})$;
  $\{u, v\} \leftarrow$ a separation pair of G;
  $(H, H') \leftarrow$ a simple pair of split graphs w.r.t. $\{u, v\}$;
  if *(H, H') is not $\widehat{\mathcal{R}}$-non-crossing* **: return** false;
  $L \leftarrow \{l_{\widehat{\mathcal{R}}}(e), r_{\widehat{\mathcal{R}}}(e) \mid e \in E(H) \cap E(G)\} \setminus \{\bigstar\}$;
  $L' \leftarrow \{l_{\widehat{\mathcal{R}}}(e), r_{\widehat{\mathcal{R}}}(e) \mid e \in E(H') \cap E(G)\} \setminus \{\bigstar\}$;
  if $|L \cap L'| \geq 3$ **: return** false;
  if $|L| > 0$ **and** $|L'| > 0$ **and** $|L \cap L'| = 0$ **: return** false;
  if $|L| = 0$ **or** $|L'| = 0$ **:**
    return (`ERCS_biconnected`$(H, \widehat{\mathcal{R}}[H])$ **and** `ERCS_biconnected`$(H', \widehat{\mathcal{R}}[H'])$);
  if $|L \cap L'| = \{\ell\}$ **:**
    if *both H and H' are not ℓ-passable* **: return** false;
    return (`ERCS_biconnected`$(H, \widehat{\mathcal{R}}[H, \ell, \ell])$ **and**
        `ERCS_biconnected`$(H', \widehat{\mathcal{R}}[H', \ell, \ell])$);
  $\{\ell_1, \ell_2\} \leftarrow L \cap L'$;
  if `local_Euler_test`$(H, \widehat{\mathcal{R}}[H, \ell_1, \ell_2], u, \ell_1)$ **:**
    return (`ERCS_biconnected`$(H, \widehat{\mathcal{R}}[H, \ell_1, \ell_2])$ **and**
        `ERCS_biconnected`$(H', \widehat{\mathcal{R}}[H', \ell_2, \ell_1])$)
  else if `local_Euler_test`$(H, \widehat{\mathcal{R}}[H, \ell_2, \ell_1], u, \ell_1)$ **:**
    return (`ERCS_biconnected`$(H, \widehat{\mathcal{R}}[H, \ell_2, \ell_1])$ **and**
        `ERCS_biconnected`$(H', \widehat{\mathcal{R}}[H', \ell_1, \ell_2])$)
  return false;

Using clever data structures we ensure that `ERCS_biconnected`$(G, \widehat{\mathcal{R}})$ spends time $\mathcal{O}(|V(G)| + |E(G)|)$ on one recursion level except for the call of the function `ERCS_skeleton`.

First, we construct the SPQR-tree of G in time $\mathcal{O}(|V(G)| + |E(G)|)$. The tree allows us to recognize in constant time whether G is an [SPQR]-skeleton. Moreover, it also helps us to find the separation pair and the simple pair of split graphs. The separation pair (H, H') can also be created in constant time by editing the graph G in place.

Next, if we represent the rotation schemes as bidirectional cyclic linked lists, then we are able to verify that (H, H') is $\widehat{\mathcal{R}}$-non-crossing in time $\mathcal{O}(|E(H)|)$. We just check that there is at most one edge of H followed by an edge of H' in $\sigma_{\widehat{\mathcal{R}}}(u)$ and $\sigma_{\widehat{\mathcal{R}}}(v)$. It also allows us to quickly construct the reductions $\widehat{\mathcal{R}}[H]$ and $\widehat{\mathcal{R}}[H']$ except for determining the opacity of the virtual edges. For that we have to make two searches (e.g. DFS) running in $\mathcal{O}(|V(H)|+|E(H)|)$ and $\mathcal{O}(|V(H')|+|E(H')|)$ respectively.

The last problems are the sets of labels L and L'. We assume that we can afford one global array indexed by the labels. On each recursion level, we first raise flags on the positions of the labels of L'. Then we do the same for the labels of L while noticing, which labels were already in L'. Those are the labels in the intersection. Finally, we again go through the labels of L and L' and remove the flags, so that the array can be used again in another recursive call. This operation costs us $\mathcal{O}(|V(G)| + |E(G)|)$.

The depth of the recursion is $\mathcal{O}(|V(G)|)$, because the SPQR-tree of a split graph has at least one node less than the SPQR-tree of the original graph. Furthermore, on each recursion level the total size of the subproblems is $\mathcal{O}(|V(G)| + |E(G)|)$. So except for the calls of `ERCS_skeleton` the algorithm runs in time $\mathcal{O}(|V(G)| \cdot (|V(G)| + |E(G)|))$. Unfortunately, the best algorithm for P-skeletons we have is quite slow. So the total time complexity of `ERCS_biconnected(`$G, \widehat{\mathcal{R}}$`)` is $\mathcal{O}(|E(G)|^6)$. □

Theorem 65. *Let G be a biconnected graph and $\widehat{\mathcal{R}}$ an embedding restriction for G with labeled opaque edges and anchored borders. Then ERCS$(G, \widehat{\mathcal{R}})$ can be solved in linear time.*

Proof. (sketch) If $\widehat{\mathcal{R}}$ has also anchored borders, then we can solve the instances for P-skeletons in linear time. So the total time spend by `ERCS_biconnected(`$G, \widehat{\mathcal{R}}$`)` in the [SPQR]-skeletons is $\mathcal{O}(|V(G)| + |E(G)|)$.

In addition, we can reduce the time complexity by selecting the simple pair of split graphs (H, H') such that H is an [SPR]-skeleton. Then, we just need to spend time $\mathcal{O}(|V(H)| + |E(H)|)$ in each non-skeleton iteration. This time can be charged to the skeleton H, so the total complexity is $\mathcal{O}(|V(G)| + |E(G)|)$. However, it requires a few changes in the implementation.

Firstly, the SPQR-tree is constructed only once and in each non-skeleton iteration we tear off the leaf node corresponding to the split graph H. Secondly, we do not compute the transparency of the virtual edge $e_G^{virt}(H)$. If $L' = \emptyset$, then $e_G^{virt}(H)$ is transparent, otherwise, it is opaque. Thirdly, the global array indexed by the labels contains for each label ℓ the value $\left|\{e \in E(G) \mid l_{\widehat{\mathcal{R}}}(e) = \ell\}\right| + \left|\{e \in E(G) \mid r_{\widehat{\mathcal{R}}}(e) = \ell\}\right|$. Before solving the instance for H' we subtract the contribution of H and add one for the labels of the virtual edge $e_G^{virt}(H')$. We can still use the array to construct the intersection $L \cap L'$. But, we further need one integer that contains the total sum of the values in the array. This integer allows us to quickly recognize when L' is empty. Finally, we do not test that either H or H' is ℓ-passable. Then, the algorithm can return some fake positive answers. However, we filter those by fully constructing the satisfying labeled embedding and verifying that it is really connected. □

4.3 AERCS of biconnected graphs

The instances of AERCS with labeled opaque edges and biconnected graphs can be also solved in polynomial time. The algorithm is based on the same recursive idea we used in the ERCS problem. However, labels in the rotation schemes bring complications that we did not encounter in the ERCS. Fortunately, these complications occur only in instances without anchored borders.

Theorem 66. *Let G be a biconnected graph and $\widehat{\mathcal{A}}$ an augmented embedding restriction of G with labeled opaque edges and anchored borders. Then, the instance AERCS$(G, \widehat{\mathcal{A}})$ can be solved in linear time.*

Proof. We can remove labels from the rotation schemes of $\widehat{\mathcal{A}}$ creating an equivalent ERCS instance, that can be solved in linear time. Let $w \in V(G)$ be a vertex such that the rotation scheme $\rho_{\widehat{\mathcal{A}}}(w)$ contains a label. If there are no border edges incident to w, then all the faces around w must be tagged by the same label. Therefore, after verifying that $\rho_{\widehat{\mathcal{A}}}(w)$ contains at most one unique label ℓ we remove all the occurrences of ℓ from $\rho_{\widehat{\mathcal{A}}}(w)$, plus we select an edge $e \in E(G)$ incident to w and set $l_{\widehat{\mathcal{A}}}(e) = \ell$. If $\rho_{\widehat{\mathcal{A}}}(w)$ contains at least two different labels, then we reject.

Otherwise, there are some border edges incident to w. In that case, we just check that the labels in $\rho_{\widehat{\mathcal{A}}}(w)$ are consistent with the placement of the border edges before removing them. Let $e_1, e_2 \in E(G)$ be border edges incident to w such that there are no other border edges between e_1 and e_2 in $\rho_{\widehat{\mathcal{A}}}(w)$ in the counter-clockwise direction. Then the faces enclosed between e_1 and e_2 must be tagged by the same label. So if $l_{\widehat{\mathcal{A}}}(e_1) \neq r_{\widehat{\mathcal{A}}}(e_2)$ or if there is a wrong label in $\rho_{\widehat{\mathcal{A}}}(w)$ between e_1 and e_2, then there is no labeled embedding of G satisfying $\widehat{\mathcal{A}}$. □

To simplify the situation we consider only AERCS instances where the augmented embedding restriction contains at least two different labels. The cases with at most one unique label can be solved as simple ERCS instances. If there is no label in the augmented embedding restriction, then the reduction to ERCS is straightforward. Else if there is just one unique label ℓ, we can ignore it, solve the problem as an ERCS instance and then tag every face of the satisfying embedding by ℓ.

Further, we observe that each vertex of a graph G is incident to at most $|E(G)|$ edges and $|E(G)|$ faces. Therefore, for an augmented embedding restriction $\widehat{\mathcal{A}}$ of G the instance AERCS$(G, \widehat{\mathcal{A}})$ is satisfiable, only if there is no vertex $w \in V(G)$ such that the rotation scheme $\rho_{\widehat{\mathcal{A}}}(w)$ has more than $2 \cdot |E(G)|$ elements. We automatically reject for the instances with too long rotation schemes, so we can assume that their sizes are at most linear with respect to the number of edges.

Similarly to the ERCS problem, we first show how to solve AERCS for [SPQR]-skeletons.

Lemma 67. *AERCS instances for [SPQR]-skeletons and augmented embedding restrictions with labeled opaque edges can be solved in polynomial time.*

Proof. [SQR]-skeletons have a limited number of labeled embeddings and we can test all of them as we did in Lemma 49. And since the augmented rotation schemes are non-crossing, we can perform the test in linear time applying Lemma 7.

For P-skeletons we can replace each label appearing in the rotation schemes by a new edge that has this label prescribed for both of its incident faces. After that, we can proceed as in Lemma 53. □

4.3.1 Division on a separation pair

If a biconnected graph is not an [SPQR]-skeleton then according to Lemma 29 it has a separation pair. So, let G be a biconnected graph with a separation pair $\{u, v\}$ and let (H, H') be a simple pair of split graphs with respect to $\{u, v\}$. Further, let $\widehat{\mathcal{A}}$ be an augmented embedding restriction of G with labeled opaque edges containing at least two different labels.

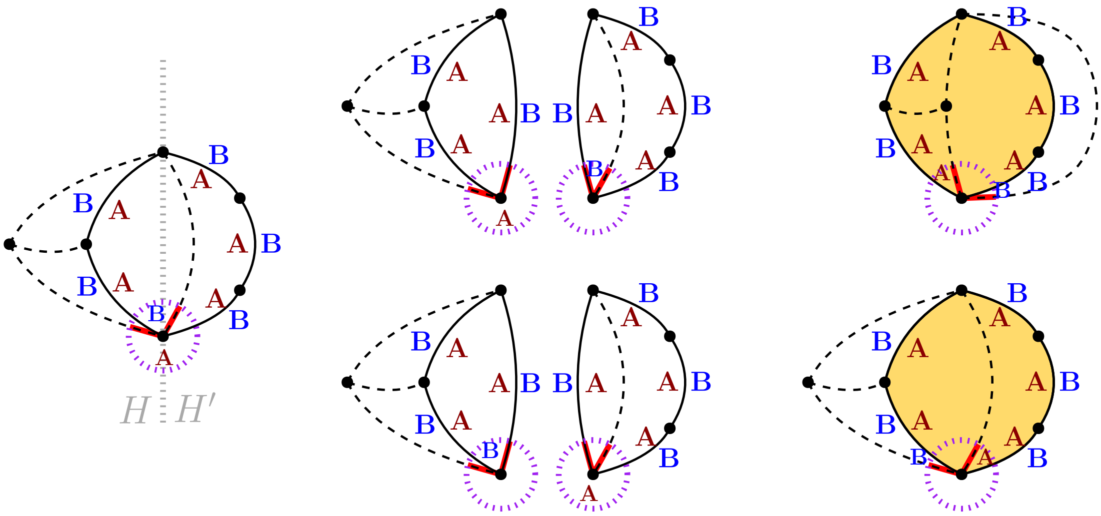


Figure 4.3: An augmented embedding restriction and its two possible divisions to the split graphs H, H'. Both of these divisions have satisfying connected labeled embeddings.

Definition 68. *Let $\widehat{\mathcal{R}}$ be the embedding restriction obtained from $\widehat{\mathcal{A}}$ by omitting labels in the rotation schemes $\rho_{\widehat{\mathcal{A}}}$. A pair of split graphs (H, H') is $\widehat{\mathcal{A}}$-*non-crossing *if it is $\widehat{\mathcal{R}}$-non-crossing.*

Corollary 69. *AERCS$(G, \widehat{\mathcal{A}})$ can be satisfied only if (H, H') is $\widehat{\mathcal{A}}$-non-crossing.*

We proceed as we did in the ERCS problem. We derive two augmented embedding restrictions from $\widehat{\mathcal{A}}$, $\widehat{\mathcal{B}}$ for H and $\widehat{\mathcal{B}}'$ for H', dividing the problem into two subproblems.

This division of $\widehat{\mathcal{A}}$ has to split in two the rotation schemes $\rho_{\widehat{\mathcal{A}}}(u)$ and $\rho_{\widehat{\mathcal{A}}}(v)$. However, there can be multiple options on how to do it. For instance, look at the highlighted separation vertex of the augmented embedding restriction illustrated in Figure 4.3.

We define a new concept that covers all the possible divisions.

Definition 70. *Let (H, H') be a pair of split graphs of a graph G with respect to a separation pair $\{u, v\}$ and let $\widehat{\mathcal{A}}$ be an augmented embedding restriction of G such that (H, H') are $\widehat{\mathcal{A}}$-non-crossing. Further, let p_L, p_R be labels. Then, the pair of augmented embedding restrictions $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ is a (p_L, p_R)-*division *of $\widehat{\mathcal{A}}$ with respect to (H, H') if:*

(i) Except for u and v the restrictions $\widehat{\mathcal{B}}$ and $\widehat{\mathcal{B}}'$ inherit the rotation schemes of $\widehat{\mathcal{A}}$,
$(\forall w \in (V(H) \setminus \{u, v\}))\ \rho_{\widehat{\mathcal{B}}}(w) = \rho_{\widehat{\mathcal{A}}}(w)$,
$(\forall w \in (V(H') \setminus \{u, v\}))\ \rho_{\widehat{\mathcal{B}}'}(w) = \rho_{\widehat{\mathcal{A}}}(w)$.

(ii) For $w \in \{u, v\}$ let φ, φ' is a partition of $\rho_{\widehat{\mathcal{A}}}(w)$ into two circular intervals such that φ does not contain any edge of $E(H')$ and φ' does not contain any edge of $E(H)$. ($\rho_{\widehat{\mathcal{A}}}(w) = (\varphi, \varphi')$.)
If φ' is empty then $\rho_{\widehat{\mathcal{B}}}(w) = (\varphi)$. Otherwise, $\rho_{\widehat{\mathcal{B}}}(w) = (\varphi, e_G^{virt}(H))$.
If φ is empty then $\rho_{\widehat{\mathcal{B}}'}(w) = (\varphi')$. Otherwise, $\rho_{\widehat{\mathcal{B}}'}(w) = (e_G^{virt}(H'), \varphi')$.

(iii) $\widehat{\mathcal{B}}$ *and* $\widehat{\mathcal{B}}'$ *inherit the functions* $l_{\widehat{\mathcal{A}}}$ *and* $r_{\widehat{\mathcal{A}}}$ *for all the relevant edges,*
$(\forall e \in (E(H) \setminus \{e_G^{virt}(H)\}))\ l_{\widehat{\mathcal{B}}}(e) = l_{\widehat{\mathcal{A}}}(e)\ \&\ r_{\widehat{\mathcal{B}}}(e) = r_{\widehat{\mathcal{A}}}(e)$,
$(\forall e \in (E(H') \setminus \{e_G^{virt}(H')\}))\ l_{\widehat{\mathcal{B}}'}(e) = l_{\widehat{\mathcal{A}}}(e)\ \&\ r_{\widehat{\mathcal{B}}'}(e) = r_{\widehat{\mathcal{A}}}(e)$.

(iv) $l_{\widehat{\mathcal{B}}}(e_G^{virt}(H)) = p_L$, $r_{\widehat{\mathcal{B}}}(e_G^{virt}(H)) = p_R$,
$l_{\widehat{\mathcal{B}}'}(e_G^{virt}(H')) = p_R$, $r_{\widehat{\mathcal{B}}'}(e_G^{virt}(H')) = p_L$.

(v) $\widehat{\mathcal{B}}$ *and* $\widehat{\mathcal{B}}'$ *also inherit the set of transparent edges of* $\widehat{\mathcal{A}}$. *In addition, the virtual edge* $e_G^{virt}(H)$ *is transparent iff there is no path between* u *and* v *in* G *using only the edges of* $(E(H')\setminus T_{\widehat{\mathcal{A}}})$. *And similarly,* $e_G^{virt}(H')$ *is transparent iff there is no path between* u *and* v *in* G *using only the edges of* $(E(H)\setminus T_{\widehat{\mathcal{A}}})$.

If $p_L = p_R$, *then we use a shorter notation* p_L*-division instead of* (p_L, p_L)*-division.*

The division of an augmented embedding restriction has a function similar to the reduction of an embedding restriction (Definition 59). But unlike the reduction, there usually is not one unique division of $\widehat{\mathcal{A}}$ with respect to (H, H'). The cause of the ambiguity is the partition of the rotation schemes $\rho_{\widehat{\mathcal{A}}}(u)$ and $\rho_{\widehat{\mathcal{A}}}(v)$ into $\rho_{\widehat{\mathcal{B}}}(u)$, $\rho_{\widehat{\mathcal{B}}'}(u)$ and $\rho_{\widehat{\mathcal{B}}}(v)$, $\rho_{\widehat{\mathcal{B}}'}(v)$. Nevertheless, an analogy of Lemma 60 still holds for AERS.

Lemma 71. *The graph* G *has a labeled embedding satisfying* $\widehat{\mathcal{A}}$ *iff there exists a* $\bigstar$*-division* $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ *of* $\widehat{\mathcal{A}}$ *w.r.t.* (H, H') *such that* H, H' *have labeled embeddings* $(\mathcal{H}, h)$, $(\mathcal{H}', h')$ *satisfying* $\widehat{\mathcal{B}}$, $\widehat{\mathcal{B}}'$ *where* $h(l_{\mathcal{H}}(e_G^{virt}(H))) = h'(r_{\mathcal{H}'}(e_G^{virt}(H')))$ *and* $h(r_{\mathcal{H}}(e_G^{virt}(H))) = h'(l_{\mathcal{H}'}(e_G^{virt}(H')))$.

Proof. (sketch) If there exists such a $\bigstar$-division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ and labeled embeddings $(\mathcal{H}, h)$, $(\mathcal{H}', h')$, then we can merge these labeled embeddings like in Lemma 60 into a labeled embedding $(\mathcal{G}, g)$ of G that satisfies $\widehat{\mathcal{A}}$.

On the other hand, if G has a labeled embedding $(\mathcal{G}, g)$ satisfying $\widehat{\mathcal{A}}$, then we can split it into labeled embeddings $(\mathcal{H}, h)$ and $(\mathcal{H}', h')$ of H and H'. These embeddings satisfies the conditions for the labels incident to the virtual edges. The splitting operation cut the augmented rotation schemes $\rho_{(\mathcal{G},g)}(u)$ and $\rho_{(\mathcal{G},g)}(v)$ in two. Since $(\mathcal{G}, g)$ satisfies $\widehat{\mathcal{A}}$, we can map $\rho_{\widehat{\mathcal{A}}}(w)$ on $\rho_{(\mathcal{G},g)}(w)$ for $w \in \{u, v\}$. Then, we cut $\rho_{\widehat{\mathcal{A}}}(w)$ at the positions corresponding to the partition of $\rho_{(\mathcal{G},g)}(w)$. This way we produce a $\bigstar$-division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ of $\widehat{\mathcal{A}}$ such that $(\mathcal{H}, h)$ satisfies $\widehat{\mathcal{B}}$ and $(\mathcal{H}', h')$ satisfies $\widehat{\mathcal{B}}'$. □

We can also generalize Lemma 62 for the AERCS problem. However, we have to be careful while looking for an alternative of the sets of labels L and L' appearing in Lemma 62, because the division of $\widehat{\mathcal{A}}$ does not refer to one unique object.

We utilize our assumption that $\widehat{\mathcal{A}}$ contains at least two different labels.

Lemma 72. *AERCS*$(G, \widehat{\mathcal{A}})$ *can be satisfied only if for every label* ℓ *of* $\widehat{\mathcal{A}}$ *there is an edge* e *such that either* $l_{\widehat{\mathcal{A}}}(e) = \ell \neq r_{\widehat{\mathcal{A}}}(e)$ *or* $l_{\widehat{\mathcal{A}}}(e) \neq \ell = r_{\widehat{\mathcal{A}}}(e)$.

Proof. Otherwise, the label ℓ cannot be separated from the other labels in a labeled embedding satisfying $\widehat{\mathcal{A}}$. □

Let $\widehat{\mathcal{R}}$ be the embedding restriction obtained from $\widehat{\mathcal{A}}$ by omitting labels in the rotation schemes. Further, let L_E and L'_E be set of labels appearing in $\widehat{\mathcal{R}}[H]$ and $\widehat{\mathcal{R}}[H']$. They can be equivalently defined as

$$L_E = \{l_{\widehat{\mathcal{A}}}(e) \mid e \in E(H)\} \cup \{r_{\widehat{\mathcal{A}}}(e) \mid e \in E(H)\} \setminus \{\bigstar\},$$
$$L'_E = \{l_{\widehat{\mathcal{A}}}(e) \mid e \in E(H')\} \cup \{r_{\widehat{\mathcal{A}}}(e) \mid e \in E(H')\} \setminus \{\bigstar\}.$$

Lemma 73. *(i) If $L_E, L'_E \neq \emptyset$, $L_E \cap L'_E = \emptyset$, then G has no labeled embedding satisfying $\widehat{\mathcal{A}}$.*

(ii) If $|L_E \cap L'_E| \geq 3$, then G has no connected labeled embedding satisfying $\widehat{\mathcal{A}}$.

(iii) Let $L_E \cap L'_E = \{\ell\}$. Then, G has a connected labeled embedding satisfying $\widehat{\mathcal{A}}$ iff there exists an ℓ-division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ of $\widehat{\mathcal{A}}$ such that H, H' have connected labeled embeddings satisfying the restrictions $\widehat{\mathcal{B}}$, $\widehat{\mathcal{B}}'$ and at least one of H, H' is ℓ-passable in $\widehat{\mathcal{B}}$, $\widehat{\mathcal{B}}'$.

(iv) Let $L_E \cap L'_E = \{\ell_1, \ell_2\}$. Then, G has a connected labeled embedding satisfying $\widehat{\mathcal{A}}$ iff there exists an (ℓ_1, ℓ_2)-division or an (ℓ_2, ℓ_1)-division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ of $\widehat{\mathcal{A}}$ such that H, H' have connected labeled embeddings satisfying $\widehat{\mathcal{B}}$, $\widehat{\mathcal{B}}'$.

(v) Let $L_E = \emptyset$ or $L'_E = \emptyset$. Then, G has a connected labeled embedding satisfying $\widehat{\mathcal{A}}$ iff there exist a label $\ell \in (L_E \cup L'_E)$ and an ℓ-division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ of $\widehat{\mathcal{A}}$ such that H, H' have connected labeled embeddings satisfying the restrictions $\widehat{\mathcal{B}}$, $\widehat{\mathcal{B}}'$.

Proof. (sketch) The points (i), (ii) are corollary of Lemmata 25 and 62.

Connected labeled embeddings of H, H' satisfying $\widehat{\mathcal{B}}$, $\widehat{\mathcal{B}}'$ can be merged utilizing Lemma 71. The resulting labeled embedding is also connected.

Further, Lemma 71 implies that a connected labeled embedding $\mathcal{G}_L$ of G satisfying $\widehat{\mathcal{A}}$ can be spit into connected labeled embeddings of H and H' satisfying a $\bigstar$-division of $\widehat{\mathcal{A}}$. The faces incident to $e_G^{virt}(H)$ and $e_G^{virt}(H')$ has to be labeled consistently. The only candidates are the labels from the lemma statement. □

4.3.2 Universal divisions

We can design a simple recursive algorithm with exponential runtime that solves AERCS for biconnected graphs. The pseudocode of the algorithm is presented as Algorithm 5. It receives on its input a graph G and an augmented embedding restriction $\widehat{\mathcal{A}}$ with labeled opaque edges. If G is an [SPQR]-skeleton, then the algorithm answers based on Lemma 67 in polynomial time. Otherwise, it finds a separation pair $\{u, v\}$ and a simple pair of its split graphs (H, H'). After that, it applies Lemma 73 and it either rejects or it tries all the possible divisions of $\widehat{\mathcal{A}}$ mentioned in the relevant case of the lemma.

There are $\mathcal{O}(|E(G)|^2)$ possibilities how to split the rotation schemes $\rho_{\widehat{\mathcal{A}}}(u)$ and $\rho_{\widehat{\mathcal{A}}}(v)$, so in total there are at most $\mathcal{O}(|E(G)|^4)$ divisions of $\widehat{\mathcal{A}}$ with respect to (H, H'). The depth of the recursion of the function `AERCS_exp_I`$(G, \widehat{\mathcal{A}})$ is again bounded by $\mathcal{O}(|V(G)|)$ as in `ERCS_biconnected`. So the time complexity of Algorithm 5 is $|E(G)|^{\mathcal{O}(|V(G)|)}$.

Algorithm 5: An exponential algorithm for AERCS of biconnected graphs.

input : A biconnected graph G, an augmented embedding restriction $\widehat{\mathcal{A}}$ of G with labeled opaque edges.

```
function AERCS_exp_I(G, Â):
    if Â contains at most 1 label :
        return ERCS_biconnected(G, omit_labels(Â))
    if G is an [SPQR]-skeleton : return AERCS_skeleton(G, Â);
    {u, v} ← a separation pair of G;
    (H, H′) ← a simple pair of split graphs w.r.t. {u, v};
    if (H, H′) is not Â-non-crossing : return false;
    L_E ← {l_Â(e), r_Â(e) | e ∈ E(H) ∩ E(G)} \ {★};
    L′_E ← {l_Â(e), r_Â(e) | e ∈ E(H′) ∩ E(G)} \ {★};
    if |L_E ∩ L′_E| ≥ 3 : return false;
    if |L_E| > 0 and |L′_E| > 0 and |L_E ∩ L′_E| = 0 : return false;
    if |L ∩ L′| = {ℓ} :
        if both H and H′ are not ℓ-passable : return false;
        foreach ℓ-division (B̂, B̂′) of Â w.r.t. (H, H′) :
            if AERCS_exp_I(H, B̂) and AERCS_exp_I(H′, B̂′) :
                return true;
    else if |L_E| = 0 or |L′_E| = 0 :
        foreach label ℓ ∈ (L_E ∪ L′_E) :
            foreach ℓ-division (B̂, B̂′) of Â w.r.t. (H, H′) :
                if AERCS_exp_I(H, B̂) and AERCS_exp_I(H′, B̂′) :
                    return true;
    else
        {ℓ1, ℓ2} ← L ∩ L′;
        foreach (ℓ1, ℓ2)-division (B̂, B̂′) of Â w.r.t. (H, H′) :
            if AERCS_exp_I(H, B̂) and AERCS_exp_I(H′, B̂′) :
                return true;
        foreach (ℓ2, ℓ1)-division (B̂, B̂′) of Â w.r.t. (H, H′) :
            if AERCS_exp_I(H, B̂) and AERCS_exp_I(H′, B̂′) :
                return true;
    return false;
```

Our goal is an algorithm running in polynomial time. And a polynomial algorithm cannot afford to try all the possible divisions $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ of $\widehat{\mathcal{A}}$. Fortunately, it is not necessary to examine all the divisions. Some of them can never be satisfied. For example, consider the following lemma.

Lemma 74. *Let* $X = E(H) \cup (L_E \setminus L'_E)$ *and* $X' = E(H') \cup (L'_E \setminus L_E)$. *If* $L_E \neq \emptyset$, *then* $\widehat{\mathcal{B}}$ *can be satisfied only if* $\rho_{\widehat{\mathcal{B}}}(u)$ *and* $\rho_{\widehat{\mathcal{B}}}(v)$ *do not contain any item of* X'. *Similarly, if* $L'_E \neq \emptyset$, *then* $\widehat{\mathcal{B}'}$ *can be satisfied only if* $\rho_{\widehat{\mathcal{B}'}}(u)$ *and* $\rho_{\widehat{\mathcal{B}'}}(v)$ *do not contain any item of* X.

Proof. The set X contains the edges of H and the labels appearing only next to the edges of H. Similarly X' consists of the edges and labels unique to H'. The presence of edges of H' in $\rho_{\widehat{\mathcal{B}}}(u)$ and $\rho_{\widehat{\mathcal{B}}}(v)$ is already forbidden by the definition of the division. If there is a label $x \in (L'_E \setminus L_E)$ present in $\rho_{\widehat{\mathcal{B}}}(u)$, then this label must be separated from the labels of L_E in every labeled embedding satisfying $\widehat{\mathcal{B}}$. But to separate the label x, there must be at least two edges e_1, e_2 with $l_{\widehat{\mathcal{B}}}(e_1) = x$ or $r_{\widehat{\mathcal{B}}}(e_1) = x$ and $l_{\widehat{\mathcal{B}}}(e_2) = x$ or $r_{\widehat{\mathcal{B}}}(e_2) = x$. This is a contradiction with the definitions of L_E and L'_E. □

Together Lemmata 73 and 74 yield Corollary 75.

Corollary 75. *G has a connected labeled embedding satisfying $\widehat{\mathcal{A}}$, only if $\rho_{\widehat{\mathcal{A}}}(u)$ and $\rho_{\widehat{\mathcal{A}}}(v)$ are (X, X')-non-crossing.*

Further in the analysis, we assume that both $\rho_{\widehat{\mathcal{A}}}(u)$ and $\rho_{\widehat{\mathcal{A}}}(v)$ are (X, X')-non-crossing. This condition can be checked in linear time with respect to the size of G.

Corollary 76. *For $w \in \{u, v\}$ if the rotation scheme $\rho_{\widehat{\mathcal{A}}}(w)$ contains an element of X and also an element of X', then there exist unique sequences $\tau, \tau, \lambda, \lambda'$ such that*

(i) $\rho_{\widehat{\mathcal{A}}}(w) = (\tau, \lambda, \tau', \lambda')$,

(ii) τ is a non-empty sequence of elements of the set $(E(H) \cup L_E)$ that does not start and does not end with a label of $(L_E \cap L'_E)$,

(iii) τ' is a non-empty sequence of elements of the set $(E(H') \cup L'_E)$ that does not start and does not end with a label of $(L_E \cap L'_E)$,

(iv) λ and λ' are either empty or they are sequences of labels of $(L_E \cap L'_E)$.

We show that there is always only a constant number of divisions of $\widehat{\mathcal{A}}$ which must be considered by the algorithm.

4.3.2.1 One label in the intersection

The case $L_E \cap L'_E = \{\ell\}$ is the simplest. In this situation there exists one universal ℓ-division which represents all the satisfiable ℓ-division of $\widehat{\mathcal{A}}$ with respect to (H, H').

Definition 77. *Let $L_E \cap L'_E = \{\ell\}$ and let $X = E(H) \cup (L_E \setminus L'_E)$ and $X' = E(H') \cup (L'_E \setminus L_E)$. The* universal *$\ell$-division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ of $\widehat{\mathcal{A}}$ with respect to (H, H') is an ℓ-division such that for each $w \in \{u, v\}$ the rotation schemes $\rho_{\widehat{\mathcal{B}}}(w)$ and $\rho_{\widehat{\mathcal{B}}'}(w)$ are defined as follows:*

(i) If $\rho_{\widehat{\mathcal{A}}}(w)$ does not contain any items of X and X', then both $\rho_{\widehat{\mathcal{B}}}(w)$ and $\rho_{\widehat{\mathcal{B}}'}(w)$ are empty.

(ii) If $\rho_{\widehat{\mathcal{A}}}(w)$ contains an item of X and there is no item of X', then $\rho_{\widehat{\mathcal{B}}}(w) = \rho_{\widehat{\mathcal{A}}}(w)$ and $\rho_{\widehat{\mathcal{B}}'}(w)$ is empty.

(iii) If $\rho_{\widehat{\mathcal{A}}}(w)$ contains an item of X' and there is no item of X, then $\rho_{\widehat{\mathcal{B}}}(w)$ is empty and $\rho_{\widehat{\mathcal{B}}'}(w) = \rho_{\widehat{\mathcal{A}}}(w)$.

(iv) *Otherwise* $\rho_{\widehat{\mathcal{A}}}(w)$ *contains an item of* X *and also an item of* X'. $\rho_{\widehat{\mathcal{A}}}(w)$ *can be uniquely decomposed as the cyclic sequence* $(\tau, \lambda, \tau', \lambda')$ *according to Corollary 76. Then* $\rho_{\widehat{\mathcal{B}}}(w) = (\tau, e_G^{virt}(H))$ *and* $\rho_{\widehat{\mathcal{B}}'}(w) = (\tau', e_G^{virt}(H'))$.

Remark. Note that the universal ℓ-division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ may not form a proper division according to Definition 70, because there might be some missing occurrences of ℓ in the rotation schemes of vertices u and v. However, the omitted instances of ℓ are not necessary. We can construct an equivalent proper division from $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ by applying Lemma 27.

Remark. We abuse this definition even in cases with $L_E = \emptyset$ or $L'_E = \emptyset$. When constructing the universal ℓ-division for $L_E = \emptyset$, we proceed as if $L_E = \{\ell\}$.

Lemma 78. *Let* $L_E \cap L'_E = \{\ell\}$ *and let* $(\widehat{\mathcal{B}}_*, \widehat{\mathcal{B}}'_*)$ *be the universal* ℓ*-division of* $\widehat{\mathcal{A}}$ *with respect to* (H, H'). *If* $\widehat{\mathcal{A}}$ *has an* ℓ*-division* $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ *such that both* H *and* H' *have connected labeled embeddings* $\mathcal{H}_L$, $\mathcal{H}'_L$ *satisfying* $\widehat{\mathcal{B}}$, $\widehat{\mathcal{B}}'$, *then* $\mathcal{H}_L$, $\mathcal{H}'_L$ *also satisfy* $\widehat{\mathcal{B}}_*$, $\widehat{\mathcal{B}}'_*$.

Proof. We need to check that $\mathcal{H}_L$, $\mathcal{H}'_L$ satisfy $\rho_{\widehat{\mathcal{B}}_*}(w)$, $\rho_{\widehat{\mathcal{B}}'_*}(w)$ for $w \in \{u, v\}$. We distinguish four cases based on $\rho_{\widehat{\mathcal{A}}}(w)$.

(i) $\rho_{\widehat{\mathcal{A}}}(w)$ does not contain any items of X and X'. Then both $\rho_{\widehat{\mathcal{B}}_*}(w)$ and $\rho_{\widehat{\mathcal{B}}'_*}(w)$ are empty, so they are always satisfied.

(ii) $\rho_{\widehat{\mathcal{A}}}(w)$ contains an item of X, but there is no item of X'. In this case $\rho_{\widehat{\mathcal{B}}_*}(w) = \rho_{\widehat{\mathcal{A}}}(w)$ and $\rho_{\widehat{\mathcal{B}}'_*}(w)$ is empty.

   $\rho_{\widehat{\mathcal{B}}'}(w)$ can cover at most one occurrence of ℓ from $\rho_{\widehat{\mathcal{A}}}(w)$. And even then the there would be $e_G^{virt}(H)$ in $\rho_{\widehat{\mathcal{B}}}(w)$ replacing the covered occurrence of ℓ, which would enforce the presence of ℓ anyway.

(iii) $\rho_{\widehat{\mathcal{A}}}(w)$ contains an item of X', but there is no item of X. This is symmetric to the case (ii).

(iv) $\rho_{\widehat{\mathcal{A}}}(w)$ contains an item of X and also an item of X'. Then $\rho_{\widehat{\mathcal{B}}_*}(w) = (\tau, e_G^{virt}(H))$ and $\rho_{\widehat{\mathcal{B}}'_*}(w) = (\tau', e_G^{virt}(H'))$.

   Lemma 74 implies that τ is contained in $\rho_{\widehat{\mathcal{B}}}(w)$ and τ' in $\rho_{\widehat{\mathcal{B}}'}(w)$. Therefore, the rotation schemes $\rho_{\widehat{\mathcal{B}}}(w)$, $\rho_{\widehat{\mathcal{B}}'}(w)$ can differ from $\rho_{\widehat{\mathcal{B}}_*}(w)$, $\rho_{\widehat{\mathcal{B}}'_*}(w)$ only by several appearances of ℓ next to the edges $e_G^{virt}(H)$, $e_G^{virt}(H')$. But these occurrences of ℓ are not necessary (Lemma 27).

□

4.3.2.2 Two labels in the intersection

The situation $L_E \cap L'_E = \{\ell_1, \ell_2\}$ is a bit more complicated. First, we have to decide whether we look for an (ℓ_1, ℓ_2)-division or an (ℓ_2, ℓ_1)-division. This can be done in linear time with respect to the size of H using Lemmata 25 and 63.

Next, we check that neither $\rho_{\widehat{\mathcal{A}}}(u)$ nor $\rho_{\widehat{\mathcal{A}}}(v)$ are (ℓ_1, ℓ_2)-crossing. Otherwise, based on Lemma 26 there is no connected labeled embedding satisfying $\widehat{\mathcal{A}}$.

After that, there can still be two nonequivalent options on how to split the rotation scheme $\rho_{\widehat{\mathcal{A}}}(u)$ so that it is not in a contradiction with Lemma 74. Thus, in this case, we have a set of universal divisions containing at most 4 elements.

Definition 79. *Let $L_E \cap L'_E = \{\ell_1, \ell_2\}$ and let $X = E(H) \cup (L_E \setminus L'_E)$ and $X' = E(H') \cup (L'_E \setminus L_E)$. For $w \in \{u, v\}$ we define two pairs of rotation schemes $(\rho_1(w), \rho'_1(w))$, $(\rho_2(w), \rho'_2(w))$ as follows:*

(i) If $\rho_{\widehat{\mathcal{A}}}(w)$ does not contain any items of X and X', then both $\rho_1(w)$ and $\rho'_1(w)$ are empty.

(ii) If $\rho_{\widehat{\mathcal{A}}}(w)$ contains an item of X and there is no item of X', then $\rho_1(w) = \rho_{\widehat{\mathcal{A}}}(w)$ and $\rho'_1(w)$ is empty.

(iii) If $\rho_{\widehat{\mathcal{A}}}(w)$ contains an item of X' and there is no item of X, then $\rho_1(w)$ is empty and $\rho'_1(w) = \rho_{\widehat{\mathcal{A}}}(w)$.

(iv) Otherwise $\rho_{\widehat{\mathcal{A}}}(w)$ contains an item of X and also an item of X'. $\rho_{\widehat{\mathcal{A}}}(w)$ can be uniquely decomposed as the cyclic sequence $(\tau, \lambda, \tau', \lambda')$ according to Corollary 76. We further distinguish several subcases based on the values of λ and λ'. In the following enumeration we assume that $w = u$. For $w = v$ it is necessary to swap the labels ℓ_1 and ℓ_2.

(a) If $\rho_{\widehat{\mathcal{A}}}(w) \in \{(\tau, \tau'), (\ell_1, \tau, \tau'), (\tau, \ell_2, \tau'), (\ell_1, \tau, \ell_2, \tau')\}$, then $\rho_1(w) = (\tau, e_G^{virt}(H))$ and $\rho'_1(w) = (e_G^{virt}(H'), \tau')$.

(b) If $\rho_{\widehat{\mathcal{A}}}(w) \in \{(\ell_1, \ell_2, \tau, \tau'), (\ell_1, \ell_2, \tau, \ell_2, \tau')\}$, then $\rho_1(w) = (\ell_2, \tau, e_G^{virt}(H))$ and $\rho'_1(w) = (e_G^{virt}(H'), \tau')$.

(c) If $\rho_{\widehat{\mathcal{A}}}(w) \in \{(\ell_2, \ell_1, \tau, \tau'), (\ell_2, \ell_1, \tau, \ell_2, \tau')\}$, then $\rho_1(w) = (\tau, e_G^{virt}(H))$ and $\rho'_1(w) = (\ell_2, e_G^{virt}(H'), \tau')$.

(d) If $\rho_{\widehat{\mathcal{A}}}(w) \in \{(\tau, \ell_2, \ell_1, \tau'), (\ell_1, \tau, \ell_2, \ell_1, \tau')\}$, then $\rho_1(w) = (\tau, e_G^{virt}(H))$ and $\rho'_1(w) = (e_G^{virt}(H'), \ell_1, \tau')$.

(e) If $\rho_{\widehat{\mathcal{A}}}(w) \in \{(\tau, \ell_1, \ell_2, \tau'), (\ell_1, \tau, \ell_1, \ell_2, \tau')\}$, then $\rho_1(w) = (\tau, \ell_1, e_G^{virt}(H))$ and $\rho'_1(w) = (e_G^{virt}(H'), \tau')$.

(f) If $\rho_{\widehat{\mathcal{A}}}(w) \in \{(\ell_1, \ell_2, \tau, \ell_1, \tau'), (\ell_2, \tau, \ell_2, \ell_1, \tau'), (\ell_1, \ell_2, \tau, \ell_2, \ell_1, \tau')\}$, then $\rho_1(w) = (\ell_2, \tau, e_G^{virt}(H))$ and $\rho'_1(w) = (e_G^{virt}(H'), \ell_1, \tau')$.

(g) If $\rho_{\widehat{\mathcal{A}}}(w) \in \{(\ell_2, \ell_1, \tau, \ell_1, \tau'), (\ell_2, \tau, \ell_1, \ell_2, \tau'), (\ell_2, \ell_1, \tau, \ell_1, \ell_2, \tau')\}$, then $\rho_1(w) = (\tau, \ell_1, e_G^{virt}(H))$ and $\rho'_1(w) = (\ell_2, e_G^{virt}(H'), \tau')$.

(h) If $\rho_{\widehat{\mathcal{A}}}(w) \in \{(\ell_2, \ell_1, \ell_2, \tau, \tau'), (\ell_2, \ell_1, \ell_2, \tau, \ell_2, \tau')\}$, then $\rho_1(w) = (\ell_2, \tau, e_G^{virt}(H))$ and $\rho'_1(w) = (\ell_2, e_G^{virt}(H'), \tau')$.

(i) If $\rho_{\widehat{\mathcal{A}}}(w) \in \{(\tau, \ell_1, \ell_2, \ell_1, \tau'), (\ell_1, \tau, \ell_1, \ell_2, \ell_1, \tau')\}$, then $\rho_1(w) = (\tau, \ell_1, e_G^{virt}(H))$ and $\rho'_1(w) = (e_G^{virt}(H'), \ell_1, \tau')$.

(j) If $\rho_{\widehat{\mathcal{A}}}(w) \in \{(\tau, \ell_1, \tau'), (\ell_1, \tau, \ell_1, \tau')\}$, then $\rho_1(w) = (\tau, \ell_1, e_G^{virt}(H))$, $\rho'_1(w) = (e_G^{virt}(H'), \tau')$ and $\rho_2(w) = (\tau, e_G^{virt}(H))$, $\rho'_2(w) = (e_G^{virt}(H'), \ell_1, \tau')$.

(k) If $\rho_{\widehat{\mathcal{A}}}(w) \in \{(\ell_2, \tau, \tau'), (\ell_2, \tau, \ell_2, \tau')\}$, then $\rho_1(w) = (\ell_2, \tau, e_G^{virt}(H))$, $\rho'_1(w) = (e_G^{virt}(H'), \tau')$ and $\rho_2(w) = (\tau, e_G^{virt}(H))$, $\rho'_2(w) = (\ell_2, e_G^{virt}(H'), \tau')$.

(l) If $\rho_{\widehat{\mathcal{A}}}(w) = (\ell_2, \tau, \ell_1, \tau')$, then $\rho_1(w) = (\ell_2, \tau, e_G^{virt}(H))$, $\rho'_1(w) = (e_G^{virt}(H'), \ell_1, \tau')$ and $\rho_2(w) = (\tau, \ell_1, e_G^{virt}(H))$, $\rho'_2(w) = (\ell_2, e_G^{virt}(H'), \tau')$.

If $(\rho_2(w), \rho'_2(w))$ *is not mentioned, then* $(\rho_2(w), \rho'_2(w)) = (\rho_1(w), \rho'_1(w))$.

Let D *be the set of* (ℓ_1, ℓ_2)*-divisions of* A *with respect to* (H, H') *such that for* $w \in \{u, v\}$ $(\rho_{\widehat{\mathcal{B}}}(w), \rho_{\widehat{\mathcal{B}}'}(w)) \in \{(\rho_1(w), \rho'_1(w)), (\rho_2(w), \rho'_2(w))\}$. *D is called* the set of universal (ℓ_1, ℓ_2)*-divisions of* $\widehat{\mathcal{A}}$ *with respect to* (H, H').

Remark. Note that the universal division may not form proper divisions according to Definition 70, because there might be some missing occurrences of ℓ_1 and ℓ_2 in the rotation schemes of vertices u and v. However, the omitted instances of ℓ_1, ℓ_2 were not necessary. We can construct an equivalent proper division from each universal division by applying Lemma 27.

Lemma 80. *Let* $L_E \cap L'_E = \{\ell_1, \ell_2\}$ *and let* D *be the set of universal* (ℓ_1, ℓ_2)*-divisions of* $\widehat{\mathcal{A}}$ *with respect to* (H, H')*. If* $\widehat{\mathcal{A}}$ *has a* (ℓ_1, ℓ_2)*-division* $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ *of* $\widehat{\mathcal{A}}$ *such that both* H *and* H' *have connected labeled embeddings* $\mathcal{H}_L$*,* $\mathcal{H}'_L$ *satisfying* $\widehat{\mathcal{B}}$*,* $\widehat{\mathcal{B}}'$*, then there is* $(\widehat{\mathcal{B}}_*, \widehat{\mathcal{B}}'_*) \in D$ *such that* $\mathcal{H}_L$*,* $\mathcal{H}'_L$ *also satisfy* $\widehat{\mathcal{B}}_*$*,* $\widehat{\mathcal{B}}'_*$*.*

Proof. We need to check that $\mathcal{H}_L$ satisfies $\rho_1(w)$ or $\rho_2(w)$ and that $\mathcal{H}'_L$ satisfies $\rho'_1(w)$ or $\rho'_2(w)$ for $w \in \{u, v\}$. We distinguish several cases based on $\rho_{\widehat{\mathcal{A}}}(w)$.

(i) $\rho_{\widehat{\mathcal{A}}}(w)$ does not contain any items of X and X'. Then both $\rho_1(w)$ and $\rho'_1(w)$ are empty, so they are always satisfied.

(ii) $\rho_{\widehat{\mathcal{A}}}(w)$ contains an item of X, but there is no item of X'. In this case we put $\rho_1(w) = \rho_{\widehat{\mathcal{A}}}(w)$ and $\rho'_1(w)$ is empty.

$\rho_{\widehat{\mathcal{B}}'}(w)$ can cover only a consecutive sequence of labels ℓ_1, ℓ_2 from $\rho_{\widehat{\mathcal{A}}}(w)$. More precisely, it can cover one of the sequences (ℓ_1), (ℓ_2), (ℓ_2, ℓ_1). The sequence (ℓ_1, ℓ_2) together with the edge $e_G^{virt}(H')$ would create the cyclic sequence $(\ell_1, \ell_2, \ell_1, \ell_2)$ which is forbidden by Lemma 26.

If $\rho_{\widehat{\mathcal{B}}'}(w)$ is not empty, then there must be $e_G^{virt}(H)$ in $\rho_{\widehat{\mathcal{B}}}(w)$ replacing the continuous subsequence covered by $\rho_{\widehat{\mathcal{B}}'}(w)$. But, $e_G^{virt}(H)$ enforces the presence of ℓ_2 and ℓ_1 next to itself. Therefore, $\rho_1(w)$ is still a subsequence of $\rho_{\mathcal{H}_L}(u)$.

(iii) $\rho_{\widehat{\mathcal{A}}}(w)$ contains an item of X', but there is no item of X. This is symmetric to case (ii).

(iv) $\rho_{\widehat{\mathcal{A}}}(w)$ contains an item of X and also an item of X'. Then we can examine all the feasible divisions of $\rho_{\widehat{\mathcal{A}}}(w)$ that do not contain a subsequence $(\ell_1, \ell_2, \ell_1, \ell_2)$ even if we include the labels incident to $e_G^{virt}(H)$ and $e_G^{virt}(H')$. We find out that every such division is equivalent (in the sense of Lemma 25) to $(\rho_1(w), \rho'_1(w))$ or $(\rho_2(w), \rho'_2(w))$.

□

4.3.2.3 No labels in the intersection

The last case is the situation when $L_E = \emptyset$ or $L'_E = \emptyset$. We show that in this setting there are at most two universal divisions covering all the satisfiable options.

Lemma 81. *If* $L_E = \emptyset$ *or* $L'_E = \emptyset$*, then there exists a set of labels* $L_P \subseteq (L_E \cup L'_E \cup \{\bigstar\})$ *such that* $|L_P| \leq 2$ *and* *(i) for each $p \in P$ there exists the universal p-division $(\widehat{\mathcal{B}}_p, \widehat{\mathcal{B}}'_p)$ of $\widehat{\mathcal{A}}$ w.r.t. (H, H'),*

(ii) if there exists a label $\ell \in (L_E \cup L'_E)$ such that H and H' have connected labeled embeddings $\mathcal{H}_L$ and $\mathcal{H}'_L$ satisfying the restrictions of an ℓ-division of $\widehat{\mathcal{A}}$ w.r.t. (H, H'), then there is $p \in L_P$ such that $\mathcal{H}_L$ satisfies $\widehat{\mathcal{B}}_p$ and $\mathcal{H}'_L$ satisfies $\widehat{\mathcal{B}}'_p$.

Proof. Without loss of generality, we suppose that $L_E = \emptyset$. Since $\widehat{\mathcal{A}}$ has labeled opaque edges, then all the edges of H except for $e_G^{virt}(H)$ are transparent. Therefore, each labeled embedding of H satisfying a division of $\widehat{\mathcal{A}}$ must have all the faces tagged by one common label. We investigate for which parental labels ℓ there exists an ℓ-division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ of $\widehat{\mathcal{A}}$ such that both H and H' have connected labeled embeddings satisfying $\widehat{\mathcal{B}}$ and $\widehat{\mathcal{B}}'$.

If there is a vertex $w \in (V(H) \setminus \{u, v\})$ with a label x appearing in $\rho_{\widehat{\mathcal{A}}}(w)$ then x is the only possible choice for a satisfiable division. Other labels cannot succeed. So if there is a label present in the rotation schemes of vertices of $(V(H) \setminus \{u, v\})$, we get at most one candidate for the parental label.

Further, we look at the constraints induced by the rotation schemes $\rho_{\widehat{\mathcal{A}}}(u)$ and $\rho_{\widehat{\mathcal{A}}}(v)$. These rotation schemes can contain edges of $E(H)$, labels of L'_E and edges of $E(H')$. Let us consider all the possible compositions of $\rho_{\widehat{\mathcal{A}}}(u)$. The situation for vertex v is identical. We show that either $\rho_{\widehat{\mathcal{A}}}(u)$ can be partitioned to two intervals ρ and ρ', where ρ contains the edges of $E(H)$ and ρ' the edges of $E(H')$ plus the labels of L'_E, or $\rho_{\widehat{\mathcal{A}}}(u)$ limits us to at most two candidates for the parental label.

(i) If $\rho_{\widehat{\mathcal{A}}}(u)$ does not contain any edges of $E(H)$, then we get the described partition by leaving ρ empty and putting $\rho' = \rho_{\widehat{\mathcal{A}}}(u)$.

(ii) If $\rho_{\widehat{\mathcal{A}}}(u)$ contains an edge of $E(H)$ and also an edge of $E(H')$, then $\rho_{\widehat{\mathcal{A}}}(u)$ can be partitioned to two intervals τ, τ' such that τ contains all the edges of $E(H)$, it starts and ends with an edge of $E(H)$ and it does not contain any edge of $E(H')$. In every division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ of $\widehat{\mathcal{A}}$ with respect to (H, H') the interval τ is included in $\rho_{\widehat{\mathcal{B}}}(u)$. Thus, if there is a label x in τ, then x is the only possibility for the parental label.

Otherwise, we get the described partition by putting $\rho = \tau$ and $\rho' = \tau'$.

(iii) If $\rho_{\widehat{\mathcal{A}}}(u)$ does not contain any edges of $E(H')$ and any labels of L'_E, then we get the described partition by putting $\rho = \rho_{\widehat{\mathcal{A}}}(u)$ and leaving ρ' empty.

(iv) Otherwise $\rho_{\widehat{\mathcal{A}}}(u)$ contains an edge of $E(H)$ and a label of L'_E, but it does not contain any edge of $E(H')$. Therefore, there is $t \geq 1$ such that $\rho_{\widehat{\mathcal{A}}}(u) = (\varepsilon_1, \lambda_1, \varepsilon_2, \lambda_2, \ldots, \varepsilon_t, \lambda_t)$ where $\varepsilon_1, \ldots, \varepsilon_t$ are intervals formed by the edges of $E(H)$ and $\lambda_1, \ldots, \lambda_t$ are intervals of labels of L'_E. In every division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ of $\widehat{\mathcal{A}}$ at least $(t-1)$ of the t intervals $\lambda_1, \ldots, \lambda_t$ goes to $\rho_{\widehat{\mathcal{B}}}(u)$. So in order to satisfy $\widehat{\mathcal{B}}$, there must be a label x such that at least $(t-1)$ of the $\lambda_1, \ldots, \lambda_t$ are just the 1-element interval (x). We further distinguish several options:

   (a) If $t = 1$, then independently on the parental label we get the described partition by putting $\rho = \varepsilon_1$ and $\rho' = \lambda_1$.

(b) If $t \geq 3$ and there is a label x such that at least $(t-1)$ intervals of $\lambda_1, \ldots, \lambda_t$ are just 1-element interval (x), then x is the only possible choice for the parental label.

(c) If $t = 2$ and there are two different labels x, y such that $\lambda_1 = (x)$ and $\lambda_2 = (y)$, then the parental label must be x or y

(d) If $t = 2$ and there is a label x such that either $\lambda_1 = \lambda_2 = (x)$, or one of λ_1 and λ_2 is (x) and the other contains at least two labels, then x is the only possibility for the parental label.

Analyzing the candidates for the parental label ℓ we get into one of the four following states:

(1) We eliminated all of the labels of L'_E. It means that there is no satisfiable division of $\widehat{\mathcal{A}}$, so we can leave $L_P = \emptyset$.

(2) There is just one candidate x remaining. We ruled out all the other labels of L'_E. In this case, we put $L_P = \{x\}$. If $\mathcal{H}_L$ and $\mathcal{H}'_L$ satisfy an x-division of $\widehat{\mathcal{A}}$, then they must also satisfy the universal x-division of $\widehat{\mathcal{A}}$.

(3) We are left with two candidates x and y. The other labels of L'_E were excluded. Then, we put $L_P = \{x, y\}$.

(4) We did not exclude any candidates for the parental label. However, we have shown that both $\rho_{\widehat{\mathcal{A}}}(u)$ and $\rho_{\widehat{\mathcal{A}}}(v)$ can be partitioned to an interval ρ containing just the edges of $E(H)$ and interval ρ' containing the edges of $E(H')$ plus the labels of L'_E. It means that independently on the parental label ℓ the universal ℓ-division $(\widehat{\mathcal{B}}_\ell, \widehat{\mathcal{B}}'_\ell)$ of $\widehat{\mathcal{A}}$ w.r.t. (H, H') does not have labels in the rotation schemes of $\widehat{\mathcal{B}}_\ell$. The only label appearing in $\widehat{\mathcal{B}}_\ell$ is the parental label at the edge $e_G^{virt}(H)$.

    Let $(\widehat{\mathcal{B}}_\bigstar, \widehat{\mathcal{B}}'_\bigstar)$ be the universal $\bigstar$-division of $\widehat{\mathcal{A}}$ with respect to (H, H'). Therefore, if there exist a label ℓ and an ℓ-division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ of $\widehat{\mathcal{A}}$ such that both H and H' have connected labeled embeddings $\mathcal{H}_L$, $\mathcal{H}'_L$ satisfying $\widehat{\mathcal{B}}$ and $\widehat{\mathcal{B}}'$, then $\mathcal{H}_L$ and $\mathcal{H}'_L$ satisfy $\widehat{\mathcal{B}}_\bigstar$ and $\widehat{\mathcal{B}}'_\bigstar$. Hence, we can put $L_P = \{\bigstar\}$.

□

We can find the set L_P in linear time with respect to the size of H. In the case (4) we get a $\bigstar$-division that is not covered by Lemma 73. However, it does not matter. If H and H' have connected labeled embeddings $\mathcal{H}_\bigstar$, $\mathcal{H}'_\bigstar$ satisfying $\widehat{\mathcal{B}}_\bigstar$, $\widehat{\mathcal{B}}'_\bigstar$, then we can put $p = l_{\mathcal{H}'_\bigstar}(e_G^{virt}(H'))$ and relabel $\mathcal{H}_\bigstar$ by p, getting a pair of connected labeled embeddings that satisfies the universal p-division of $\widehat{\mathcal{A}}$ w.r.t. (H, H').

Also notice that if $L_E = \emptyset$, then the edge $e_G^{virt}(H)$ in the universal $\bigstar$-division might be opaque with no labels for its incident edges. But in this case there is just one opaque edge in H, so we can make it transparent and preserve the labeled opaque edges property.

Algorithm 6: A faster exponential algorithm for AERCS of biconnected graphs.

input : A biconnected graph G, an augmented embedding restriction $\widehat{\mathcal{A}}$ of G with labeled opaque edges.

function `AERCS_exp_II`$(G, \widehat{\mathcal{A}})$**:**
- **if** $\left|\{\textit{the labels prescribed by } l_{\widehat{\mathcal{A}}}, r_{\widehat{\mathcal{A}}}\}\right| \leq 1$ **:**
  - **if** $\left|\{\textit{the labels prescribed by } l_{\widehat{\mathcal{A}}}, r_{\widehat{\mathcal{A}}}, \rho_{\widehat{\mathcal{A}}}\}\right| \geq 2$ **: return** false;
  - **else return** `ERCS_biconnected`$(G,$ `omit_labels`$(\widehat{\mathcal{A}}))$;
- **if** *G is an [SPQR]-skeleton* **: return** `AERCS_skeleton`$(G, \widehat{\mathcal{A}})$;
- $\{u, v\} \leftarrow$ a separation pair of G;
- $(H, H') \leftarrow$ a simple pair of split graphs w.r.t. $\{u, v\}$;
- **if** *(H, H') is not $\widehat{\mathcal{A}}$-non-crossing* **: return** false;
- $L_E \leftarrow \{l_{\widehat{\mathcal{A}}}(e), r_{\widehat{\mathcal{A}}}(e) \mid e \in E(H) \cap E(G)\} \setminus \{\bigstar\}$;
- $L'_E \leftarrow \{l_{\widehat{\mathcal{A}}}(e), r_{\widehat{\mathcal{A}}}(e) \mid e \in E(H') \cap E(G)\} \setminus \{\bigstar\}$;
- $X \leftarrow E(H) \cup L_E \setminus L'_E$; $X' \leftarrow E(H') \cup L'_E \setminus L_E$;
- **if** $|L_E \cap L'_E| \geq 3$ **: return** false;
- **if** $|L_E| > 0$ **and** $|L'_E| > 0$ **and** $|L_E \cap L'_E| = 0$ **: return** false;
- **if** $|L \cap L'| = \{\ell\}$ **:**
  - **if** *$\rho_{\widehat{\mathcal{A}}}(u)$ or $\rho_{\widehat{\mathcal{A}}}(v)$ is not (X, X')-non-crossing* **: return** false;
  - **if** *both H and H' are not ℓ-passable* **: return** false;
  - $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}') \leftarrow$ the universal ℓ-division of $\widehat{\mathcal{A}}$ w.r.t. (H, H');
  - **if** `AERCS_exp_II`$(H, \widehat{\mathcal{B}})$ **and** `AERCS_exp_II`$(H', \widehat{\mathcal{B}}')$ **: return** true ;
- **else if** $|L_E| = 0$ **or** $|L'_E| = 0$ **:**
  - $L_P \leftarrow$ the set of labels from Lemma 81;
  - **foreach** *label $p \in L_P$* **:**
    - $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}') \leftarrow$ the universal p-division of $\widehat{\mathcal{A}}$ w.r.t. (H, H');
    - **if** `AERCS_exp_II`$(H, \widehat{\mathcal{B}})$ **and** `AERCS_exp_II`$(H', \widehat{\mathcal{B}}')$ **: return** true ;
- **else**
  - $\{\ell_1, \ell_2\} \leftarrow L_E \cap L'_E$;
  - **if** *$\rho_{\widehat{\mathcal{A}}}(u)$ or $\rho_{\widehat{\mathcal{A}}}(v)$ is not (X, X')-non-crossing* **: return** false;
  - $\widehat{\mathcal{R}} \leftarrow$ `omit_labels_in_rotation_schemes`$(\widehat{\mathcal{A}})$;
  - **if** `local_Euler_test`$(H, \widehat{\mathcal{R}}[H, \ell_1, \ell_2], u, \ell_1)$ **:** $(x, y) \leftarrow (\ell_1, \ell_2)$;
  - **else** $(x, y) \leftarrow (\ell_2, \ell_1)$;
  - **foreach** *universal (x, y)-divisions $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ of $\widehat{\mathcal{A}}$ w.r.t. (H, H')* **:**
    - **if** `AERCS_exp_II`$(H, \widehat{\mathcal{B}})$ **and** `AERCS_exp_II`$(H', \widehat{\mathcal{B}}')$ **:**
      - **return** true;
- **return** false;

4.3.2.4 A faster exponential algorithm

We can design a faster exponential-time algorithm utilizing Lemmata 78, 80 and 81. Algorithm 6 tries at most 4 divisions at every recursion level, so its time complexity is $|E(G)| \cdot 4^{\mathcal{O}(|V(G)|)}$.

4.3.3 Merging restrictions

We further improve Algorithm 6, so that it runs in polynomial time. The revision is based on two ideas.

The first one is that we always select the separation pair $\{u, v\}$ and the simple pair of split graphs (H, H') in such a way that H is an [SPR]-skeleton. This allows us to solve quickly the subproblem for H. Therefore, for each universal division we only have to make the recursive call for H'.

The second notion is more complicated. We show that in cases with multiple universal divisions it is possible to process all the subproblems for H' simultaneously. Hence, the function solving AERCS$(G, \widehat{\mathcal{A}})$ does not have to make more than one recursive call.

Let us look at the cases that produce more than one universal division.

4.3.3.1 Bicolored edges

We start with the situation $L_E = \emptyset$. The algorithm generates two universal divisions $(\widehat{\mathcal{B}}_x, \widehat{\mathcal{B}}'_x)$ and $(\widehat{\mathcal{B}}_y, \widehat{\mathcal{B}}'_y)$ of $\widehat{\mathcal{A}}$ with respect to (H, H'), only if there exists $w \in \{u, v\}$ such that $\rho_{\widehat{\mathcal{A}}}(w) = (\varepsilon_1, x, \varepsilon_2, y)$, where ε_1, ε_2 are sequences of edges of $E(H)$ and $x, y \in L'_E$. The rotation scheme of the second vertex w' of $\{u, v\}$ must be either $(\varepsilon_3, x, \varepsilon_4, y)$ or (ε_5, τ'), where ε_3, ε_4, ε_5 are sequences of edges of $E(H)$, ε_5 might be empty, and τ' is a sequence of edges of $E(H')$ and labels of L'_E. The universal divisions satisfy the following conditions.

- $l_{\widehat{\mathcal{B}}_x}(e_G^{virt}(H)) = r_{\widehat{\mathcal{B}}_x}(e_G^{virt}(H)) = l_{\widehat{\mathcal{B}}'_x}(e_G^{virt}(H')) = r_{\widehat{\mathcal{B}}'_x}(e_G^{virt}(H')) = x$,
  $\rho_{\widehat{\mathcal{B}}_x}(w) = (\varepsilon_1, x, \varepsilon_2, e_G^{virt}(H))$, $\rho_{\widehat{\mathcal{B}}'_x}(w) = (e_G^{virt}(H'), y)$.
- $l_{\widehat{\mathcal{B}}_y}(e_G^{virt}(H)) = r_{\widehat{\mathcal{B}}_y}(e_G^{virt}(H)) = l_{\widehat{\mathcal{B}}'_y}(e_G^{virt}(H')) = r_{\widehat{\mathcal{B}}'_y}(e_G^{virt}(H')) = y$,
  $\rho_{\widehat{\mathcal{B}}_y}(w) = (\varepsilon_1, y, \varepsilon_2, e_G^{virt}(H))$, $\rho_{\widehat{\mathcal{B}}'_y}(w) = (e_G^{virt}(H'), x)$.

Notice that both $\rho_{\widehat{\mathcal{B}}'_x}(w)$ and $\rho_{\widehat{\mathcal{B}}'_y}(w)$ are equivalent to (x, y). Therefore, we can replace $\widehat{\mathcal{B}}'_x$ and $\widehat{\mathcal{B}}'_y$ by equivalent augmented embedding restrictions $\widehat{\mathcal{C}}'_x$ and $\widehat{\mathcal{C}}'_y$ that differ only in the labels of the edge $e_G^{virt}(H')$.

If $\rho_{\widehat{\mathcal{A}}}(w') = (\varepsilon_3, x, \varepsilon_4, y)$, then not only we have to replace $\rho_{\widehat{\mathcal{B}}'_x}(w)$ and $\rho_{\widehat{\mathcal{B}}'_y}(w)$ by the equivalent cyclic sequence (x, y), but we must also substitute for $\rho_{\widehat{\mathcal{B}}'_x}(w')$ and $\rho_{\widehat{\mathcal{B}}'_y}(w')$. Otherwise, $\rho_{\widehat{\mathcal{B}}'_x}(w')$ and $\rho_{\widehat{\mathcal{B}}'_y}(w')$ are already identical.

The little difference of $\widehat{\mathcal{C}}'_x$ and $\widehat{\mathcal{C}}'_y$ motivates us to merge these two restrictions into a single object.

Definition 82. *Let $\widehat{\mathcal{A}}$ be an augmented embedding restriction of a graph G. A* bicolored *edge of $\widehat{\mathcal{A}}$ is a triplet (e, x, y), where $e \in T_{\widehat{\mathcal{A}}}$ is an transparent edge and x, y are labels of $\widehat{\mathcal{A}}$.*

Further, let $Q_{\widehat{\mathcal{A}}}$ be a set of bicolored edges. An augmented embedding restriction $\widehat{\mathcal{E}}$ is an expansion *of $(\widehat{\mathcal{A}}, Q_{\widehat{\mathcal{A}}})$ if for each $(e, x, y) \in Q_{\widehat{\mathcal{A}}}$ there exists a label $\ell_e \in \{x, y\}$ such that $\widehat{\mathcal{E}}$ is the embedding restriction derived from $\widehat{\mathcal{A}}$ by setting $l_{\widehat{\mathcal{E}}}(e) = r_{\widehat{\mathcal{E}}}(e) = \ell_e$ for all the bicolored edges of $Q_{\widehat{\mathcal{A}}}$.*

A (connected) labeled embedding satisfies *$(\widehat{\mathcal{A}}, Q_{\widehat{\mathcal{A}}})$ if it satisfies an expansion of $(\widehat{\mathcal{A}}, Q_{\widehat{\mathcal{A}}})$.*

Corollary 83. *Let $\widehat{\mathcal{C}}'$ be the augmented embedding restriction obtained from $\widehat{\mathcal{C}}'_x$ by setting $l_{\widehat{\mathcal{C}}'}(e_G^{virt}(H')) = r_{\widehat{\mathcal{C}}'}(e_G^{virt}(H')) = \bigstar$ and let $Q_{\widehat{\mathcal{C}}'} = \{(e_G^{virt}(H'), x, y)\}$. Then a (connected) labeled embedding of H' satisfies $\widehat{\mathcal{B}}'_x$ or $\widehat{\mathcal{B}}'_y$ iff it satisfies $(\widehat{\mathcal{C}}', Q_{\widehat{\mathcal{C}}'})$.*

The intuition for the cases with $L_E = \emptyset$, which have two universal divisions, is to first solve AERCS($H, \widehat{\mathcal{B}}_x$) and AERCS($H, \widehat{\mathcal{B}}_y$). If both of them fails, then we reject. Else if only one of them succeeds, then we can focus on the one corresponding instance for H'. Else we construct the augmented embedding restriction $\widehat{\mathcal{C}}'$ from Corollary 83 and we continue solving AERCS($H', \widehat{\mathcal{C}}'$) with the bicolored edge $e_G^{virt}(H')$.

We have a different strategy for $L'_E = \emptyset$. If we receive two universal divisions $(\widehat{\mathcal{B}}_x, \widehat{\mathcal{B}}'_x)$ and $(\widehat{\mathcal{B}}_y, \widehat{\mathcal{B}}'_y)$ of $\widehat{\mathcal{A}}$ with respect to (H, H'), then each of $\widehat{\mathcal{B}}'_x$ and $\widehat{\mathcal{B}}'_y$ contains at most one unique label. Therefore, we can solve AERCS($H, \widehat{\mathcal{B}}'_x$) and AERCS($H, \widehat{\mathcal{B}}'_y$) in polynomial time as ERCS instances without labels. And AERCS($H, \widehat{\mathcal{B}}_x$) and AERCS($H, \widehat{\mathcal{B}}_y$) are already solvable in polynomial time because H is an [SPR]-skeleton. Hence, we do not have to make any recursive call in this case.

Let $b = (e, x, y)$ be a bicolored edge of $\widehat{\mathcal{A}}$. If $\widehat{\mathcal{A}}$ has labeled opaque edges and there is at least one label present in $\widehat{\mathcal{A}}$, then it might be possible to exclude some of the labels of b. If the label x does not appear in $\widehat{\mathcal{A}}$, then the expansion of b to x cannot yield a satisfiable augmented embedding restriction. Therefore, it makes sense to regularize the bicolored edges. If we manage to exclude both of the labels, then there is no satisfiable expansion. And if we eliminated just one of the labels, then we can automatically perform the expansion to the second label.

4.3.3.2 Flipping anchors and paths

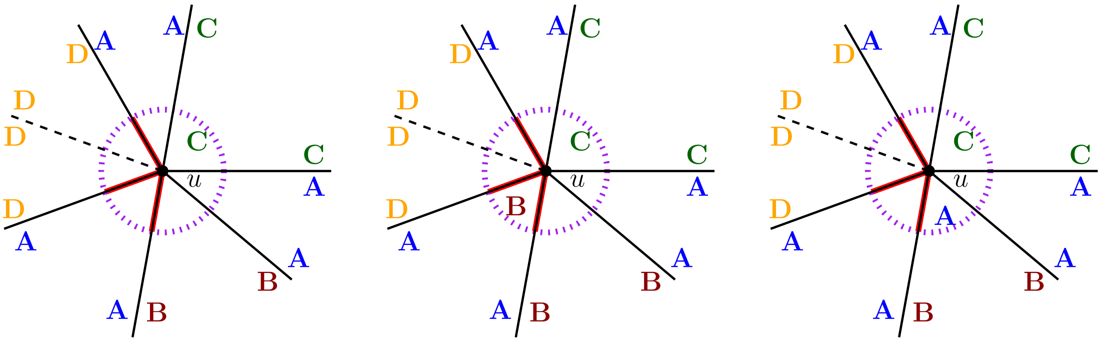
The last situation that needs to be investigated is $L_E \cap L'_E = \{\ell_1, \ell_2\}$. We observe that if the set of universal divisions has more than one element, then these universal divisions differ only in the labels appearing next to the edges $e_G^{virt}(H)$, $e_G^{virt}(H')$ in the rotation schemes $\rho_{\widehat{\mathcal{B}}}(u)$, $\rho_{\widehat{\mathcal{B}}'}(u)$ and $\rho_{\widehat{\mathcal{B}}}(v)$, $\rho_{\widehat{\mathcal{B}}'}(v)$. Again we would like to merge the relevant restrictions for H' into one single object.

Definition 84. *Let $\widehat{\mathcal{A}}$ be an augmented embedding restriction of a graph G and let $e \in (E(G) \setminus T_{\widehat{\mathcal{A}}})$, $e = (w, w')$, be an opaque edge of G such that $l_{\widehat{\mathcal{A}}}(e) = x$, $r_{\widehat{\mathcal{A}}}(e) = y$, $x \neq y$ and e is anchored at $\rho_{\widehat{\mathcal{A}}}(w)$. Next, let $R \subseteq \{(e), (x, e), (e, y)\}$, $|R| \geq 2$, be the* list of possible expansions*. The triplet (w, e, R) is called a* flipping anchor *of $\widehat{\mathcal{A}}$.*

Further, let $O_{\widehat{\mathcal{A}}}$ be a set of flipping anchors of $\widehat{\mathcal{A}}$. An augmented embedding restriction $\widehat{\mathcal{E}}$ is an expansion *of $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}})$ if for each $(w_e, e, R_e) \in O_{\widehat{\mathcal{A}}}$ there exists $\varphi_e \in R_e$ such that $\widehat{\mathcal{E}}$ is the restriction derived from $\widehat{\mathcal{A}}$ by replacing the edge e in $\rho_{\widehat{\mathcal{A}}}(w_e)$ by φ_e for all the anchors of $O_{\widehat{\mathcal{A}}}$. If the substitution produces a rotation scheme with two consecutive occurrences of the same label, then we merge these appearances of the same label into one.*

A (connected) labeled embedding satisfies *$(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}})$ if it satisfies an expansion of $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}})$.*

Flipping anchors can be used in situations when the universal divisions differ only in the rotation scheme of one of the separation vertices. However, the universal divisions may diverge in the rotation schemes of both vertices. These cases can be covered by pairing two flipping anchors together.



flipping anchor $(u, e, \{(e, A), (B, e)\})$ | expansion to (B, e) | expansion to (e, A)

Figure 4.4: A flipping anchor $(u, e, \{(e, A), (B, e)\})$ of an embedding restriction $\widehat{\mathcal{A}}$ and its two possible expansions. Note that the expansion on the right side can be satisfied, whereas the expansion on the left cannot.

Definition 85. *Let $\widehat{\mathcal{A}}$ be an augmented embedding restriction of a graph G and let $a = (w_a, e_a, R_a)$, $b = (w_b, e_b, R_b)$ be two flipping anchors. Next, let p be a trail from w_a to w_b containing only opaque edges of G such that*

(i) the first edge of p is e_a, the last edge is e_b,

(ii) for each edge e of p it holds that $l_{\widehat{\mathcal{A}}}(e) \neq r_{\widehat{\mathcal{A}}}(e)$,

(iii) for each pair of consecutive edges e, e' of p, $e = \{w, w'\}$, $e' = \{w', w''\}$, the edges e and e' are adjacent in $\rho_{\widehat{\mathcal{A}}}(w')$. Moreover, if we orient the edges e, e' from w' outwards and $\rho_{\widehat{\mathcal{A}}}(w') = (\varphi, e', e)$ for a possibly empty sequence φ, then $l_{\widehat{\mathcal{A}}}(e) = r_{\widehat{\mathcal{A}}}(e')$, and in the label ordering graph of w' in $\widehat{\mathcal{A}}$ there is no directed path from $r_{\widehat{\mathcal{A}}}(e)$ to $l_{\widehat{\mathcal{A}}}(e)$ avoiding $l_{\widehat{\mathcal{A}}}(e')$.

If φ is empty, then it is enough when the condition holds for one of the two possible directions. (I.e. if $l_{\widehat{\mathcal{A}}}(e) = r_{\widehat{\mathcal{A}}}(e')$, then it is not necessary that $r_{\widehat{\mathcal{A}}}(e) = l_{\widehat{\mathcal{A}}}(e')$.)

Next, let $R \subseteq (R_a \times R_b)$, $|R| \geq 2$. The quadruplet (a, b, p, R) is called a flipping path *of $\widehat{\mathcal{A}}$.*

Further, let $P_{\widehat{\mathcal{A}}}$ be a set of flipping paths of $\widehat{\mathcal{A}}$. An augmented embedding restriction $\widehat{\mathcal{E}}$ is an expansion *of $(\widehat{\mathcal{A}}, P_{\widehat{\mathcal{A}}})$ if for each $(a_p, b_p, p, R_p) \in P_{\widehat{\mathcal{A}}}$, $a_p = (w_{ap}, e_{ap}, R_{ap})$, $b_p = (w_{bp}, e_{bp}, R_{bp})$, there exists $(\varphi_p, \varphi'_p) \in R_p$ such that $\widehat{\mathcal{E}}$ is the augmented embedding restriction derived from $\widehat{\mathcal{A}}$ by replacing the edge e_{ap} in $\rho_{\widehat{\mathcal{A}}}(w_{ap})$ by φ_p and the edge e_{bp} in $\rho_{\widehat{\mathcal{A}}}(w_{bp})$ by φ'_p for all the flipping edges of $P_{\widehat{\mathcal{A}}}$. If the substitution produces a rotation scheme with two consecutive occurrences of the same label, then we merge these appearances of the same label into one.*

A (connected) labeled embedding satisfies *$(\widehat{\mathcal{A}}, P_{\widehat{\mathcal{A}}})$ if it satisfies an expansion of $(\widehat{\mathcal{A}}, P_{\widehat{\mathcal{A}}})$.*

A flipping path of $\widehat{\mathcal{A}}$ is based on a trail p, so it can visit one vertex multiple times. However, if this is the case, then there is no connected labeled embedding satisfying $\widehat{\mathcal{A}}$. (The trail p contains a cycle C such that for each edge e of C it holds that $l_{\widehat{\mathcal{A}}}(e) \neq r_{\widehat{\mathcal{A}}}(e)$. Furthermore, the conditions from the definition ensure

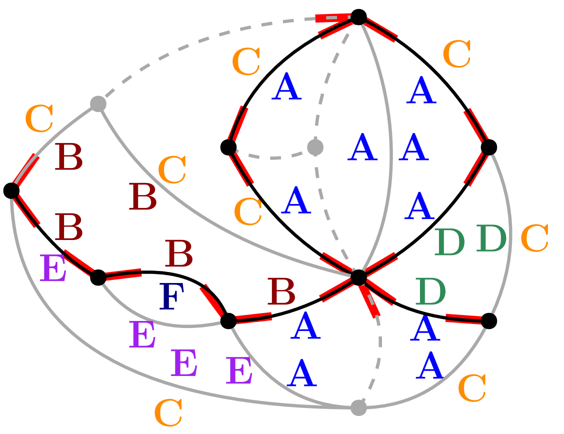


Figure 4.5: An embedding restriction with a flipping path (trail). The edges of the flipping path are highlighted in black.

that in each labeled embedding of G satisfying $\widehat{\mathcal{A}}$ has a label ℓ tagging a face located inside C and a face outside C.) Therefore, it make sense to talk just about flipping paths instead of flipping trails.

The point (iii) of Definition 85 seems very complicated. Especially, the part about no directed path from $r_{\widehat{\mathcal{A}}}(e)$ to $l_{\widehat{\mathcal{A}}}(e)$ avoiding $l_{\widehat{\mathcal{A}}}(e')$ in the label ordering graph of w' in $\widehat{\mathcal{A}}$ is quite perplexing. Basically, this condition says that in each connected labeled embedding $\mathcal{G}_L$ of G satisfying $\widehat{\mathcal{A}}$ there is no face incident to the vertex w' and tagged by the label $l_{\widehat{\mathcal{A}}}(e)$ between the edges e' and e in the counter-clockwise direction. If there was such a face, then $(e', l_{\widehat{\mathcal{A}}}(e'), l_{\widehat{\mathcal{A}}}(e), r_{\widehat{\mathcal{A}}}(e), e, l_{\widehat{\mathcal{A}}}(e))$ would be a subsequence of the augmented rotation scheme $\rho_{\mathcal{G}_L}(w')$. However, the condition implies that, there must be another occurrence of the label $l_{\widehat{\mathcal{A}}}(e')$ between $l_{\widehat{\mathcal{A}}}(e)$ and $r_{\widehat{\mathcal{A}}}(e)$. Therefore, $(l_{\widehat{\mathcal{A}}}(e'), l_{\widehat{\mathcal{A}}}(e), l_{\widehat{\mathcal{A}}}(e'), l_{\widehat{\mathcal{A}}}(e))$ would be a subsequence of $\rho_{\mathcal{G}_L}(w')$ and this is a contradiction with the connectivity of $\mathcal{G}_L$.

Notice that if there is a standalone flipping anchor (w, e, R) such that $\varphi \in R$, $\psi \in R$ and φ is a subsequence of ψ, then the expansion to ψ is redundant. If a labeled embedding $\mathcal{G}_L$ satisfies an expansion with (w, e, R) expanded to ψ, then $\mathcal{G}_L$ also satisfies the expansion to φ. Moreover, it means that if $(e) \in R$, then the flipping anchor (w, e, R) is useless.

Similarly for a flipping path (a, b, p, S) such that $(\varphi, \varphi') \in S$, $(\psi, \psi') \in S$, φ is a subsequence of ψ and φ' is a subsequence of ψ', the expansion to (ψ, ψ') is redundant. And if $((e_a), (e_b)) \in S$ then the flipping path is useless.

Further in the text, we assume that all the flipping anchors and flipping paths do not contain redundant expansions. Otherwise, we can regularize them by removing the unnecessary expansions. Moreover, if a flipping anchor (path) allows only one expansion after the regularization, then we perform this expansion and remove the anchor (path) from the set of flipping anchors (paths). And if a flipping path (a, b, p, R) admits only one expansion of the anchor a (or b) then we undertake the expansion and change the flipping path to a flipping anchor.

Also if a flipping anchor (w, e, R) of $\widehat{\mathcal{A}}$ allows the expansion of e to (x, e) for a label x and if there already is an occurrence of x immediately preceding e in $\rho_{\widehat{\mathcal{A}}}(w)$, then the label x in the expansion (x, e) is not necessary and we can remove it during the regularization.

4.3.3.3 A*ERCS problem

We define a new problem by adding sets of bicolored edges, flipping anchors and paths to an augmented embedding restriction.

Definition 86. *Let $\widehat{\mathcal{A}}$ be an augmented embedding restriction of a graph G, $O_{\widehat{\mathcal{A}}}$ a set of flipping anchors of $\widehat{\mathcal{A}}$, $P_{\widehat{\mathcal{A}}}$ a set of flipping paths of $\widehat{\mathcal{A}}$ and $Q_{\widehat{\mathcal{A}}}$ a set of bicolored edges of $\widehat{\mathcal{A}}$. An augmented embedding restriction $\widehat{\mathcal{E}}$ is an* expansion *of $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$ if there exist augmented embedding restriction $\widehat{\mathcal{E}}_1$, $\widehat{\mathcal{E}}_2$ such that $\widehat{\mathcal{E}}_1$ is an expansion of $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}})$, $\widehat{\mathcal{E}}_2$ is an expansion of $(\widehat{\mathcal{E}}_1, P_{\widehat{\mathcal{A}}})$ and $\widehat{\mathcal{E}}$ is an expansion of $(\widehat{\mathcal{E}}_2, Q_{\widehat{\mathcal{A}}})$. A (connected) labeled embedding* satisfies $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$ *if it satisfies an expansion of $(\widehat{\mathcal{A}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$.*

Problem 87 (A*ERCS)**.**
Input: *A biconnected graph G, an augmented embedding restriction $\widehat{\mathcal{A}}$ of G with labeled opaque edges, a set of flipping anchors $O_{\widehat{\mathcal{A}}}$, a set of edge-disjoint flipping paths $P_{\widehat{\mathcal{A}}}$ that is also disjoint with the edges of the flipping anchors of $O_{\widehat{\mathcal{A}}}$, a set of bicolored edges $Q_{\widehat{\mathcal{A}}}$.*
Question: *Does there exist a connected labeled embedding $\mathcal{G}_L$ of G satisfying $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$?*

4.3.4 A*ERCS of biconnected graphs

We derive a polynomial algorithm solving the A*ERCS instances with a limited number of bicolored edges. The algorithm employs the same divide-and-conquer notion as in the ERCS problem. Again, we start with the SPQR skeletons and later we describe how to divide the problem for a simple pair of split graphs.

Lemma 88. *Let G be an [SQR]-skeleton, $\widehat{\mathcal{A}}$ an augmented embedding restriction of G with labeled opaque edges, $O_{\widehat{\mathcal{A}}}$ a set of flipping anchors, $P_{\widehat{\mathcal{A}}}$ a set of edge disjoint flipping paths and $Q_{\widehat{\mathcal{A}}}$ a set of bicolored edges. Then the instance A*ERCS$(G, \widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$ can be solved in linear time.*

Proof. [SQR]-skeletons have only a constant number of planar embeddings and all of them can be found in linear time with respect to the size of G. So we can afford to try all of them.

Let $\mathcal{G}$ be a planar embedding of G. Based on Lemma 48 (ii) we distinguish three cases.

(i) There is no connected face-labeling function g such that the connected labeled embedding $(\mathcal{G}, g)$ satisfies $\widehat{\mathcal{A}}$. Then apparently there is no connected labeled embedding satisfying $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$.

(ii) There is just one connected face-labeling function g such that the labeled embedding $(\mathcal{G}, g)$ satisfies $\widehat{\mathcal{A}}$. We further need to check the conditions of the flipping anchors, paths and bicolored edges. For each bicolored edge, we verify its incident labels in constant time.

For each flipping anchor $(w, e, R) \in O_{\widehat{\mathcal{A}}}$ we examine the surrounding of the edge e in the augmented rotation scheme $\rho_{(\mathcal{G},g)}(w)$. We do not have to travel beyond the neighbors of e in $\rho_{\widehat{\mathcal{A}}}(w)$. Notice that even if e is adjacent to a label x in $\rho_{\widehat{\mathcal{A}}}(w)$, we can stop the investigation at the first occurrence of x in $\rho_{(\mathcal{G},g)}(w)$ we encounter, because g is connected and so there cannot be a subsequence (x, y, x, y) in $\rho_{(\mathcal{G},g)}(u)$ for a label y.

We can check each flipping anchor independently on the others. If for a vertex w there are two neighboring flipping anchors (w, e_1, R_1) and (w, e_2, R_2) in $\rho_{\widehat{\mathcal{A}}}(w)$ such that $l_{\widehat{\mathcal{A}}}(e_1) = \ell_1$ and $r_{\widehat{\mathcal{A}}}(e_2) = \ell_2$, then $(e_2, \ell_1, \ell_2, e_1)$ cannot be a subsequence of $\rho_{\mathcal{G},g}(w)$. So if we find occurrences of the labels ℓ_1 and ℓ_2 between e_1 and e_2 that satisfy the conditions of these flipping anchors then the appearances of ℓ_1 and ℓ_2 cannot cross each other.

Therefore, we are able to verify the fulfillment of the conditions induced by the flipping anchors of $Q_{\widehat{\mathcal{A}}}$ in time linear with the total length of the augmented rotation schemes of $(\mathcal{G}, g)$.

For a flipping path (a, b, p, R) we first independently investigate which expansions of the anchors a and b are satisfied. Then in constant time, we compare the lists of satisfied expansions with R. So the fulfillment of the flipping paths can be also checked in linear time with the size of $(\mathcal{G}, g)$.

In total the test whether $(\mathcal{G}, g)$ satisfies $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$ can be done in $\mathcal{O}(|V(G)| + |E(G)| + \left|Q_{\widehat{\mathcal{A}}}\right|)$.

(iii) All the edges of $\widehat{\mathcal{A}}$ are transparent and there is no label appearing in $\widehat{\mathcal{A}}$. Then for each label ℓ the labeled embedding $(\mathcal{G}, g)$ satisfies $\widehat{\mathcal{A}}$ where g is the face-labeling function that assigns ℓ to all the faces of $\mathcal{G}$. There are no flipping anchors or borders, so we just need to find out whether there is a label ℓ that would satisfy all the bicolored edges of $Q_{\widehat{\mathcal{A}}}$. And this can be done in time $\mathcal{O}(\left|Q_{\widehat{\mathcal{A}}}\right|)$. □

Lemma 89. *Let G be a P-skeleton, $\widehat{\mathcal{A}}$ an augmented embedding restriction of G with labeled opaque edges, $O_{\widehat{\mathcal{A}}}$ a set of flipping anchors, $P_{\widehat{\mathcal{A}}}$ a set of edge-disjoint flipping paths and $Q_{\widehat{\mathcal{A}}}$ a set of bicolored edges such that $\left|Q_{\widehat{\mathcal{A}}}\right| \in \mathcal{O}(1)$. Then the instance A*ERCS$(G, \widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$ can be solved in polynomial time.*

Proof. We assume that $O_{\widehat{\mathcal{A}}}$ and $P_{\widehat{\mathcal{A}}}$ are regularized. We show that a satisfiable instance for a P-skeleton has only a constant number of flipping anchors and paths. If the number of flipping anchors and paths exceeds this constant, then we reject. Otherwise, we can generate all the possible expansions of $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$ in linear time. Then we apply Lemma 67 to process these expansions and we accept if at least one of them is successful. The total time complexity of this algorithm is polynomial with respect to the size of G. More precisely, the algorithm can be implemented in time $\mathcal{O}(|E(G)|^6)$.

Let u, v be the vertices of G and let (u, e, R) be a flipping anchor such that $l_{\widehat{\mathcal{A}}}(e) = x$ and $r_{\widehat{\mathcal{A}}}(e) = y$. Then by definition e must be anchored in $\rho_{\widehat{\mathcal{A}}}(u)$. If there is another edge e' anchored in $\rho_{\widehat{\mathcal{A}}}(u)$ such that $l_{\widehat{\mathcal{A}}}(e') \neq r_{\widehat{\mathcal{A}}}(e')$, then the expansions of e to (x, e) and (e, y) in the rotation scheme of u do not lead to a satisfying connected labeled embedding. The label x or y would be disconnected because of the edge e'. So the only the expansion to (e) has a chance to succeed.

If $(\left|O_{\widehat{\mathcal{A}}}\right| + 2\left|P_{\widehat{\mathcal{A}}}\right|) \geq 3$ then without loss of generality there are at least two flipping edges $a_1 = (u, e_1, R_1)$, $a_2 = (u, e_2, R_2)$ at vertex u including the anchors of the flipping paths. If a_1 cannot be expanded to (e_1) or a_2 to (e_2), then the previous observation implies that there is no connected labeled embedding of G satisfying $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$. Otherwise, both a_1 and a_2 must be part of a flipping path, since regularized flipping anchors forbid expansions without extra labels. The regularized flipping paths also prohibit trivial expansions without extra labels. Therefore, there must be another two flipping anchors that cannot be expanded to just the anchored edge. Applying our observation on the new flipping anchors we find out that there cannot be a connected labeled embedding of G satisfying $\widehat{\mathcal{A}}$. □

Open problem 90. *Can be A*ERCS solved in polynomial time for P-skeletons even if the number of bicolored edges is not bounded by a constant?*

Let G be a biconnected graph with a separation pair $\{u, v\}$ and a simple pair of split graphs (H, H') with respect to $\{u, v\}$. Further, let $\widehat{\mathcal{A}}$ be an augmented embedding restriction of G with labeled opaque edges, $Q_{\widehat{\mathcal{A}}}$ a set of flipping anchors of $\widehat{\mathcal{A}}$, $P_{\widehat{\mathcal{A}}}$ a set of flipping paths of $\widehat{\mathcal{A}}$ and $Q_{\widehat{\mathcal{A}}}$ a set of bicolored edges of $\widehat{\mathcal{A}}$. We define the following sets, partitioning $O_{\widehat{\mathcal{A}}}$, $P_{\widehat{\mathcal{A}}}$, $Q_{\widehat{\mathcal{A}}}$ into the parts relevant only for H and H'.

Definition 91.

$$\begin{aligned}
L_E &= \{l_{\widehat{\mathcal{A}}}(e) \mid e \in E(H)\} \cup \{r_{\widehat{\mathcal{A}}}(e) \mid e \in E(H)\} \setminus \{\bigstar\},\\
L'_E &= \{l_{\widehat{\mathcal{A}}}(e) \mid e \in E(H')\} \cup \{r_{\widehat{\mathcal{A}}}(e) \mid e \in E(H')\} \setminus \{\bigstar\},\\
O &= \{(w, e, R) \in O_{\widehat{\mathcal{A}}} \mid e \in E(H)\},\\
O' &= \{(w, e, R) \in O_{\widehat{\mathcal{A}}} \mid e \in E(H')\},\\
P &= \{(a, b, p, R) \in P_{\widehat{\mathcal{A}}} \mid p \textit{ uses only edges of } E(H)\},\\
P' &= \{(a, b, p, R) \in P_{\widehat{\mathcal{A}}} \mid p \textit{ uses only edges of } E(H')\},\\
P_c &= P_{\widehat{\mathcal{A}}} \setminus (P \cup P'),\\
Q &= \{(e, x, y) \in Q_{\widehat{\mathcal{A}}} \mid e \in E(H)\},\\
Q' &= \{(e, x, y) \in Q_{\widehat{\mathcal{A}}} \mid e \in E(H')\},\\
X &= E(H) \cup (L_E \setminus L'_E),\\
X' &= E(H') \cup (L'_E \setminus L_E).
\end{aligned}$$

Similarly to the AERCS problem we can again presume that $|L_E \cup L'_E| \geq 2$. The labeled embeddings of $\widehat{\mathcal{G}}$ satisfying the instances with at most one label in $L_E \cup L'_E$ must have all the faces tagged by one common label. Furthermore, $O_{\widehat{\mathcal{A}}}$ and $P_{\widehat{\mathcal{A}}}$ must be empty. Thus, we can first ignore the labels in $\widehat{\mathcal{A}}$ and solve it as an ERCS instance. If it succeeds, then we verify whether there exists a common label that is consistent with $\widehat{\mathcal{A}}$ and $Q_{\widehat{\mathcal{A}}}$.

4.3.4.1 One label in the intersection

We start with the easiest case $L_E \cap L'_E = \{\ell\}$. In this setting, it is enough to solve one A*ERCS instance for H and one for H'.

Lemma 92. *Let $L_E \cap L'_E = \{\ell\}$, $P_c = \emptyset$ and let $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ be the universal ℓ-division of $\widehat{\mathcal{A}}$ with respect to (H, H'). Then G has a connected labeled embedding satisfying $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$ iff H, H' have connected labeled embeddings satisfying $(\widehat{\mathcal{B}}, O, P, Q)$, $(\widehat{\mathcal{B}}', O', P', Q')$ respectively, and at least one of H and H' is ℓ-passable in $\widehat{\mathcal{B}}$, $\widehat{\mathcal{B}}'$.*

Proof. Let us assume that there are connected labeled embeddings $\mathcal{H}_L$ and $\mathcal{H}'_L$ satisfying $(\widehat{\mathcal{B}}, O, P, Q)$ and $(\widehat{\mathcal{B}}', O', P', Q')$ and that at least one of H and H' is ℓ-passable. Thus there is an expansion $\widehat{\mathcal{D}}$ of $(\widehat{\mathcal{B}}, O, P, Q)$ satisfied by $\mathcal{H}_L$ and an expansion $\widehat{\mathcal{D}}'$ of $(\widehat{\mathcal{B}}', O, P', Q')$ satisfied by $\mathcal{H}'_L$. Let $\widehat{\mathcal{E}}$ be the expansion of $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$ that expands $O_{\widehat{\mathcal{A}}}$, $P_{\widehat{\mathcal{A}}}$ and $Q_{\widehat{\mathcal{A}}}$ in the same way as $\widehat{\mathcal{D}}$ and $\widehat{\mathcal{D}}'$. $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$ is equivalent to an ℓ-division of $\widehat{\mathcal{E}}$ w.r.t. (H, H'). Thus, Lemma 73 implies that there is a connected labeled embedding of G satisfying $\widehat{\mathcal{E}}$. $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$ is only equivalent to a division, because some occurrences of ℓ may be absorbed by the virtual edges $e_G^{virt}(H)$ and $e_G^{virt}(H')$.

The proof of the second implication follows the thought process in the opposite direction. Let $\mathcal{G}_L$ be a connected labeled embedding of G satisfying an expansion $\widehat{\mathcal{E}}$ of $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$. Let $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$ be the universal ℓ-division of $\widehat{\mathcal{E}}$ w.r.t. (H, H'). Then Lemmata 73 and 78 imply that at least one of H and H' is ℓ-passable and that there exist connected labeled embeddings of H and H' satisfying $\widehat{\mathcal{D}}$ and $\widehat{\mathcal{D}}'$ respectively. Let $\widehat{\mathcal{D}}_*$ and $\widehat{\mathcal{D}}'_*$ be respectively the expansions of $(\widehat{\mathcal{B}}, O, P, Q)$ and $(\widehat{\mathcal{B}}', O', P', Q')$ that expand everything in the same way as $\widehat{\mathcal{E}}$. We just need to show that $\widehat{\mathcal{D}}$ is equivalent to $\widehat{\mathcal{D}}_*$ and $\widehat{\mathcal{D}}'$ to $\widehat{\mathcal{D}}'_*$.

Except for the rotation schemes of u and v, the restrictions $\widehat{\mathcal{D}}$ and $\widehat{\mathcal{D}}_*$ are identical. Let us investigate the rotation scheme of u. If there are no flipping anchors in $O_{\widehat{\mathcal{A}}}$ and $P_{\widehat{\mathcal{A}}}$ producing a label in $\widehat{\mathcal{E}}$, then $\rho_{\widehat{\mathcal{D}}}(u)$ and $\rho_{\widehat{\mathcal{D}}_*}(u)$ are also identical. And if there is an anchor $b = (u, e, R)$ that generates a label x in $\widehat{\mathcal{E}}$, then we distinguish two situations. If $x \neq \ell$, then the occurrence of x remains next to the edge e in the division $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$. Else if $x = \ell$, then the generated appearance of ℓ might be absorbed by the virtual edges $e_G^{virt}(H)$ or $e_G^{virt}(H')$ in the division $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$. However, the absorbed label can be restored following Lemma 27. Therefore $\rho_{\widehat{\mathcal{D}}_*}(u)$ is equivalent to $\rho_{\widehat{\mathcal{D}}}(u)$. □

The previous lemma presumed that there are no flipping paths in P_c. Let us look at what happens if P_c is not empty.

Lemma 93. *If $L_E \cap L'_E = \{\ell\}$ and $P_c \neq \emptyset$, then there is no connected labeled embedding of G satisfying $\widehat{\mathcal{A}}$.*

Proof. Let (a, b, p, R) be a flipping path in P_c. Without loss of generality, there are two consecutive edges e, e' of p incident to the vertex u such that $e \in E(H)$ and $e' \in E(H')$. Let us orient the edges e and e' from u outwards. We can further assume that e follows immediately after e' in the rotation scheme $\rho_{\widehat{\mathcal{A}}}(u)$ and $l_{\widehat{\mathcal{A}}}(e) = r_{\widehat{\mathcal{A}}}(e') = \ell$. Then the definition of the flipping paths implies that G has no connected labeled embedding satisfying the augmented embedding restriction $\widehat{\mathcal{A}}^*$, which is identical to $\widehat{\mathcal{A}}$ except for an occurrence of the label ℓ inserted between e and e' in the rotation scheme of the vertex u. However, ℓ is the only label that can tag the two faces that are simultaneously incident to the edges of H and H' in every labeled embedding of G satisfying $\widehat{\mathcal{A}}$. Thus, every labeled embedding of G satisfying $\widehat{\mathcal{A}}$ also satisfies $\widehat{\mathcal{A}}^*$. Therefore, there is no connected labeled embedding of G satisfying $\widehat{\mathcal{A}}$. □

4.3.4.2 Flipping anchors

In this section we investigate the flipping anchors incident to the separation vertices in the cases with $L_E \cap L'_E = \{\ell_1, \ell_2\}$. We consider both the flipping anchors of the set $O_{\widehat{\mathcal{A}}}$ and the flipping anchors contained in the paths of the set $P_{\widehat{\mathcal{A}}}$. Without loss of generality, we can assume that there is no satisfiable (ℓ_2, ℓ_1)-division of $\widehat{\mathcal{A}}$, so we can focus only on the (ℓ_1, ℓ_2)-divisions.

Let $b = (w, e, R)$ be a flipping anchor of $\widehat{\mathcal{A}}$ such that $w \in \{u, v\}$, $e \in E(H)$ and $(x, e) \in R$ for a label x. Next, let $\widehat{\mathcal{E}}$ be an expansion of $\widehat{\mathcal{A}}$ where b is expanded to (x, e) and let $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$ be a universal (ℓ_1, ℓ_2)-division of $\widehat{\mathcal{E}}$. Then the occurrence of x generated by b either stays next to the edge e in the rotation scheme $\rho_{\widehat{\mathcal{D}}}(w)$, or it traverses to the rotation scheme $\rho_{\widehat{\mathcal{D}}'}(w)$ of the second split graph, or it is absorbed by the edge $e_G^{virt}(H)$. Moreover, it is possible that x traverses in one universal (ℓ_1, ℓ_2)-division, but it stays next to e in another one.

Definition 94. *Let $b = (w, e, R)$ be a flipping anchor of $\widehat{\mathcal{A}}$ for $w \in \{u, v\}$, let $\widehat{\mathcal{E}}$ be an expansion of $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$ and let $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$ be a universal (ℓ_1, ℓ_2)-division of $\widehat{\mathcal{E}}$. The division $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$ is* b-traversing *if the expansion of b in $\widehat{\mathcal{E}}$ produces a label x and either $e \in E(H)$ and the generated occurrence of x goes to $\rho_{\widehat{\mathcal{D}}'}(w)$, or $e \in E(H')$ and x goes to $\rho_{\widehat{\mathcal{D}}}(w)$.*

Next, we say that the anchor b affects *the (ℓ_1, ℓ_2)-division w.r.t. (H, H') if there is a b-traversing division $(\widehat{\mathcal{F}}, \widehat{\mathcal{F}}')$ of an expansion of $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$ w.r.t. (H, H') such that both H and H' have connected labeled embeddings satisfying $\widehat{\mathcal{F}}$ and $\widehat{\mathcal{F}}'$.*

Finally, we say that $\varphi \in R$ is always b-traversing *if φ contains a label and for each expansion $\widehat{\mathcal{E}}_*$ of $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$ where b is expanded to φ, only the b-traversing (ℓ_1, ℓ_2)-divisions of $\widehat{\mathcal{E}}_*$ has satisfying connected labeled embeddings. If an (ℓ_1, ℓ_2)-division $(\widehat{\mathcal{D}}_*, \widehat{\mathcal{D}}'_*)$ of $\widehat{\mathcal{E}}_*$ is not b-traversing, then $\widehat{\mathcal{D}}_*$ has no satisfying connected labeled embedding or $\widehat{\mathcal{D}}'_*$ has none. The anchor b is* always traversing *if for each $\psi \in R$, $\psi \neq (e)$, ψ is always b-traversing.*

The flipping anchors that do not affect the division do not cause problems during the division on the separation pair. Independently on the selected expansion, the label generated by a non-affecting flipping anchor (w, e, R) can be left next to the edge e. In some cases, the label might be absorbed by one of the edges $e_G^{virt}(H)$ or $e_G^{virt}(H')$, but we are able to restore it following Lemma 27. It means that if $e \in E(H)$ then the anchor (w, e, R) can be moved to the subproblem for H unchanged. And similarly, for $e \in E(H')$ we can move it to the subproblem for H'.

It is not easy to recognize the flipping anchors affecting the division. However, we are able to prove that the majority of the flipping anchors do not affect the division.

Definition 95. *For $w \in \{u, v\}$ let $\rho_{\widehat{\mathcal{A}}}(w) = (\tau, \lambda, \tau', \lambda')$ be the unique decomposition from Corollary 76. A flipping anchor (w, e, R), e is directed from w outwards, is* marginal *with respect to (H, H') if $w = u$ and at least one of the following conditions holds.*

(i) e is the the first element of τ', $l_{\widehat{\mathcal{A}}}(e) = \ell_1$ and $(\ell_1, e) \in R$,

(ii) e is the the last element of τ', $r_{\widehat{\mathcal{A}}}(e) = \ell_2$ and $(e, \ell_2) \in R$,

(iii) e is the the first element of τ, $l_{\widehat{\mathcal{A}}}(e) = \ell_2$ and $(\ell_2, e) \in R$,

(iv) e is the the last element of τ, $r_{\widehat{\mathcal{A}}}(e) = \ell_1$ and $(e, \ell_1) \in R$.

The anchor is marginal for $w = v$ if at least one of the conditions holds with ℓ_1 and ℓ_2 swapped.

Based on the satisfied conditions we distinguish 6 types of marginal anchors. The types (i) through (iv) fulfill just the one corresponding condition. And then there are two special types (i-ii) and (iii-iv) that satisfy two of the conditions.

Comparing the previous definition with the definition of the universal (ℓ_1, ℓ_2)-division we get the following corollary.

Corollary 96. *A flipping anchor that is not marginal w.r.t. (H, H') do not affect the (ℓ_1, ℓ_2)-division w.r.t. (H, H').*

It means that there are at most eight flipping anchors affecting the division. We can further improve this bound to four.

Lemma 97. *Let $w \in \{u, v\}$ and let $a_1 = (w, e_1, R_1)$ and $a_2 = (w, e_2, R_2)$ be two different flipping anchors of $\widehat{\mathcal{A}}$ such that either a_1, a_2 are of types (i) and (ii), or they are of types (iii) and (iv). Then, there is no connected labeled embedding satisfying $\widehat{\mathcal{A}}$.*

Proof. Without loss of generality, let a_1, a_2 be of types (i) and (ii). For each (ℓ_1, ℓ_2)-division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ of $\widehat{\mathcal{A}}$ the scheme $\rho_{\widehat{\mathcal{B}}'}(w)$ contains $(e_1, e_2, e_G^{virt}(H'))$ as a subsequence. The incident labels of these edges enforce the subsequence of labels $(\ell_1, \ell_2, \ell_1, \ell_2)$, so there is no connected labeled embedding of H' satisfying $\widehat{\mathcal{B}}'$. □

For simplicity, the three following lemmata speak only about the marginal flipping anchors of type (i) that are incident to the vertex u. However, similar statements can be proven for the other marginal flipping anchors.

The marginal flipping anchors of types (i) through (iv) can have two different expansions generating labels. One of these expansions is required by the definition, but the second is not mandatory. We show that the second expansion can be never satisfied. So for each flipping anchor, we can remove this unsatisfiable expansion from the list of possible expansions and regularize the flipping anchor (path).

Lemma 98. *Let $b = (u, e, R)$ be a marginal flipping anchor of type (i), such that $r_{\widehat{\mathcal{A}}}(e) = x$ and $(e, x) \in R$. Then no connected labeled embedding of G satisfies an expansion of $\widehat{\mathcal{A}}$ where b is expanded to (e, x).*

Proof. Let $\widehat{\mathcal{E}}$ be an expansion of $\widehat{\mathcal{A}}$ where b is expanded to (e, x). Then in every universal (ℓ_1, ℓ_2)-division $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$ of $\widehat{\mathcal{E}}$ the rotation scheme $\rho_{\widehat{\mathcal{D}}'}(u)$ contains $(e_G^{virt}(H'), e, x)$ as a subsequence. This holds even for $x = \ell_2$, because b is not of type (i-ii). Thus, every labeled embedding satisfying $\widehat{\mathcal{D}}'$ must have (ℓ_1, x, ℓ_1, x) as a subsequence of the augmented rotation scheme of u. And so it cannot be connected. □

Next, we show that only a fraction of the marginal flipping anchors can affect the division.

Lemma 99. *If a marginal flipping anchor $b = (u, e, R)$ of type (i) affects the (ℓ_1, ℓ_2)-division w.r.t. (H, H'), then the decomposition of the rotation scheme $\rho_{\widehat{\mathcal{A}}}(u)$ according to Corollary 76 must be either (τ, τ') or (ℓ_1, τ, τ').*

Proof. We assume that the anchor b is regularized, so there is not an occurrence of ℓ_1 preceding τ' in $\rho_{\widehat{\mathcal{A}}}(u)$. Let us look at all the remaining $(\tau, \lambda, \tau', \lambda')$ decompositions of $\rho_{\widehat{\mathcal{A}}}(u)$.

If $\rho_{\widehat{\mathcal{A}}}(u)$ matches one of $(\ell_2, \ell_1, \tau, \ell_2, \tau')$, $(\ell_2, \tau, \ell_1, \ell_2, \tau')$, $(\ell_2, \ell_1, \tau, \ell_1, \ell_2, \tau')$, $(\ell_2, \ell_1, \ell_2, \tau, \tau')$, $(\ell_2, \ell_1, \ell_2, \tau, \ell_2, \tau')$, then the rotation scheme $\rho_{\widehat{\mathcal{A}}}(u)$ together with the condition $l_{\widehat{\mathcal{A}}}(e) = \ell_1$ enforce the subsequence $(\ell_1, \ell_2, \ell_1, \ell_2)$. So there cannot be any connected labeled embedding of G satisfying $\widehat{\mathcal{A}}$.

If $\rho_{\widehat{\mathcal{A}}}(u) = (\ell_2, \ell_1, \tau, \tau')$ then for each universal (ℓ_1, ℓ_2)-division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ of $\widehat{\mathcal{A}}$ the rotation scheme $\rho_{\widehat{\mathcal{B}}'}(u) = (\ell_2, e_G^{virt}(H'), \tau')$. It again enforces the subsequence $(\ell_2, \ell_1, \ell_2, \ell_1)$. So there also is no connected labeled embedding satisfying $\widehat{\mathcal{A}}$.

If $\rho_{\widehat{\mathcal{A}}}(u)$ matches one of (τ, ℓ_2, τ'), $(\ell_1, \tau, \ell_2, \tau')$, $(\ell_1, \ell_2, \tau, \tau')$, $(\ell_1, \ell_2, \tau, \ell_2, \tau')$, $(\tau, \ell_1, \ell_2, \tau')$, $(\ell_1, \tau, \ell_1, \ell_2, \tau')$, then the flipping anchor b does not affect the (ℓ_1, ℓ_2)-division. The expanded label ℓ_1 always stays next to the edge e in each universal (ℓ_1, ℓ_2)-division.

If $\rho_{\widehat{\mathcal{A}}}(u) = (\ell_2, \tau, \tau')$ then there are two possibilities how to divide $\rho_{\widehat{\mathcal{A}}}(u)$. We either put $\rho_1 = (\tau, e_G^{virt}(H))$ and $\rho_1' = (\ell_2, e_G^{virt}(H'), \tau')$, or $\rho_2 = (\ell_2, \tau, e_G^{virt}(H))$ and $\rho_2' = (e_G^{virt}(H'), \tau')$. Notice that ρ_1' cannot be satisfied, because it enforces the subsequence $(\ell_2, \ell_1, \ell_2, \ell_1)$. Similarly, for an expansion $\widehat{\mathcal{E}}$ of $\widehat{\mathcal{A}}$, that generates the label ℓ_1 next to e, we get $\rho_{\widehat{\mathcal{E}}}(u) = (\ell_2, \tau_{exp}, \ell_1, \tau'_{exp})$, where τ_{exp} and τ'_{exp} are the sequences produced form τ and τ' by the expansion $\widehat{\mathcal{E}}$. There are again up to two possible divisions. Either $\rho_3 = (\tau_{exp}, \ell_1, e_G^{virt}(H))$ and $\rho_3' = (\ell_2, e_G^{virt}(H'), \tau'_{exp})$, or $\rho_4 = (\ell_2, \tau_{exp}, e_G^{virt}(H))$ and $\rho_4' = (e_G^{virt}(H'), \ell_1, \tau'_{exp})$. Again, ρ_3' cannot be satisfied, because it enforces the subsequence $(\ell_2, \ell_1, \ell_2, \ell_1)$. Thus, it makes sense to consider only the divisions with ρ_4 and ρ_4', where the expanded label ℓ_1 stays next to the edge e. Thus, the flipping anchor b does not affect the (ℓ_1, ℓ_2)-division.

Finally, if $\rho_{\widehat{\mathcal{A}}}(u) = (\ell_2, \tau, \ell_2, \tau')$ then the anchor b also does not affect the (ℓ_2, ℓ_1)-division. The situation is similar to the previous one, but it admits only one possible division of the rotation scheme of u in cases when the expansion generates ℓ_1 next to e. □

Let us look closer at the marginal flipping anchor $b = (u, e, R)$ of type (i) when $\rho_{\widehat{\mathcal{A}}}$ is (τ, τ') or (ℓ_1, τ, τ'). We consider the two possible expansions of b. For simplicity, we assume that there are no other flipping anchors incident to u. If in an expansion $\widehat{\mathcal{E}}$ of $\widehat{\mathcal{A}}$ the anchor b does not produce the label ℓ_1, then in every universal (ℓ_1, ℓ_2)-division $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$ of $\widehat{\mathcal{E}}$ we have $\rho_{\widehat{\mathcal{D}}}(u) = (\tau, e_G^{virt}(H))$ and $\rho_{\widehat{\mathcal{D}}'}(u) = (e_G^{virt}(H'), \tau')$. However, if b generates the label ℓ_1 then there are two ways how to split the rotation scheme of u. A universal (ℓ_1, ℓ_2)-division uses either the rotation schemes $\rho_1(u) = (\tau, e_G^{virt}(H))$ and $\rho_1'(u) = (e_G^{virt}(H'), \ell_1, \tau')$, or $\rho_2(u) = (\tau, \ell_1, e_G^{virt}(H))$ and $\rho_2'(u) = (e_G^{virt}(H'), \tau')$.

Lemma 100. *Let $b = (u, e, R)$ is a marginal flipping anchor of type (i) and let $\widehat{\mathcal{E}}$ be an expansion of $\widehat{\mathcal{A}}$ where b is expanded to (ℓ_1, e). Next, let $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$ be a universal (ℓ_1, ℓ_2)-division of $\widehat{\mathcal{E}}$ with $\rho_{\widehat{\mathcal{D}}'}(u) = (e_G^{virt}(H'), \ell_1, \tau')$. And finally, let $\widehat{\mathcal{D}}^*$ be the augmented embedding restriction of H' that is identical to $\widehat{\mathcal{D}}'$ except for $\rho_{\widehat{\mathcal{D}}^*}(u) = (e_G^{virt}(H'), \tau')$. Then H' has no connected labeled embedding satisfying $\widehat{\mathcal{D}}'$, or every connected labeled embedding satisfying $\widehat{\mathcal{D}}^*$ also satisfies $\widehat{\mathcal{D}}'$.*

Proof. If $r_{\widehat{\mathcal{A}}}(e) = \ell_2$, then every labeled embedding of H' satisfying $\widehat{\mathcal{D}}'$ has $(\ell_1, \ell_2, \ell_1, \ell_2)$ as a subsequence of the augmented rotation scheme of u, so it is not connected.

Otherwise, let $r_{\widehat{\mathcal{A}}}(e) = x \neq \ell_2$ and let G_L^u be the label ordering graph of u in $\widehat{\mathcal{D}}'$ (Definition 32). If G_L^u does not have an Eulerian circuit without crossings then there is no connected labeled embedding satisfying $\widehat{\mathcal{D}}'$. So we can further assume that G_L^u has an Eulerian circuit ε without crossings. We show that exactly one of the following statements is true.

(i) There is a directed path from x to ℓ_1 avoiding ℓ_2 in G_L^u.

(ii) There is a directed path from x to ℓ_2 avoiding ℓ_1 in G_L^u.

The existence of the circuit ε implies that at least one of them is true. We can fix an arbitrary occurrence of x in ε and start traversing G_L^u following the circuit ε. If we visit ℓ_1 before ℓ_2, then (i) holds. Otherwise, (ii) is satisfied.

For a contradiction, let us assume that both (i) and (ii) holds. We have a directed path p_1 from x to ℓ_1 avoiding ℓ_2 and path p_2 from x to ℓ_2 avoiding ℓ_1. Let w be the last vertex of p_2 that lies on p_1. Then there are two directed edge-disjoint paths from w to ℓ_1 in G_L^u. The first one is the suffix of p_1 starting at w. The second is the suffix of p_2 starting at w plus the edge from ℓ_2 to ℓ_1 that corresponds to the edge $e_G^{virt}(H')$. Lemma 36 then implies that each Eulerian circuit of G_L^u has a crossing, but ε does not have any.

If the statement (i) is false, then every directed path from x to ℓ_1 visits ℓ_2. Therefore, for each labeled embedding of H' satisfying $\widehat{\mathcal{D}}'$ the augmented rotation scheme of u contains $(\ell_1, \ell_2, \ell_1, \ell_2)$ as a subsequence. There must be ℓ_2 present between the edge e and the occurrence of ℓ_1 from $\rho_{\widehat{\mathcal{D}}'}(u)$. So in this case, there is no connected labeled embedding satisfying $\widehat{\mathcal{D}}'$.

Otherwise, each directed path from x to ℓ_2 passes through ℓ_1. It means that every labeled embedding satisfying $\widehat{\mathcal{D}}^*$ also satisfies $\widehat{\mathcal{D}}'$. □

Moreover, we are able to recognize the two possible outcomes of Lemma 100 in linear time with respect to the length of the rotation scheme $\rho_{\widehat{\mathcal{D}}'}(u)$ by examining an Euler circuit of the label ordering graph of u in $\widehat{\mathcal{D}}'$.

If each connected labeled embedding satisfying $\widehat{\mathcal{D}}^*$ also satisfies $\widehat{\mathcal{D}}'$, then the expansion of b to (ℓ_1, e) is unnecessary. In this situation, every labeled embedding $\mathcal{H}'_L$ of H' satisfying the corresponding part of a universal (ℓ_1, ℓ_2)-division of $\widehat{\mathcal{A}}$ has ℓ_1 between the edges $e_G^{virt}(H')$ and e in the augmented rotation scheme $\rho_{\mathcal{H}'_L}(u)$. It means that we can remove (ℓ_1, e) from the list of possible expansions of the anchor b.

And if we find out that there is no connected labeled embedding of H' satisfying $\widehat{\mathcal{D}}'$, then we do not have to consider the universal (ℓ_1, ℓ_2)-divisions that split the rotation scheme of u into $\rho_1(u) = (\tau, e_G^{virt}(H))$ and $\rho'_1(u) = (e_G^{virt}(H'), \ell_1, \tau')$. We can solely focus on the universal divisions with $\rho_2(u) = (\tau, \ell_1, e_G^{virt}(H))$ and $\rho'_2(u) = (e_G^{virt}(H'), \tau')$. Therefore, the expansion $(\ell_1, e) \in R$ is always b-traversing.

Corollary 101. *A marginal flipping anchor $b = (u, e, R)$ of type (i) affects the (ℓ_1, ℓ_2)-division, only if $(\ell_1, e) \in R$ is always b-traversing.*

As we have already mentioned, similar propositions to the last three lemmata plus the previous corollary can be proven about the other marginal flipping anchors. The following definition put together our findings about the flipping anchors affecting the division.

Definition 102. *Let $b = (w, e, R)$, $w \in \{u, v\}$, be a marginal flipping anchor and let the label ordering graph G_L^w of w in $\widehat{\mathcal{A}}$ have an Eulerian circuit without crossings. For $w = u$ we say that b is* suspicious *w.r.t. (H, H') if one of the following conditions is true:*

(i) b is of type (i), $\rho_{\widehat{\mathcal{A}}}(w)$ is (τ, τ') or (ℓ_1, τ, τ'), and there is no directed path from $r_{\widehat{\mathcal{A}}}(e)$ to ℓ_1 avoiding ℓ_2 in G_L^w.

(ii) b is of type (ii), $\rho_{\widehat{\mathcal{A}}}(w)$ is (τ, τ') or (τ, ℓ_2, τ'), and there is no directed path from ℓ_2 to $l_{\widehat{\mathcal{A}}}(e)$ avoiding ℓ_1 in G_L^w.

(iii) b is of type (iii), $\rho_{\widehat{\mathcal{A}}}(w)$ is (τ, τ') or (τ, ℓ_2, τ'), and there is no directed path from $r_{\widehat{\mathcal{A}}}(e)$ to ℓ_2 avoiding ℓ_1 in G_L^w.

(iv) b is of type (iv), $\rho_{\widehat{\mathcal{A}}}(w)$ is (τ, τ') or (ℓ_1, τ, τ'), and there is no directed path from ℓ_1 to $l_{\widehat{\mathcal{A}}}(e)$ avoiding ℓ_2 in G_L^w.

($*$) b is of type (i-ii) or (iii-iv) and $\rho_{\widehat{\mathcal{A}}}(w)$ is (τ, τ'), (ℓ_1, τ, τ') or (τ, ℓ_2, τ').

For $w = v$ the flipping anchor b is suspicious *w.r.t. (H, H') if one of the previous conditions holds with ℓ_1 and ℓ_2 swapped.*

Corollary 103. *If a flipping anchor affects the (ℓ_1, ℓ_2)-division, then it is suspicious.*

Corollary 104. *Every suspicious flipping anchor is always traversing.*

The set of suspicious anchors contains all the flipping anchors affecting the division and all the anchors in the set are always traversing. Moreover, the set can be constructed in linear time with respect to the lengths of $\rho_{\widehat{\mathcal{A}}}(u)$ and $\rho_{\widehat{\mathcal{A}}}(v)$. Therefore, the set of suspicious flipping anchors is a good approximation for the set of flipping edges that affect the (ℓ_1, ℓ_2)-division.

So far we investigated which flipping anchors affect the universal division. But how do the affecting flipping anchors interact with each other?

Lemma 105. *Let $w \in \{u, v\}$ and let $a_1 = (w, e_1, R_1)$, $a_2 = (w, e_2, R_2)$ be two suspicious flipping anchors of $\widehat{\mathcal{A}}$. If an expansion $\widehat{\mathcal{E}}$ of $\widehat{\mathcal{A}}$ has a satisfying connected labeled embedding, then a_1 must be expanded to (e_1) and a_2 to (e_2).*

Proof. Without loss of generality, let $w = u$. We have already covered the case when a_1 and a_2 are of types (i) and (ii), or (iii) and (iv) in Lemma 97.

Let a_1 be of type (i). For a contradiction let us assume that a_1 is expanded to (ℓ_1, e_1). Then, the rotation scheme of u must be of form (τ, τ') or (ℓ_1, τ, τ'), because a_1 is always traversing. Let $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$ be a a_1-traversing (ℓ_1, ℓ_2)-division of an expansion of $\widehat{\mathcal{A}}$. Hence, $\rho_{\widehat{\mathcal{D}}}(u) = (\tau, \ell_1, e_G^{virt}(H))$.

If a_2 is of type (iv) or (iii-iv), then the edges e_2 and $e_G^{virt}(H)$ together with the label ℓ_1 enforce the subsequence $(\ell_1, \ell_2, \ell_1, \ell_2)$. Therefore, there is no connected labeled embedding satisfying $\widehat{\mathcal{D}}$.

If a_2 is of type (iii), then there is also no labeled embedding satisfying $\widehat{\mathcal{D}}$, because a_2 is also always traversing, therefore no labeled embedding of H satisfying the corresponding part of a universal (ℓ_1, ℓ_2)-division of $\widehat{\mathcal{A}}$ has label ℓ_1 between the edges e_2 and $e_G^{virt}(H)$ in the augmented rotation scheme of the vertex u.

The other configurations are symmetric to the one mentioned. A marginal flipping anchor of type (i–ii) behaves as a flipping anchor of type (i) or (ii). □

Given an instance A*ERCS$(G, \widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$, we can modify the sets $O_{\widehat{\mathcal{A}}}$ and $P_{\widehat{\mathcal{A}}}$ in linear time with respect to the lengths of $\rho_{\widehat{\mathcal{A}}}(u)$ and $\rho_{\widehat{\mathcal{A}}}(v)$ in such a way that we get an equivalent instance with at most one suspicious flipping anchor incident to u and at most one incident to v.

Another fact worth noting is that if there is a suspicious flipping anchor b incident to a vertex $w \in \{u, v\}$ then for each expansion of b there is only one reasonable option how to split the rotation scheme of w in universal (ℓ_1, ℓ_2)-divisions.

4.3.4.3 Two labels in the intersection

In this section, we show how to proceed in the cases where $L_E \cup L'_E = \{\ell_1, \ell_2\}$. We still assume that there is no satisfiable (ℓ_2, ℓ_1)-division of $\widehat{\mathcal{A}}$, so we focus on the (ℓ_1, ℓ_2)-divisions. There are three different phenomena that can cause some complications.

Firstly, the set of universal (ℓ_1, ℓ_2)-divisions of $\widehat{\mathcal{A}}$ may contain more than one element. In this case, there is a vertex $w \in \{u, v\}$ for which we have two options on how to split the rotation scheme $\rho_{\widehat{\mathcal{A}}}(w)$. This is the original cause that motivated us to establish the flipping anchors and paths.

Secondly, we may encounter some flipping anchors incident to the separation vertices affecting the universal division. The flipping anchors were analyzed in the previous section.

And thirdly, there might be flipping paths connecting the split graphs H and H'. These are the paths of the set P_c. We can easily observe that there are at most 4 paths in P_c because at most two flipping paths can pass through the vertex u and at most two through v.

In addition, more than one of these phenomena can arise simultaneously. Later, we demonstrate which combinations are possible and how to resolve them. But, we start with the simplest case with no suspicious flipping anchors and $P_c = \emptyset$.

Lemma 106. *Let $L_E \cap L'_E = \{\ell_1, \ell_2\}$, $P_c = \emptyset$, let there be no suspicious flipping anchors in $O_{\widehat{\mathcal{A}}}$ and $P_{\widehat{\mathcal{A}}}$ and let U be the set of universal (ℓ_1, ℓ_2)-divisions of $\widehat{\mathcal{A}}$ w.r.t. (H, H'). Then G has a connected labeled embedding satisfying $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$ iff there exists a division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}') \in U$ such that H, H' have connected labeled embeddings satisfying $(\widehat{\mathcal{B}}, O, P, Q)$, $(\widehat{\mathcal{B}}', O', P', Q')$ respectively.*

Proof. The proof utilize the same idea that was used in Lemma 92.

Let us assume that there exists $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}') \in U$ such that there are two connected labeled embeddings $\mathcal{H}_L$ and $\mathcal{H}'_L$ satisfying $(\widehat{\mathcal{B}}, O, P, Q)$ and $(\widehat{\mathcal{B}}', O, P, Q)$ respectively. There must be an expansion $\widehat{\mathcal{D}}$ of $(\widehat{\mathcal{B}}, O, P, Q)$ satisfied by $\mathcal{H}_L$ and an expansion $\widehat{\mathcal{D}}'$ of $(\widehat{\mathcal{B}}', O', P', Q')$ satisfied by $\mathcal{H}'_L$. Let $\widehat{\mathcal{E}}$ be the expansion of $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$ that expands $O_{\widehat{\mathcal{A}}}$, $P_{\widehat{\mathcal{A}}}$ and $Q_{\widehat{\mathcal{A}}}$ in the same way as $\widehat{\mathcal{D}}$ and $\widehat{\mathcal{D}}'$. $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$ is equivalent to a (ℓ_1, ℓ_2)-division of $\widehat{\mathcal{E}}$ w.t.r. (H, H'). Lemma 73 then implies that there is a connected labeled embedding of G satisfying $\widehat{\mathcal{E}}$. $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$ is only equivalent to a division, because some labels may be absorbed by the virtual edges $e_G^{virt}(H)$ and $e_G^{virt}(H')$

The second implication follows a similar thought process in the opposite direction. We start with a connected labeled embedding $\mathcal{G}_L$ of G satisfying an expansion $\widehat{\mathcal{E}}$ of $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$. Lemmata 73 and 80 imply that there is a universal (ℓ_1, ℓ_2)-division $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$ of $\widehat{\mathcal{E}}$ w.r.t. (H, H') such that both $\widehat{\mathcal{D}}$ and $\widehat{\mathcal{D}}'$ have satisfying connected labeled embeddings. Since there are no flipping anchors affecting the division, then each label in $\widehat{\mathcal{E}}$ produced by an expansion of an anchor goes to the same restriction as the edge of the anchor. We can imagine that for each anchor the edge and the generated label act together as an inseparable unit during the division. Therefore, there exists $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}') \in U$ such that $\widehat{\mathcal{D}}$ is equivalent to an expansion of $(\widehat{\mathcal{B}}, O, P, Q)$ and $\widehat{\mathcal{D}}'$ to an expansion of $(\widehat{\mathcal{B}}', O', P', Q')$. □

So if $\widehat{\mathcal{A}}$ has just one universal (ℓ_1, ℓ_2)-division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$, then we can first solve the instance A*ERCS$(H, \widehat{\mathcal{B}}, O, P, Q)$ in polynomial time for an [SPQR]-skeleton H. If it fails then we reject. Otherwise, we make a recursive call for the instance A*ERCS$(H', \widehat{\mathcal{B}}', O', P', Q')$.

Now let us look at the more complicated cases. Our strategy for such situations is to expand the problematic elements and then consider all the relevant universal (ℓ_1, ℓ_2)-divisions of these expansions with respect to (H, H'). This process yields a set T of pairs of A*ERCS instances where the first component of each pair is an instance for the split graph H and the second is an instance for H'. From Lemma 106 it follows that the graph G has a connected labeled embedding satisfying $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$ iff there exists a pair $t \in T$ such that both the instances of t accept.

The size of the set T is bounded by a constant, so if H is an [SPQR]-skeleton then we can solve in polynomial time all the instances for H in T. We start with an empty set T' and for each $t \in T$ we add the H'-instance of t into T' iff the H-instance of t accepts. Then, G has the desired embedding iff there is a satisfiable instance in T'.

We show that the instances in T' can be merged into one by adding a flipping anchor or a path. Therefore, it is possible to investigate whether there is a satisfiable instance in T' using just one recursive call.

It remains to show how to do the merging of the instances in T'. First, we define the augmented embedding restriction $\widehat{\mathcal{B}}'_*$ that is used in the instance replacing T'. If there is just one universal (ℓ_1, ℓ_2)-division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ of $\widehat{\mathcal{A}}$ w.r.t. (H, H'), then we put $\widehat{\mathcal{B}}'_* = \widehat{\mathcal{B}}'$. Otherwise there are several occurrences of the labels ℓ_1 and ℓ_2 in $\rho_{\widehat{\mathcal{A}}}(u)$ and $\rho_{\widehat{\mathcal{A}}}(v)$, that in some universal divisions go to the restriction for H and in some to the restriction for H'. Let $\widehat{\mathcal{A}}_*$ be the augmented embedding restriction obtained from $\widehat{\mathcal{A}}$ by removing these appearances of ℓ_1 and ℓ_2. $\widehat{\mathcal{A}}_*$ has only one universal (ℓ_1, ℓ_2)-division $(\widehat{\mathcal{B}}_*, \widehat{\mathcal{B}}'_*)$ w.r.t. (H, H').

We examine all the possible situations that may arise. For each of them, we describe a flipping anchor or a flipping path that should be added to $(\widehat{\mathcal{B}}'_*, O', P', Q')$. We always consider the most general case, so it may be possible to regularize the added flipping path or anchor. For example, if T' contains too few elements, then the proposed flipping path can change just to a flipping anchor or it may disappear completely.

First, we consider the settings with just one problematic element. We assume that all the flipping anchor of $O_{\widehat{\mathcal{A}}}$ and the paths of $P_{\widehat{\mathcal{A}}}$ are regularized and that there is at most one suspicious flipping anchor incident to u and at most one incident to v.

Two universal divisions There are two possibilities how to split the rotation scheme $\rho_{\widehat{\mathcal{A}}}(u)$ in the universal (ℓ_1, ℓ_2)-divisions. In this case we add a flipping anchor $b = (u, e_G^{virt}(H'), R)$ where R describes the differences of the universal divisions in T' from $\widehat{\mathcal{B}}'_*$.

One suspicious anchor There is one suspicious flipping anchor $b = (u, e, R)$ incident to u. We further distinguish four cases.

If $b \in O'$, then we do not have to add anything, because the suspicious anchors are always traversing.

If $b \in O$, then we add a flipping anchor $b' = (u, e_G^{virt}(H'), R')$ where R' covers the always traversing expansions of b in restrictions of T'.

If b is a flipping anchor of a flipping path $p \in P'$, then let $b' = (w, e', R')$ be the second anchor of p. We recycle the anchor b' as the flipping anchor $b'' = (w, e', R'')$ for a subset $R'' \subseteq R'$ covering the restrictions of T'.

Else b is a flipping anchor of a path $p \in P$. We add the flipping anchor $b' = (u, e_G^{virt}(H'), R')$ where R' covers the always traversing expansions of b in restrictions of T'.

One flipping path There is one flipping path $p \in P_c$, going from H to H' through u. Let $b = (w, e, R)$ be the flipping anchor of p in H'. We recycle the anchor b as the flipping anchor $b' = (w, e, R')$ for a subset $R' \subseteq R$ covering the restrictions of T'.

Two complications at one vertex We have described how to react if we encounter just one complication at one of the separation vertices. But what if there are two different issues at the same vertex? We prove that if such a thing happens, then there is no connected labeled embedding of G satisfying $\widehat{\mathcal{A}}$.

In the section about flipping anchors, we demonstrated how to reduce the number of suspicious flipping anchors incident to a separation vertex to one. We also observed that if there is a suspicious anchor incident to $w \in \{u, v\}$, then there is always only one relevant way how to split the rotation scheme of w in the universal (ℓ_1, ℓ_2)-divisions. So we just have to investigate the cases involving flipping paths of P_c.

There are only two possibilities of how a flipping path can pass from H to H' through u. The rotation scheme $\rho_{\widehat{\mathcal{A}}}(u)$ must be of form $(\tau, \lambda, \tau', \lambda')$ and the flipping path must use either the first edge of τ and the last edge of τ', or the last edge of τ and the first edge of τ'. The definition further demands that every two consecutive edges of a flipping path share a label prescribed for the incident faces. The only two candidates that can fulfill this condition through a separation vertex are ℓ_1 and ℓ_2. But that is not all. If a flipping path with the first edge of τ and the last edge of τ' uses the label ℓ_1 to pass through the vertex u, then there is no successful (ℓ_1, ℓ_2)-division of $\widehat{\mathcal{A}}$, because no connected labeled embedding of G can tag by ℓ_1 the face simultaneously incident to edges of H and H', between H' and H in the counterclockwise direction. Thus, if we look for a (ℓ_1, ℓ_2)-division, then we are down to two possibilities. The first one is a path containing the first edge of τ and using the label ℓ_2. It further requires that λ' is empty. The second one is a path with the last edge of τ using ℓ_1 and demanding that λ is empty.

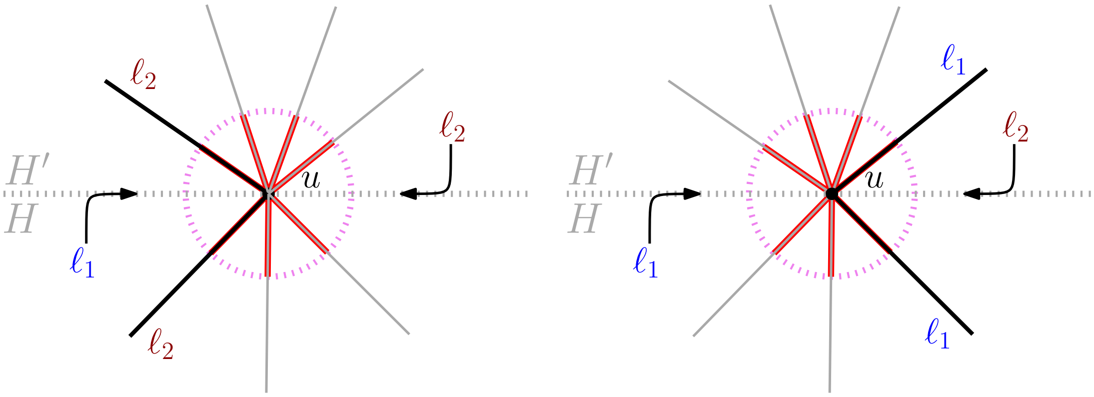


Figure 4.6: Two possibilities how a flipping path can pass from H' to H through the separation vertex u. The flipping path is highlighted in black. The labels ℓ_1 and ℓ_2 next to the arrows do not appear in the rotation scheme $\rho_{\widehat{\mathcal{A}}}(u)$, but they are enforced by the (ℓ_1, ℓ_2)-division.

If we detect a flipping path going form H to H' through u that does not satisfy one of the two mentioned conditions, then we reject. We can easily observe that both of these options cannot happen simultaneously because they enforce the subsequence $(\ell_1, \ell_2, \ell_1, \ell_2)$ in the rotation scheme of u for both the split graphs. Similarly, there cannot be a suspicious flipping anchor incident to u, since it would create the same problem in the split graph containing the edge of the anchor. Lastly, we have the complication with two possibilities for the splitting of the rotation scheme. Notice that the flipping path with the first edge of τ allows this extra complication only with $\rho_{\widehat{\mathcal{A}}}(u) = (\tau, \ell_1, \tau')$, because λ' must be empty. But we are again in the same situation. The split graph receiving the problematic occurrence of ℓ_1 enforces the subsequence $(\ell_1, \ell_2, \ell_1, \ell_2)$ for the rotation scheme of u. Therefore, each connected labeled embedding of G satisfying $\widehat{\mathcal{A}}$ tolerates at most one complication at one separation vertex. If there are at least two, then we reject.

We described all the possible cases where only one of the separation vertices met some complications. Now let us look what happens when there are issues with both u and v.

Four universal divisions There are two options how to split $\rho_{\widehat{\mathcal{A}}}(u)$ and $\rho_{\widehat{\mathcal{A}}}(v)$. In this case, we add a flipping path p consisting only of the edge $e_G^{virt}(H')$ such that the expansions of p cover the restrictions in T'.

One suspicious anchor and two universal divisions There is a suspicious flipping anchor $b = (u, e, R)$ incident to the vertex u and there are two ways how to split the rotation scheme $\rho_{\widehat{\mathcal{A}}}(v)$. We further distinguish four cases based on the position of b.

If $b \in O'$, then we add just a flipping anchor $b' = (v, e_G^{virt}(H'), R)$.

If $b \in O$, then we add a flipping path p consisting only of the edge $e_G^{virt}(H')$.

If b is a flipping anchor of a flipping path $p \in P'$, then we extend the path p to the vertex v using the edge $e_G^{virt}(H')$. The expansions of the new extended path cover the restrictions in T'.

Else b is a flipping anchor of a path $p \in P$. Then we add a flipping path p consisting only of the edge $e_G^{virt}(H')$.

One flipping path and two universal divisions There is a flipping path $p \in P_c$ going from H to H' through u and there are two options how to split the rotation scheme $\rho_{\widehat{\mathcal{A}}}(v)$. In this case, we reroute the path p to the vertex v using the edge $e_G^{virt}(H')$. We recycle the part of p in H'. The expansions of the new rerouted path cover the restrictions in T'.

Two suspicious anchors There are two suspicious anchors $b_u = (u, e_u, R_u)$ and $b_v = (v, e_v, R_v)$. We further distinguish several cases.

If $b_u \in O'$, then we do not have to do anything for the anchor b_u, because the suspicious anchors are always traversing. We just have to resolve the anchor b_v following the directions for one suspicious anchor.

If both $b_u, b_v \in O$, then we add a flipping path consisting only of the edge $e_G^{virt}(H')$.

If $b_u \in O$ and b_v is a flipping anchor of a flipping path $p_v \in P'$, then we extend the path p_v to the vertex u using the edge $e_G^{virt}(H')$.

If $b_u \in O$ and b_v is a flipping anchor of a flipping path $p_v \in P$, then we add a flipping path consisting only of the edge $e_G^{virt}(H')$.

If b_u is a flipping anchor of a flipping path $p_u \in P'$ and b_v is an anchor of a flipping path $p_v \in P'$, then we connect the two paths using the edge $e_G^{virt}(H')$. However, if $p_u = p_v$, then we do not add anything.

If b_u is a flipping anchor of a flipping path $p_u \in P'$ and b_v is an anchor of a flipping path $p_v \in P$, then we extend the path p_u to the vertex v using the edge $e_G^{virt}(H')$.

Else b_u is a flipping anchor of a path $p_u \in P$ and b_v is a flipping anchor of a path $p_v \in P$. Then we add a flipping path consisting only of the edge $e_G^{virt}(H')$.

One suspicious anchor and one flipping path There is a suspicious flipping anchors $b_u = (u, e_u, R_u)$ and there is a flipping path $p_v \in P_c$ going from H to H' through v. We further distinguish several cases based on the position of the anchor b. Also sometimes we react differently if b is an anchor of p_v.

If $b_v \in O'$, then let $b_v = (w, e_v, R_v)$ be the the anchor of p_v in H'. We just recycle the anchor b_v as the flipping anchor $b' = (w, e_v, R')$ for a subset $R' \subseteq R_v$ covering the restrictions of T'.

If $b_u \in O$, then we reroute the path p_v to the vertex u using the edge $e_G^{virt}(H')$. We recycle the part of p_v in H'.

If b_u is a flipping anchor of a flipping path $p_u \in P'$, then we connect the path p_u and the part of p_v in H' using the edge $e_G^{virt}(H')$. However, if $p_u = p_v$, then we do not add anything.

Else b_u is a flipping anchor of a path $p_u \in P$. Then we reroute the path p_v to the vertex u using the edge $e_G^{virt}(H')$. We recycle the part of p_v in H'.

Two flipping paths There is a flipping path $p_u \in P_c$ going from H to H' through u and there is a flipping path $p_v \in P_c$ going from H to H' through v.

If $p_u \neq p_v$, then we take the parts of p_u and p_v in H' and connect them using the edge $e_G^{virt}(H')$.

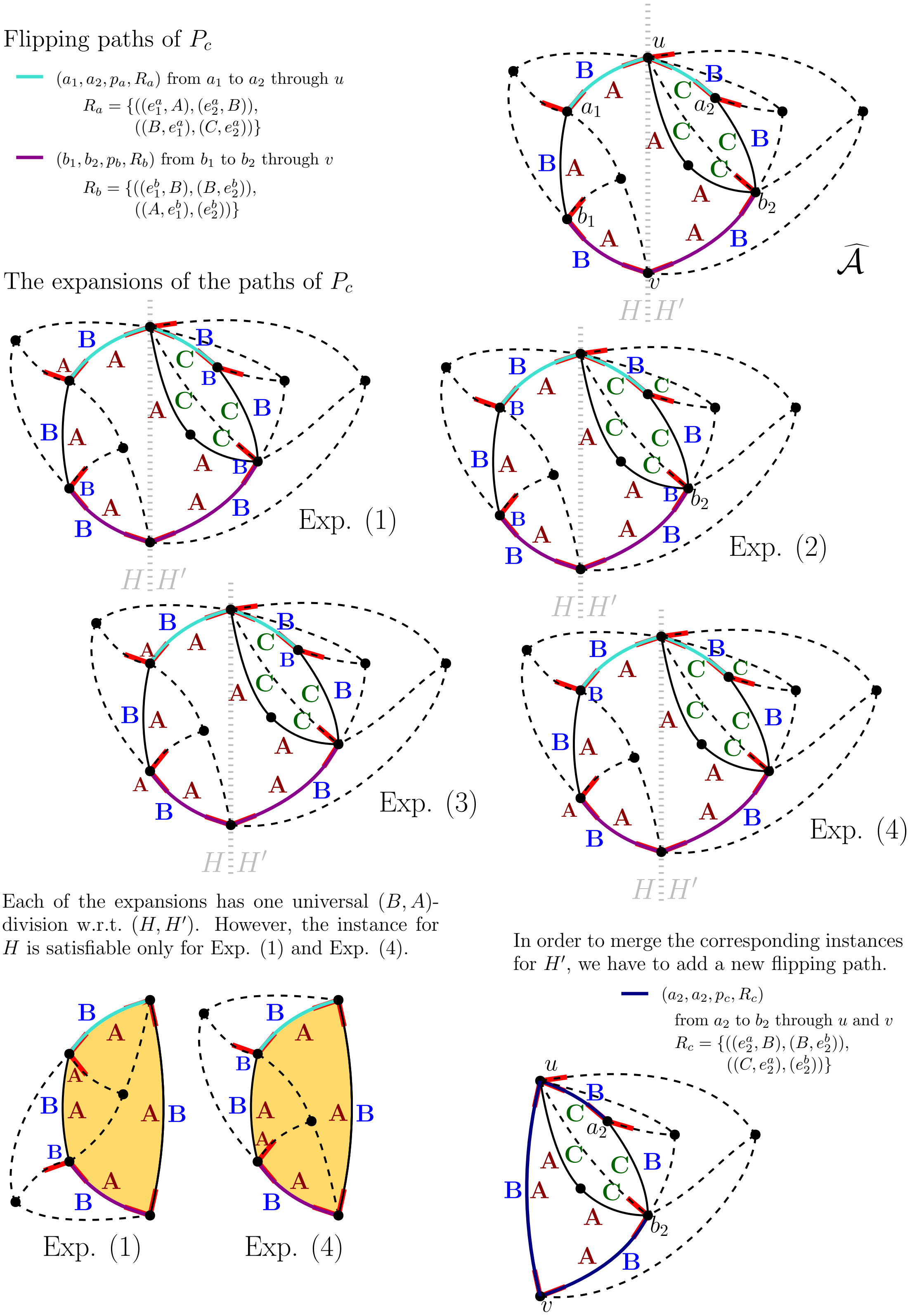


Figure 4.7: An example of the situation with two flipping paths in the set P_c.

Else if $p_u = p_v$ and the anchors are in H', then we just shortcut the path through $e_G^{virt}(H')$ without changing the expansions.

Else $p_u = p_v$ and the anchors are in H. In this case we do not add anything.

We have demonstrated how to merge the instances of T' into one. However, we still have to verify that the created instance satisfies all the requirements. Especially, we have to check that the newly added flipping paths have the required properties.

Except for $e_G^{virt}(H')$, the newly added anchors and paths recycle the edges of the old paths. Therefore, all the flipping anchors and paths of the new instance are edge-disjoint. Also, the newly added flipping path trivially satisfies the properties (i) and (ii) of the Definition 85. We just have to check the point (iii) at the vertices u and v.

We focus only on the vertex u. Let $\rho_{\widehat{\mathcal{A}}}(u) = (\tau, \lambda, \tau', \lambda')$, otherwise no new flipping path through u is added. There are two possibilities how a new path p is created. Either we extend a path of the set P' ending in u by a suspicious flipping anchor, or we reroute a path of the set P_c to v. In both cases after coming to u the path p continues by the edge $e_G^{virt}(H)$. Without loss of generality, we can assume that p arrives at u via the first edge of the sequence τ' and that there are at least two elements in τ'.

In the first case, when we extend a path with a suspicious anchor $b = (u, e, R)$, then it holds that $l_{\widehat{\mathcal{A}}}(e) = \ell_1 = r_{\widehat{\mathcal{A}}}(e_G^{virt}(H'))$. And since b is suspicious, then there is no directed path from $r_{\widehat{\mathcal{A}}}(e)$ to $\ell_1 = l_{\widehat{\mathcal{A}}}(e)$ avoiding $\ell_2 = l_{\widehat{\mathcal{A}}}(e_G^{virt}(H'))$ in the label ordering graph G_L^u of u in $\widehat{\mathcal{A}}$. The label ordering graph K_L^u of u in $\widehat{\mathcal{B}'_*}$ without the edge $e_G^{virt}(H')$ is a subgraph of G_L^u, so the condition (iii) is also satisfied.

In the other case, we reroute a path p', that comes to u from H' via an edge e. We orient the edge e from u outwards. We did not reject in the previous inspection of p', so $l_{\widehat{\mathcal{A}}}(e) = \ell_1 = r_{\widehat{\mathcal{A}}}(e_G^{virt}(H'))$. The path p' is a flipping path, so there is no directed path from $r_{\widehat{\mathcal{A}}}(e)$ to $l_{\widehat{\mathcal{A}}}(e) = \ell_1$ avoiding a label $x \in (L_E \setminus \{\ell_1\})$ in G_L^u. It means that each directed path from $r_{\widehat{\mathcal{A}}}(e)$ to ℓ_1 in G_L^u passes through x. But it implies that each directed path from $r_{\widehat{\mathcal{A}}}(e)$ to ℓ_1 also passes through ℓ_2. The previous statement is trivial for $x = \ell_2$. If $x \neq \ell_2$, then the path must visit ℓ_1 or ℓ_2 before x in order to switch to a label of the set X. But a path cannot visit ℓ_1 twice. Therefore there is no directed path from $r_{\widehat{\mathcal{A}}}(e)$ to ℓ_1 avoiding ℓ_2 in G_L^u. And since K_L^u without the edge $e_G^{virt}(H')$ is a subgraph of G_L^u, then the condition (iii) is satisfied.

4.3.4.4 No labels in the intersection

Finally, we consider the case with $L_E = \emptyset$ or $L'_E = \emptyset$. Let L_P be the set of candidates for the parental label from Lemma 81. We already know that G has connected labeled embedding satisfying $\widehat{\mathcal{A}}$ iff there exists $p \in L_P$ such that the split graphs H and H' have connected labeled embeddings satisfying the universal p-division of $\widehat{\mathcal{A}}$. However, in the A*ERCS problem, the candidates for the parental label can be also influenced by the bicolored edges.

Definition 107. *Let Z be set of bicolored edges. Then the* intesection of labels *of Z is the set*

$$IL(Z) = \begin{cases} \{\bigstar\}, & \text{if } Z = \emptyset, \\ \bigcap_{(e,x,y)\in Z}\{x, y\}, & \text{otherwise.} \end{cases}$$

If $L_E = \emptyset$, then the parental label must match at least one member of the set $IL(Q)$. Therefore, the parental label needs to be in the intersection of the sets L_P and $IL(Q)$. However, the intersection must respect the special function of the token $\bigstar$.

Definition 108. *Let A and B be sets. The* $\bigstar$-inersection *of A and B is defined as*

$$A \cap_\bigstar B = \begin{cases} B, & \text{if } A = \{\bigstar\}, \\ A, & \text{else if } B = \{\bigstar\}, \\ A \cap B, & \text{otherwise.} \end{cases}$$

Lemma 109. *Let $L_E = \emptyset$. Then, G has connected labeled embedding satisfying $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$ iff there exists $p \in (L_P \cap_\bigstar IL(Q))$ such that respectively H and H' have connected labeled embeddings satisfying $(\widehat{\mathcal{B}}, O, P, Q)$ and $(\widehat{\mathcal{B}}', O', P', Q')$ where $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ is the universal p-division of $\widehat{\mathcal{A}}$ w.r.t. (H, H').*

Proof. The proof is similar to the Lemmata 92 and 106. Let $L_I = (L_P \cap_\bigstar IL(Q))$.

Let $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ be the universal p-division of $\widehat{\mathcal{A}}$ for a label $p \in L_I$. Further, let $\mathcal{H}_L$ be a connected labeled embeddings satisfying an expansion $\widehat{\mathcal{D}}$ of $(\widehat{\mathcal{B}}, O, P, Q)$ and $\mathcal{H}'_L$ a connected labeled embedding satisfying an expansion $\widehat{\mathcal{D}}'$ of $(\widehat{\mathcal{B}}', O', P', Q')$. Let $\widehat{\mathcal{E}}$ be the expansion of $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$ where everything is expanded the same way as in $\widehat{\mathcal{D}}$ and $\widehat{\mathcal{D}}'$. $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$ is equivalent to a division of $\widehat{\mathcal{E}}$. Therefore, if $p \neq \bigstar$, then by Lemma 73 the graph G has the desired embedding. And if $p = \bigstar$, then there are no bicolored edges in Q, so we can relabel $\mathcal{H}_L$ by the label $l_{\mathcal{H}'_L}(E_G^{virt}(H'))$ and after that use Lemma 73.

For the second implication, we assume that G has a connected labeled embedding satisfying an expansion $\widehat{\mathcal{E}}$ of $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$. If $Q = \emptyset$, then by applying Lemmata 73 and 81 we get a label $p \in L_I$ such that the both restrictions of the universal p-division $(\widehat{\mathcal{D}}, \widehat{\mathcal{D}}')$ of $\widehat{\mathcal{E}}$ have satisfying connected labeled embeddings. Let $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ be the universal p-division of $\widehat{\mathcal{A}}$. Then, $\widehat{\mathcal{D}}$ is equivalent to an expansion of $(\widehat{\mathcal{B}}, O, P, Q)$ and $\widehat{\mathcal{D}}'$ to an expansion of $(\widehat{\mathcal{B}}', O', P', Q')$.

Otherwise $Q \neq \emptyset$. In this case, there is exactly one label p in the intersection of the set of labels incident to the edges of H in $\widehat{\mathcal{E}}$ and the set of labels incident to the edges of H' in $\widehat{\mathcal{E}}$. Apparently, $p \in IL(Q)$, because the edges of Q have prescribed labels in H. In addition $p \in L_I$, since the other labels cannot tag the faces incident to the edges of H in a labeled embedding of G satisfying $\widehat{\mathcal{A}}$. We finish the proof in the same way as in Lemma 92. □

Remark. A similar lemma can be proven for $L'_E = \emptyset$. We just need to replace $IL(Q)$ by $IL(Q')$ in the statement of the lemma.

Lemma 109 yields several implications about the case $L_E = \emptyset$. When the intersection $(L_P \cap_\bigstar IL(Q))$ is empty, then there is no connected labeled embedding of G satisfying $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$. And if $(L_P \cap_\bigstar IL(Q)) = \{p\}$, then we just have to solve the instances A*ERCS$(H, \widehat{\mathcal{B}}_p, O, P, Q)$ and A*ERCS$(H', \widehat{\mathcal{B}}'_p, O', P', Q')$ where $(\widehat{\mathcal{B}}_p, \widehat{\mathcal{B}}'_p)$ is the universal p-division of $\widehat{\mathcal{A}}$. Lastly, if $(L_P \cap_\bigstar IL(Q)) = \{p_1, p_2\}$, then let $(\widehat{\mathcal{B}}_1, \widehat{\mathcal{B}}'_1)$ denote the universal p_1-division of $\widehat{\mathcal{A}}$ and $(\widehat{\mathcal{B}}_2, \widehat{\mathcal{B}}'_2)$ the universal p_2-division. We first solve the instances A*ERCS$(H, \widehat{\mathcal{B}}_1, O, P, Q)$ and A*ERCS$(H, \widehat{\mathcal{B}}_2, O, P, Q)$. If both of them rejects, then we also reject. If just one of them succeeds, then we proceed to solve the corresponding instance for H'. And if both of them succeed, then merge the instances for H' like in Corollary 83 using the new bicolored edge $(e_G^{virt}(H'), p_1, p_2)$. It is sometimes necessary for the merging to replace the rotation schemes of the separation vertices by the equivalent cyclic sequence (p_1, p_2). We substitute when $\rho_{\widehat{\mathcal{B}}'_1}(w) = (e_G^{virt}(H'), p_2)$ and $\rho_{\widehat{\mathcal{B}}'_2}(w) = (e_G^{virt}(H'), p_1)$ for $w \in \{u, v\}$.

We use a different strategy when $L'_E = \emptyset$. Here we utilize the fact that we are able to solve the A*ERCS instances with at most one unique label prescribed by the functions $l_{\widehat{\mathcal{A}}}$ and $r_{\widehat{\mathcal{A}}}$ in polynomial time, even if the graph is not an [SPQR]-skeleton. Therefore, for each label $p \in (L_P \cap IL(Q'))$ we can solve the instances A*ERCS$(H, \widehat{\mathcal{B}}_p, O, P, Q)$ and A*ERCS$(H', \widehat{\mathcal{B}}'_p, O', P', Q')$ in polynomial time, where $(\widehat{\mathcal{B}}_P, \widehat{\mathcal{B}}'_p)$ is the universal p-division of $\widehat{\mathcal{A}}$.

4.3.4.5 The algorithm

Putting together our findings about the A*ERCS problem, we design Algorithm 7. Unfortunately, we do not know whether the algorithm runs in polynomial time. The issue is that we are capable of solving the A*ERCS instances for P-skeletons, only if the size of the set of bicolored edges is bounded by a constant. However, we can prove that Algorithm 7 is polynomial if we start with no bicolored edges. In this setting, all the generated A*ERCS instances for P-skeletons have at most two bicolored edges.

Lemma 110. *Let G be a biconnected graph and $\widehat{\mathcal{A}}$ an augmented embedding restriction of G with labeled opaque edges. Further, let $O_{\widehat{\mathcal{A}}}$ be a set of flipping anchors of $\widehat{\mathcal{A}}$ and $P_{\widehat{\mathcal{A}}}$ a set of flipping paths of $\widehat{\mathcal{A}}$. Then* `A*ERCS(`$G, \widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, \emptyset$`)` *runs in polynomial time with respect to the size of G.*

Proof. We say that an A*ERCS instance is trivial if its graph is an [SPQR]-skeleton or if the functions $l_{\widehat{\mathcal{A}}}$ and $r_{\widehat{\mathcal{A}}}$ prescribe at most one unique label for the incident faces. Observe that in each iteration, the function `A*ERCS` makes at most one non-trivial recursive call. Let k denote the total number of non-trivial recursive calls made by `A*ERCS(`$G, \widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, \emptyset$`)`. In each non-trivial iteration the function `A*ERCS` cuts off an [SPR]-skeleton from the SPQR-tree of G, so $k \in \mathcal{O}(|V(G)|)$. Let $(G_0, \widehat{\mathcal{A}}_0, O_0, P_0, Q_0) = (G, \widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, \emptyset)$ be the original parameters and for each $i \in \{1, \dots, k\}$ let $(G_i, \widehat{\mathcal{A}}_i, O_i, P_i, Q_i)$ be the parameters of the non-trivial call made by `A*ERCS(`$G_{i-1}, \widehat{\mathcal{A}}_{i-1}, O_{i-1}, P_{i-1}, Q_{i-1}$`)`.

We already know that except for the P-skeletons the trivial A*ERCS instances are solved in polynomial time. We show that each P-skeleton instance generated by `A*ERCS(`$G, \widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, \emptyset$`)` has at most two bicolored edges. Then, we can apply Lemma 89 to solve it also in polynomial time.

New bicolored edges are added only on the line 31. Let (e, x, y), $e = \{u_i, v_i\}$, be a bicolored edge created by `A*ERCS(`$G_i, \widehat{\mathcal{A}}_i, O_i, P_i, Q_i$`)` for $i \in \{0, \dots, k\}$. Let (H_i, H'_i) be the simple pair of split graphs used in the i-th iteration. Then for each edge $e_i \in (E(H_i) \setminus \{e_{G_i}^{virt}(H_i)\})$ it must hold that $l_{\widehat{\mathcal{A}}_i}(e_i) = r_{\widehat{\mathcal{A}}_i}(e_i) = \bigstar$.

Algorithm 7: A polynomial algorithm for A*ERCS.

input : A biconnected graph G, an augmented embedding restriction $\widehat{\mathcal{A}}$ of G with labeled opaque edges, a set of flipping anchors $O_{\widehat{\mathcal{A}}}$, a set of flipping paths $P_{\widehat{\mathcal{A}}}$, a set of bicolored edges $Q_{\widehat{\mathcal{A}}}$.

function `A*ERCS(`$G, \widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}}$`)` **:**
- **if** $\left|\{\textit{the labels prescribed by } l_{\widehat{\mathcal{A}}}, r_{\widehat{\mathcal{A}}}\}\right| \leq 1$ **:**
  - **if** $\left|\{\textit{the labels prescribed by } l_{\widehat{\mathcal{A}}}, r_{\widehat{\mathcal{A}}}, \rho_{\widehat{\mathcal{A}}}\}\right| \geq 2$ **: return** false;
  - **else if** $(IL(Q_{\widehat{\mathcal{A}}}) \cap_\bigstar \{\textit{the label of } \widehat{\mathcal{A}} \textit{ or } \bigstar\}) = \emptyset$ **: return** false;
  - **else return** `ERCS_biconnected(`G, `omit_labels(`$\widehat{\mathcal{A}}$`))`;
- **if** *G is an [SPQR]-skeleton* **:**
  - **return** `A*ERCS_skeleton(`$G, \widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}}$`)`;
- $(u,v), (H,H') \leftarrow$ a separation pair of G and a simple pair of split graphs w.r.t. $\{u,v\}$ such that H is an [SPR]-skeleton;
- **if** *(H,H') is not $\widehat{\mathcal{A}}$-non-crossing* **: return** false;
- $L_E, L'_E, X, X', O, O', P, P', P_c, Q, Q'$ as specified in Definition 91;
- $T' \leftarrow \emptyset$;
- **if** $|L_E \cap L'_E| \geq 3$ **: return** false;
- **if** $|L_E| > 0$ **and** $|L'_E| > 0$ **and** $|L_E \cap L'_E| = 0$ **: return** false;
- **if** $|L \cap L'| = \{\ell\}$ **:**
  - **if** *$\rho_{\widehat{\mathcal{A}}}(u)$ or $\rho_{\widehat{\mathcal{A}}}(v)$ is not (X,X')-non-crossing* **: return** false;
  - **if** *both H and H' are not ℓ-passable* **: return** false;
  - **if** $P_c \neq \emptyset$ **: return** false;
  - $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}') \leftarrow$ the universal ℓ-division of $\widehat{\mathcal{A}}$ w.r.t. (H,H');
  - **return** (`A*ERCS(`$H, \widehat{\mathcal{B}}, O, P, Q$`)` **and** `A*ERCS(`$H', \widehat{\mathcal{B}}', O', P', Q'$`)`);
- **if** $|L'_E| = 0$ **:**
  - **foreach** *label* $p \in (L_P \cap_\bigstar IL(Q'))$ **:**
    - $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}') \leftarrow$ the universal p-division of $\widehat{\mathcal{A}}$ w.r.t. (H,H');
    - **if** `A*ERCS(`$H, \widehat{\mathcal{B}}, O, P, Q$`)` **and** `A*ERCS(`$H', \widehat{\mathcal{B}}', O', P', Q'$`)` **:**
      - **return** true;
  - **return** false;
- **if** $|L_E| = 0$ **:**
  - **foreach** *label* $p \in (L_P \cap_\bigstar IL(Q))$ **:**
    - $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}') \leftarrow$ the universal p-division of $\widehat{\mathcal{A}}$ w.r.t. (H,H');
    - **if** `A*ERCS(`$H, \widehat{\mathcal{B}}, O, P, Q$`)` **:** Add `A*ERCS(`$H', \widehat{\mathcal{B}}', O', P', Q'$`)` to T' ;
  - **if** $T' = \emptyset$ **: return** false;
  - **return** `A*ERCS(`*the merge of the instances of T'*`)`;
- $\{\ell_1, \ell_2\} \leftarrow L_E \cap L'_E$;
- **if** *$\rho_{\widehat{\mathcal{A}}}(u)$ or $\rho_{\widehat{\mathcal{A}}}(v)$ is not (X,X')-non-crossing* **: return** false;
- $\widehat{\mathcal{R}} \leftarrow$ `omit_labels_in_rotation_schemes(`$\widehat{\mathcal{A}}$`)`;
- **if** `local_Euler_test(`$H, \widehat{\mathcal{R}}[H, \ell_1, \ell_2], u, \ell_1$`)` **:** $(x,y) \leftarrow (\ell_1, \ell_2)$;
- **else** $(x,y) \leftarrow (\ell_2, \ell_1)$;
- regularize the flipping anchors and the flipping paths;
- remove elements with suspicious flipping anchors from O, O', P and P';
- **if** *there are more than 2 complications at u or at v* **: return** false;
- **foreach** *expansion $\widehat{\mathcal{E}}$ of just the elements with suspicious flipping anchors and the paths of P_c in $(\widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, Q_{\widehat{\mathcal{A}}})$* **:**
  - **foreach** *universal (x,y)-division $(\widehat{\mathcal{B}}, \widehat{\mathcal{B}}')$ of $\widehat{\mathcal{E}}$ w.r.t. (H,H')* **:**
    - **if** `A*ERCS(`$H, \widehat{\mathcal{B}}, O, P, Q$`)` **:** Add `A*ERCS(`$H', \widehat{\mathcal{B}}', O', P', Q'$`)` to T' ;
- **return** `A*ERCS(`*the merge of the instances of T'*`)`;

There are two reasons why the bicolored edge (e, x, y) is produced. The first possibility is that $L_P = \{x, y\}$, $Q = \emptyset$ and there is $w \in \{u_i, v_i\}$ such that $\rho_{\widehat{\mathcal{A}}_i}(w) = (\varepsilon, x, \varepsilon', y)$, where ε, ε' are non-empty sequences of edges of H_i. In this case (e, x, y) is really a new bicolored edge and we say that w is a *key* vertex of (e, x, y). It holds that $\rho_{\widehat{\mathcal{A}}_{i+1}}(w) = (x, y)$ and for each iteration $j > i$ the rotation scheme $\rho_{\widehat{\mathcal{A}}_j}(w)$ contains at most two elements.

The other option is that $IL(Q) = \{x, y\}$ and either $L_P = \{\bigstar\}$ or $L_P = \{x, y\}$. It means that there is already a bicolored edge in H_i which just changes its position. Therefore, there must be an iteration $j < i$ when a new bicolored edge (e_j, x, y), $e_j = \{u_j, v_j\}$, is introduced in the part of the graph represented by the split graph H_i. Let $w_j \in \{u_j, v_j\}$ be the key vertex of (e_j, x, y). It holds that $\rho_{\widehat{\mathcal{A}}_{j+1}}(w) = (x, y)$. The vertex w_j must be either u_i or v_i. Otherwise, the labels x, y in $\rho_{\widehat{\mathcal{A}}_{j+1}}(w)$ cannot disappear until the i-th iteration and they would cause that $L_P = \emptyset$. So again, there is a vertex $w \in \{u_i, v_i\}$ and an iteration $j' \leq i$ such that $\rho_{\widehat{\mathcal{A}}_{i'}}(w)$ has at least 4 elements and for each iteration $i' > j'$ the rotation scheme $\rho_{\widehat{\mathcal{A}}_{i'}}(w)$ contains at most two elements. We say that w is a *key* vertex of (e, x, y).

Hence, each bicolored edge has a key vertex where the edge originally appeared. Later, the edge might have rotated around the key vertex to its current position. Each vertex is a key vertex for at most one bicolored edge because after the creation of the first bicolored edge the rotation scheme is too short to produce another one. Thus, each P-skeleton has at most two bicolored edges, since it has only two vertices.

All the trivial A*ERCS instances generated by `A*ERCS`$(G, \widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, \emptyset)$ are solved in polynomial time. Moreover, each non-trivial iteration of `A*ERCS` except for the recursive calls also runs in polynomial time. Actually, using some clever data structures the i-th iteration of `A*ERCS` can be implemented in time $\mathcal{O}(|V(G_i)| + |E(G_i)|)$. The required simple pair of split graphs (H, H') can be found using the SPQR- tree of G_i.

There are $\mathcal{O}(|V(G)|)$ non-trivial iterations, therefore the total time complexity of `A*ERCS`$(G, \widehat{\mathcal{A}}, O_{\widehat{\mathcal{A}}}, P_{\widehat{\mathcal{A}}}, \emptyset)$ is polynomial with respect to the size of G. More precisely, there is an implementation running in $\mathcal{O}(|E(G)|^6)$. □

The A*ERCS problem is just a generalization of the AERCS problem for biconnected graphs and augmented embedding restrictions with labeled opaque edges. Therefore, we can use Algorithm 7 to solve such AERCS instances. We just substitute the empty set for the sets of flipping anchors, flipping paths and bicolored edges.

Theorem 111. *Let G be a biconnected graph and let $\widehat{\mathcal{A}}$ be an augmented embedding restriction of G with labeled opaque edges. Then AERCS$(G, \widehat{\mathcal{A}})$ can be solved in polynomial time with respect to the size of G.*

4.4 Connected graphs

In this section, we derive a polynomial algorithm for the ERCS instances with connected graphs and embedding restrictions with labeled opaque edges. The algorithm is a bit more general. It can even solve the AERCS problem. We have

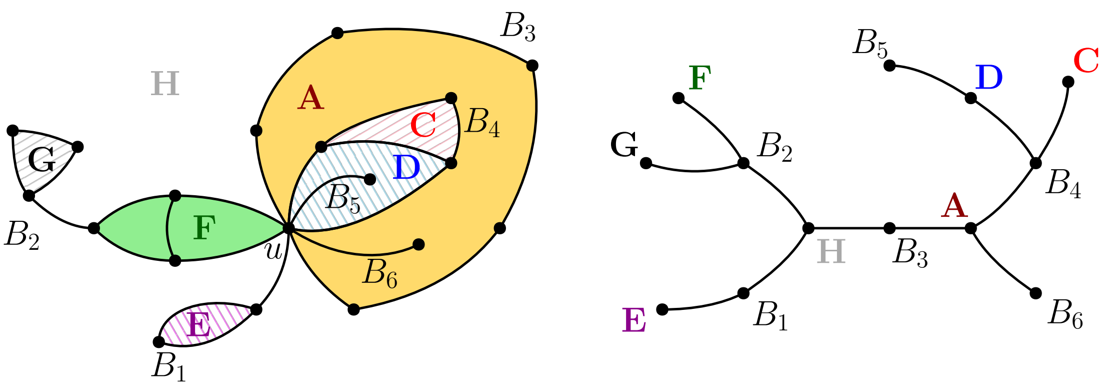


Figure 4.8: A connected labeled embedding of a connected graph with a cut vertex u and its u-block-label tree.

already derived the functions `ERCS_biconnected` and `A*ERCS` that deal with the biconnected graphs. So now we look at the instances where the input graph has a cut vertex.

Let G be a connected graph with a cut vertex $u \in V(G)$ and let $\widehat{\mathcal{A}}$ be an augmented embedding restriction of G with labeled opaque edges. For simplicity, we assume that there are no labels in the rotation scheme $\rho_{\widehat{\mathcal{A}}}(u)$. If there are some, then we can replace each such label by a new edge e incident to u and prescribe the label for the incident faces of e.

We decompose G into smaller blocks that share only the vertex u. Our goal is to reduce the instance AERCS$(G, \widehat{\mathcal{A}})$ into a set of independent AERCS instances for these blocks such that G has a connected labeled embedding satisfying $\widehat{\mathcal{A}}$ iff all of the block instances accept. We must further ensure that the satisfying embeddings of the block instances can be put back together around the vertex u.

Definition 112. *Let $C_1, \ldots, C_k$ be the components of the graph $G - u$. For each $i \in \{1, \ldots, k\}$ the subgraph of G induced by the set of vertices $(V(C_i) \cup \{u\})$ is called a u-*block *of G.*

Notice that for each embedding $\mathcal{G}$ of G and every pair of u-blocks B_1, B_2 there is at most one face of $\mathcal{G}$ that is incident to both B_1 and B_2. It means that in each connected labeled embedding of G the blocks B_1 and B_2 share at most one label. Actually, the u-blocks and the labels of a connected labeled embedding form a tree structure.

Definition 113. *Let $(\mathcal{G}, g)$ be a connected labeled embedding of G. The u-*block-label tree *of (G, g) is the bipartite graph where the first part is the set of the u-blocks of G, the second part is the set of the labels of $(\mathcal{G}, g)$, and a u-block B and a label ℓ are connected by an edge iff there exists an edge $e \in E(B)$ such that ℓ is incident to e (formally $g(l_{(\mathcal{G},g)}(e)) = \ell$ or $g(r_{(\mathcal{G},g)}(e)) = \ell$).*

Lemma 114. *The u-block-label tree of connected labeled embedding $(\mathcal{G}, g)$ is a tree, i.e. it is connected and acyclic.*

Proof. Let G^* be the dual planar graph to the embedding $\mathcal{G}$. Observe that there is a natural bijection between the edges of G and the edges of G^*. Then, a label ℓ and a u-block B are connected in the u-block-label tree iff there is an edge

$e \in E(G)$ of the block B such that its corresponding edge in G^* is incident to the vertex representing ℓ. And since the dual planar graphs are always connected, then the u-block-label tree of $(\mathcal{G}, g)$ must be also connected.

It remains to show that the u-block-label tree is acyclic. Apparently, it holds for the cases where $(\mathcal{G}, g)$ contains only one label. Otherwise, let ℓ_1, ℓ_2 be two distinct labels of $(\mathcal{G}, g)$. Then, there must be a cycle C in $(\mathcal{G}, g)$ separating the faces tagged by ℓ_1 and by ℓ_2. A cycle is a biconnected graph, therefore there exists a u-block B such that C is a subgraph of B. Except for B, each u-block B' of G is either inside or outside C in $\mathcal{G}$, so every path connecting ℓ_1 and ℓ_2 in the u-block-label tree of $(\mathcal{G}, g)$ must pass through B. Thus, there is no cycle in the u-block-label tree containing both ℓ_1 and ℓ_2. □

Analogously, we can construct the u-block-label graph for an augmented embedding restriction.

Definition 115. *The* u-block-label graph *of the augmented embedding restriction $\widehat{\mathcal{A}}$ is the bipartite graph where the first part is the set if the u-blocks of G, the second part is the set of the labels appearing $\widehat{\mathcal{A}}$, and a u-block B and a label ℓ are connected by an edge iff either there exists an edge $e \in E(B)$ such that $l_{\widehat{\mathcal{A}}}(e) = \ell$ or $r_{\widehat{\mathcal{A}}}(e) = \ell$, or there is a vertex $w \in (V(B) \setminus u)$ such that ℓ appears in the rotation scheme $\rho_{\widehat{\mathcal{A}}}(w)$.*

Let $BL_{\widehat{\mathcal{A}}}$ be the u-block-label graph of $\widehat{\mathcal{A}}$. $BL_{\widehat{\mathcal{A}}}$ does not have to be a tree, but it is still close to one if AERCS$(G, \widehat{\mathcal{A}})$ is satisfiable.

Lemma 116. *If G has a connected labeled embedding satisfying $\widehat{\mathcal{A}}$, then at most one of the components of $BL_{\widehat{\mathcal{A}}}$ is a tree and the remaining components are isolated vertices representing u-blocks.*

Proof. Let $\mathcal{G}_L$ be a connected labeled embedding of G satisfying $\widehat{\mathcal{A}}$ and let $BL_{\mathcal{G}_L}$ be the u-block-label tree of $\mathcal{G}_L$. Apparently, $BL_{\widehat{\mathcal{A}}}$ must be a subgraph of $BL_{\mathcal{G}_L}$. It means that $BL_{\widehat{\mathcal{A}}}$ is acyclic.

Further, we utilize the fact that $\widehat{\mathcal{A}}$ has labeled opaque edges. If there is a u-block B that is adjacent to at least two different labels in $BL_{\mathcal{G}_L}$, then B must neighbor with the same labels in $BL_{\widehat{\mathcal{A}}}$. Therefore, all the u-blocks, that are not leaves of $BL_{\mathcal{G}_L}$, are adjacent to the same labels in $BL_{\widehat{\mathcal{A}}}$ and $BL_{\mathcal{G}_L}$. These u-blocks and the labels of $\widehat{\mathcal{A}}$ form the tree component of $BL_{\widehat{\mathcal{A}}}$. The remaining vertices are the leaves of $BL_{\mathcal{G}_L}$ representing u-blocks. Each of them is either connected to the tree component of $BL_{\widehat{\mathcal{A}}}$ or an isolated vertex of $BL_{\widehat{\mathcal{A}}}$. □

We further assume that at most one component of $BL_{\widehat{\mathcal{A}}}$ is a tree and the remaining are isolated vertices representing u-blocks. The isolated vertices correspond to the u-blocks that must be labeled by just one label, but we do not know which one.

First, we look at the situation when $BL_{\widehat{\mathcal{A}}}$ is a tree. Let $\mathcal{G}_L$ be a connected labeled embedding of G satisfying $\widehat{\mathcal{A}}$. Thanks to the assumption, the u-block-label tree of $\mathcal{G}_L$ is identical to $BL_{\widehat{\mathcal{A}}}$.

Two adjacent edges in the rotation scheme $\rho_{\mathcal{G}_L}(u)$ share an incident face, so they must also share its label. Therefore, only the edges of the same u-block and the edges of the u-blocks with a common neighbor in $BL_{\widehat{\mathcal{A}}}$ can be adjacent in

$\rho_{\mathcal{G}_L}(u)$. So if there are two edges e_1 and e_2 next to each other in the rotation scheme $\rho_{\widehat{\mathcal{A}}}(u)$, then in order to traverse from the block of the edge e_1 to the block of e_2 we must go through all the blocks on the path connecting them in $BL_{\widehat{\mathcal{A}}}$.

Definition 117. *The* block rotation scheme *of u is a cyclic sequence $\beta_{\widehat{\mathcal{A}}}(u)$ of u-blocks derived from $\rho_{\widehat{\mathcal{A}}}(u)$ in three steps:*

(i) We take $\rho_{\widehat{\mathcal{A}}}(u)$ and replace each edge e by the u-block B for which $e \in E(B)$.

(ii) For each pair of adjacent blocks B_1, B_2 we insert between B_1 and B_2 the u-blocks on the path from B_1 to B_2 in $BL_{\widehat{\mathcal{A}}}$.

(iii) We delete consecutive duplicates. While there are two consecutive occurrences of the same u-block, we remove one of them.

If we replace each edge in $\rho_{\mathcal{G}_L}(u)$ by its u-block, then this cyclic sequence of blocks must contain $\beta_{\widehat{\mathcal{A}}}(u)$ as a subsequence.

Lemma 118. *If there are two u-blocks B_1, B_2 such that $\beta_{\widehat{\mathcal{R}}}(u)$ is (B_1, B_2)-crossing, then there is no connected labeled embedding of G satisfying $\widehat{\mathcal{R}}$.*

Proof. A presence of a u-block B in $\beta_{\widehat{\mathcal{R}}}(u)$ implies that each connected labeled embedding of G satisfying $\widehat{\mathcal{R}}$ must have an edge of $E(B)$ in this place. Either there is an edge directly in the rotation scheme $\sigma_{\widehat{\mathcal{R}}}(u)$, or an edge of B is needed to traverse between two other blocks with edges in $\sigma_{\widehat{\mathcal{R}}}(u)$. So if $\beta_{\widehat{\mathcal{R}}}(u)$ is (B_1, B_2)-crossing, then in each connected labeled embedding $\mathcal{G}_L$ satisfying $\widehat{\mathcal{R}}$ there must be edges $e_1, e_1' \in E(B_1)$ and $e_2, e_2' \in E(B_2)$ such that (e_1, e_2, e_1', e_2') is a subsequence of $\sigma_{\widehat{\mathcal{R}}}(u)$. But there is a path connecting the other endpoints of e_1 and e_1' avoiding u, and similarly there is a path connecting the other endpoints of e_2 and e_2' avoiding u. And this is in a contradiction with the planarity of $\mathcal{G}_L$. □

Definition 119. *Let T be a tree and $v \in V(T)$ its vertex. We say that v is a* twig *of T if it has in T exactly one neighbor which is not a leaf.*

Next, we define the augmented embedding restrictions for the AERCS instances for the u-blocks. We reuse the terminology and the notation from the section about ERCS of biconnected graphs.

Definition 120. *Let B be a u-block of G. An augmented embedding restriction $\widehat{\mathcal{B}} = (\rho_{\widehat{\mathcal{B}}}, l_{\widehat{\mathcal{B}}}, r_{\widehat{\mathcal{B}}}, T_{\widehat{\mathcal{B}}})$ is the* reduction *of $\widehat{\mathcal{A}}$ to B if:*

(i) $\widehat{\mathcal{B}}$ inherits the functions $l_{\widehat{\mathcal{A}}}$ and $r_{\widehat{\mathcal{A}}}$ for all the edges of B,
$(\forall e \in E(B))\ l_{\widehat{\mathcal{B}}}(e) = l_{\widehat{\mathcal{A}}}(e)\ \&\ r_{\widehat{\mathcal{B}}}(e) = r_{\widehat{\mathcal{A}}}(e)$.

(ii) $\widehat{\mathcal{B}}$ also inherits the set of transparent edges, $T_{\widehat{\mathcal{B}}} = T_{\widehat{\mathcal{A}}} \cap E(B)$.

(iii) For each vertex $w \in V(B)$ except for u the rotation scheme $\rho_{\widehat{\mathcal{B}}}(w)$ is the same as $\rho_{\widehat{\mathcal{A}}}(w)$.

(iv) If $\rho_{\widehat{\mathcal{A}}}(u)$ is not empty, then the rotation scheme $\rho_{\widehat{\mathcal{B}}}(u)$ is derived from $\rho_{\widehat{\mathcal{A}}}(u)$ by replacing each edge e that is not in $E(B)$ by the first label on the path from B to the u-block of e in the $BL_{\widehat{\mathcal{A}}}$. Plus, we remove the consecutive duplicates.

Else if $\rho_{\widehat{\mathcal{A}}}(u)$ is empty, B is a leaf of $BL_{\widehat{\mathcal{A}}}$ and there is a label ℓ adjacent to B in $BL_{\widehat{\mathcal{A}}}$, then we put $\rho_{\widehat{\mathcal{B}}}(u) = (\ell)$.

Else if $\rho_{\widehat{\mathcal{A}}}(u)$ is empty and B is a twig of $BL_{\widehat{\mathcal{A}}}$, then we put in $\rho_{\widehat{\mathcal{B}}}(u)$ just the label adjacent to B that is not a leaf of $BL_{\widehat{\mathcal{A}}}$.

Otherwise, we leave $\rho_{\widehat{\mathcal{B}}}(u)$ empty.

We use the notation $\widehat{\mathcal{A}}[B]$ for the reduction of $\widehat{\mathcal{A}}$ to B.

Lemma 121. *If $BL_{\widehat{\mathcal{A}}}$ is a tree, then the graph G has a connected labeled embedding satisfying $\widehat{\mathcal{A}}$ iff the three following conditions hold:*

(i) The block rotation scheme $\beta_{\widehat{\mathcal{A}}}(u)$ is (B_1, B_2)-non-crossing for each pair of distinct u-blocks B_1 and B_2.

(ii) Each u-block B of G has a connected labeled embedding satisfying $\widehat{\mathcal{A}}[B]$.

(iii) For each u-block B which is neither a leaf nor a twig of $BL_{\widehat{\mathcal{A}}}$, and for each label ℓ adjacent to B that is not a leaf of $BL_{\widehat{\mathcal{A}}}$, there is an edge $e \in E(B)$ such that $l_{\widehat{\mathcal{A}}}(e) = \ell$ or $r_{\widehat{\mathcal{A}}}(e) = \ell$.

Proof. Let $\mathcal{G}_L$ be a connected labeled embedding of G satisfying $\widehat{\mathcal{A}}$. Then, the condition (i) is satisfied as we have already shown in Lemma 118. For each u-block B the instance AERCS$(B, \widehat{\mathcal{A}}[B])$ is satisfied. We can just ignore all the other blocks in $\mathcal{G}_L$. The extra labels added to $\rho_{\widehat{\mathcal{A}}[B]}(u)$ are also present in the augmented rotation scheme $\rho_{\mathcal{G}_L}$, because they are the only option how to traverse to the other blocks. Similarly, the condition (iii) holds, since all the relevant labels must be incident to the shared cut vertex u. And if there are at least two different labels around u, then there also must be some edges to separate them.

For the proof of the second implication, we assume that the conditions (i), (ii) and (iii) are satisfied. We show how to put together the satisfying embeddings of the u-blocks creating a connected labeled embedding $\mathcal{G}_L$ of G satisfying $\widehat{\mathcal{A}}$. We take the satisfying embedding of one u-block and then we gradually extend it by adding the other blocks one by one.

First, we process the blocks that are present in the block rotation scheme $\beta_{\widehat{\mathcal{A}}}(u)$, if there are any. Let ψ be the elimination ordering of $\beta_{\widehat{\mathcal{A}}}(u)$ (Definition 5). We start with the satisfying embedding $\mathcal{G}_L$ of the last block of ψ. Then we go through the sequence ψ in the reversed direction adding the satisfying embeddings of the blocks to $\mathcal{G}_L$. After that we need to process the u-blocks that are not in $\beta_{\widehat{\mathcal{A}}}(u)$. We do the depth-first search of the unprocessed part of $BL_{\widehat{\mathcal{A}}}$ while adding the satisfying embeddings of the newly visited u-blocks to $\mathcal{G}_L$. If $\beta_{\widehat{\mathcal{A}}}(u)$ is not empty, then the DFS starts from the set of blocks already in $\mathcal{G}_{\mathcal{L}}$. (The blocks in $\beta_{\widehat{\mathcal{A}}}(u)$ and their adjacent labels induce a connected subgraph of the tree $BL_{\widehat{\mathcal{A}}}$.) Otherwise, we start from an arbitrary block of G.

This approach guarantees that when we are adding the satisfying embedding $\mathcal{B}_L$ of a u-block B, then $\mathcal{B}_L$ shares a common label with $\mathcal{G}_L$. Furthermore, according to Lemma 6 we can just insert the edges of B into the rotation scheme of u as one interval even for the blocks in $\beta_{\widehat{\mathcal{A}}}(u)$. So in order to join the labeled embeddings $\mathcal{G}_L$ and $\mathcal{B}_L$, we just have to select positions (corners of faces) in the augmented rotation schemes $\rho_{\mathcal{G}_L}(u)$ and $\rho_{\mathcal{B}_L}(u)$ that are tagged by the common label. Then we insert $\rho_{\mathcal{B}_L}(u)$ at the chosen position in $\rho_{\mathcal{B}_L}(u)$ in such a way that from the point of view of B we put $\rho_{\mathcal{G}_L}(u)$ at the selected position in $\rho_{\mathcal{B}_L}(u)$.

It remains to determine how to chose these positions so that $\rho_{\widehat{\mathcal{A}}}(u)$ is satisfied. We utilize the block rotation scheme $\beta_{\widehat{\mathcal{A}}}(u)$. There is a unique mapping of the rotation scheme $\rho_{\widehat{\mathcal{A}}[B]}(u)$ on $\beta_{\widehat{\mathcal{A}}}(u)$. The edges in $\rho_{\widehat{\mathcal{A}}[B]}(u)$ are projected to the corresponding occurrence of B in $\beta_{\widehat{\mathcal{A}}}(u)$ and each label ℓ in $\rho_{\widehat{\mathcal{A}}[B]}(u)$ represents a maximal interval in $\beta_{\widehat{\mathcal{A}}}(u)$ not containing B. Moreover, in the tree $BL_{\widehat{\mathcal{A}}}$, the path from B to each of the blocks represented by ℓ leaves B through the vertex ℓ. So if B is in $\beta_{\widehat{\mathcal{A}}}(u)$, then we select the position of $\rho_{\mathcal{B}_L}(u)$ that matches the label of $\rho_{\widehat{\mathcal{A}}[B]}(u)$ representing the interval of $\beta_{\widehat{\mathcal{A}}}(u)$ containing the blocks already in $\mathcal{G}_L$. Similarly for $\mathcal{G}_L$ we maintain a mapping of the augmented rotation scheme $\rho_{\mathcal{G}_L}(u)$ on $\beta_{\widehat{\mathcal{A}}}(u)$. So there is a label in $\rho_{\mathcal{G}_L}(u)$ corresponding to the interval of $\beta_{\widehat{\mathcal{A}}}(u)$ containing B. This is the required position for the insertion of $\rho_{\mathcal{B}_L}(u)$.

If B is not in $\beta_{\widehat{\mathcal{A}}}(u)$, then we can select arbitrary positions in $\rho_{\mathcal{B}_L}(u)$ and $\rho_{\mathcal{G}_L}(u)$ that are tagged by the common label. The condition (iii) and the definition of the $\rho_{\widehat{\mathcal{A}}[B]}(u)$ in cases when B is a leaf or a twig ensure that there is a face incident to u with the correct label. □

If $BL_{\widehat{\mathcal{A}}}$ is not a tree, then there are some isolated vertices in $BL_{\widehat{\mathcal{A}}}$ representing u-blocks. Each isolated u-block B has only transparent edges and there are no labels prescribed for B in $\widehat{\mathcal{A}}$. It means that in every connected labeled embedding of G satisfying $\widehat{\mathcal{A}}$ all the faces incident to the edges of B must be tagged by the same label.

The isolated u-blocks may have some edges in the rotation scheme $\rho_{\widehat{\mathcal{A}}}(u)$. We first check that there are not two distinct u-block B_1 and B_2 such that $\rho_{\widehat{\mathcal{A}}}(u)$ is $(E(B_1), E(B_2))$-crossing. If there are some crossing blocks, then G has no labeled embedding satisfying $\widehat{\mathcal{A}}$ and we reject.

Lemma 122. *Let B be a u-block of G such that B is an isolated vertex in $BL_{\widehat{\mathcal{A}}}$ and either there are no edges of B in $\rho_{\widehat{\mathcal{A}}}(u)$, or the edges of B in $\rho_{\widehat{\mathcal{A}}}(u)$ form one continuous interval. Next, let G' be the graph obtained from G by removing the edges and vertices of B except for u. Further, let $\widehat{\mathcal{A}}'$ be the augmented embedding restriction derived from $\widehat{\mathcal{A}}$ by removing the vertices and edges not in G' and let $\widehat{\mathcal{B}}$ be the restriction derived from $\widehat{\mathcal{A}}$ by omitting the vertices and edges not in B. Then G has a connected labeled embedding satisfying $\widehat{\mathcal{A}}$ iff G' and B have connected labeled embeddings satisfying $\widehat{\mathcal{A}}'$ and $\widehat{\mathcal{B}}$ respectively.*

Proof. Obviously, if G has a connected labeled embedding satisfying $\widehat{\mathcal{A}}$, then we can split it to connected labeled embeddings of G' and B that satisfy $\widehat{\mathcal{A}}'$ and $\widehat{\mathcal{B}}$ respectively.

For the second implication let us assume that G' and B have connected labeled embeddings $\mathcal{G}'_L$ and $\mathcal{B}_L$ satisfying $\widehat{\mathcal{A}}'$ and $\widehat{\mathcal{B}}$ respectively. Then we can join the embeddings $\mathcal{G}'_L$ and $\mathcal{B}_L$, creating a connected labeled embedding of G satisfying $\widehat{\mathcal{A}}$. If there are no edges of B in $\rho_{\widehat{\mathcal{A}}}(u)$, then we insert the scheme $\rho_{\mathcal{B}_L}(u)$ at an arbitrary position in $\rho_{\mathcal{G}'_L}(u)$. Otherwise, the position in $\rho_{\mathcal{G}'_L}(u)$ is determined by the interval of edges of B. In both cases, we may need to relabel the embedding $\mathcal{B}_L$, so that the labeling is consistent with the face of $\mathcal{G}'_L$ where we insert $\mathcal{B}_L$. □

We can repeatedly apply Lemma 122 until there is no isolated u-block B such that B does not have any edges in $\rho_{\widehat{\mathcal{A}}}(u)$ or the edges of B form one interval in

$\rho_{\widehat{\mathcal{A}}}(u)$. After that we just have to deal with the instances where for each isolated u-block B there are at least two intervals of edges of B in $\rho_{\widehat{\mathcal{A}}}(u)$. So there is $k \geq 2$ such that $\rho_{\widehat{\mathcal{A}}}(u) = (\varepsilon_1, \varphi_1, \varepsilon_2, \varphi_2, \ldots, \varepsilon_k, \varphi_k)$, where $\varepsilon_1, \ldots, \varepsilon_k$ are non-empty sequences of edges of B and $\varphi_1, \ldots, \varphi_k$ are non-empty sequences of edges not in B. Furthermore, since $\rho_{\widehat{\mathcal{A}}}(u)$ is non-crossing for every pair of u-blocks, then for each $i \in \{1, \ldots, k\}$ there is an edge of a u-block of the tree component of $BL_{\widehat{\mathcal{A}}}$ in φ_i. If φ_i contained only edges of the isolated u-blocks, then some of them could be removed applying Lemma 122.

In each connected labeled embedding $\mathcal{G}_L$ of G satisfying $\widehat{\mathcal{A}}$ the label tagging the faces of B is a cut vertex in the u-block-label tree of $\mathcal{G}_L$. It separates the blocks that has edges in different sequences of $\{\varphi_1, \ldots, \varphi_k\}$. The label must also be a cut vertex in $BL_{\widehat{\mathcal{A}}}$, because each sequence in $\{\varphi_1, \ldots, \varphi_k\}$ contains an edge of a block in the tree component of $BL_{\widehat{\mathcal{A}}}$. Therefore we can construct the set of candidates C_B for the label of B. A label ℓ is in C_B iff ℓ is on the path between B_1 and B_2 for each pair of block B_1, B_2 of the tree component of $BL_{\widehat{\mathcal{A}}}$ such that B_1 and B_2 have edges in two different sequences of $\{\varphi_1, \ldots, \varphi_k\}$.

Only the labels of C_B can be used to tag the faces of B. So if C_B is empty, then there is no connected labeled embedding of G satisfying $\widehat{\mathcal{A}}$. And if $C_B = \{\ell\}$, we can just add B to the tree component of $BL_{\widehat{\mathcal{A}}}$ by connecting it to the label ℓ.

Notice that if $k \geq 3$, then $|C_B| \leq 1$, because in a tree there is at most one vertex that lies simultaneously on the paths from x_1 to x_2, from x_2 to x_3 and from x_3 to x_1 for three distinct vertices x_1, x_2, x_3. However, there can be multiple candidates in case of $k = 2$. Nevertheless, there is a path p in $BL_{\widehat{\mathcal{A}}}$ containing all the labels of C_B such that the endpoints of p are also in C_B. The rotation scheme $\rho_{\widehat{\mathcal{A}}}(u)$ does not contain any edges of blocks of the path p. Thus, we can choose an arbitrary candidate $c \in C_B$ and connect B to the tree component of $BL_{\widehat{\mathcal{A}}}$ via c. If there is a connected labeled embedding $\mathcal{G}_L$ of G satisfying $\widehat{\mathcal{A}}$ where B is tagged by a different label, then we can rearrange the rotation scheme $\sigma_{\mathcal{G}_L}(u)$, so that the edges of B get into a face tagged by the label c.

If there is more than one isolated u-block, then it is necessary to process them one by one. The selection of the label to tag B can affect the set of candidates for the other isolated blocks.

Theorem 123. *Let G be a connected graph and let $\widehat{\mathcal{A}}$ be an augmented embedding restriction of G with labeled opaque edges. Then AERCS$(G, \widehat{\mathcal{A}})$ can be solved in polynomial time with respect to the size of G.*

Proof. The function `AERCS_connected`$(G, \widehat{\mathcal{A}})$ accepts iff G has a connected labeled embedding satisfying $\widehat{\mathcal{A}}$. The algorithm first tries to create an equivalent AERCS instance satisfying the assumptions of Lemma 121 and then it verifies the three conditions of the lemma.

Except for the recursive calls, the function `AERCS_connected` runs in polynomial time with respect to the size of G. The algorithm makes at most one recursive call for each u-block. The u-blocks have together one fewer cut vertices than G. Thus, in total at most $|V(G)|$ recursive calls are made. So the total run time is also polynomial. There is an implementation running in time $\mathcal{O}(|E(G)|^6)$. □

Theorem 124. *Let G be a connected graph and let $\widehat{\mathcal{A}}$ be an augmented embedding restriction of G with labeled opaque edges and anchored borders. Then*

Algorithm 8: A polynomial algorithm solving AERCS for connected graphs.

input : A connected graph G, an augmented embedding restriction $\widehat{\mathcal{A}}$ with labeled opaque edges.

function `AERCS_connected`($G, \widehat{\mathcal{A}}$):
- **if** *G is biconnected* **: return** `A*ERCS`($G, \widehat{\mathcal{A}}, \emptyset, \emptyset, \emptyset$) ;
- $u \leftarrow$ a cut vertex of G;
- **if** *there are labels in $\rho_{\widehat{\mathcal{A}}}(u)$* **:**
  - edit G and $\widehat{\mathcal{A}}$ replacing the labels of $\rho_{\widehat{\mathcal{A}}}(u)$ by new edges;
- $BL_{\widehat{\mathcal{A}}} \leftarrow$ the u-block-label graph of $\widehat{\mathcal{A}}$;
- **if** *$BL_{\widehat{\mathcal{A}}}$ is not acyclic or it has at least 2 components that are not isolated u-blocks* **: return** false ;
- **if** *$\rho_{\widehat{\mathcal{A}}}(u)$ is $(E(B_1), E(B_2))$-crossing for some blocks B_1, B_2* **: return** false ;
- **while** *there is an isolated u-block B such that B has no edges in $\rho_{\widehat{\mathcal{A}}}(u)$ or the edges of B form one continuous interval in $\rho_{\widehat{\mathcal{A}}}(u)$* **do**
  - $\widehat{\mathcal{B}} \leftarrow$ the augmented embedding restriction derived from $\widehat{\mathcal{A}}$ by removing the edges and vertices not in B;
  - **if not** `AERCS_connected`($B, \widehat{\mathcal{B}}$) **: return** false ;
  - remove the edges and vertices of B except for u from G and $\widehat{\mathcal{A}}$;
- **while** *there is an isolated u-block B* **do**
  - $C_B \leftarrow$ the set of candidates for the label of B;
  - **if** $C_B = \emptyset$ **: return** false;
  - select an arbitrary $c \in C_B$ and connect B to c in $BL_{\widehat{\mathcal{A}}}$;
- $\beta_{\widehat{\mathcal{A}}}(u) \leftarrow$ the block rotation scheme of u;
- **if** *$\beta_{\widehat{\mathcal{A}}}(u)$ is (B_1, B_2)-crossing for some u-blocks B_1, B_2* **: return** false ;
- **foreach** *u-block B of G that is not a leaf or a twig of $BL_{\widehat{\mathcal{A}}}$* **do**
  - **foreach** *label ℓ adjacent to B in $BL_{\widehat{\mathcal{A}}}$ that is not a leaf* **do**
    - **if** *there is no edge $e \in E(B)$ incident to u such that $l_{\widehat{\mathcal{A}}}(e) = \ell$ or $r_{\widehat{\mathcal{A}}}(e) = \ell$* **:**
      - **return** false;
- **foreach** *u-block B of G* **do**
  - **if not** `AERCS_connected`($B, \widehat{\mathcal{A}}[B]$) **: return** false ;
- **return** true;

AERCS($G, \widehat{\mathcal{A}}$) can be solved in linear time with respect to the size of G.

Proof. (sketch) In order to achieve linear time, we must cleverly select the cut-vertex u. We construct the block-cut tree of G. It is a bipartite graph where the first part is the cut-vertices of G and the second part are the maximal biconnected subgraphs of G, which are called blocks. A cut-vertex w is linked with a block B iff w is a vertex of B. The block-cut tree of G can be constructed in time $\mathcal{O}(|V(G)| + |E(G)|)$ [9].

We choose a cut-vertex u that is a twig of the block-cut tree, so at most one u-block B is not a leaf of the block-cut tree. This allows us to construct the important part of the u-block-label graph $BL_{\widehat{\mathcal{A}}}$ in time linear with the sum of the sizes of the u-blocks excluding the u-block B. We do not need to include the labels that are incident only to B in $BL_{\widehat{\mathcal{A}}}$. (We utilize the same technique counting the occurrences of each label as we used to construct the intersection

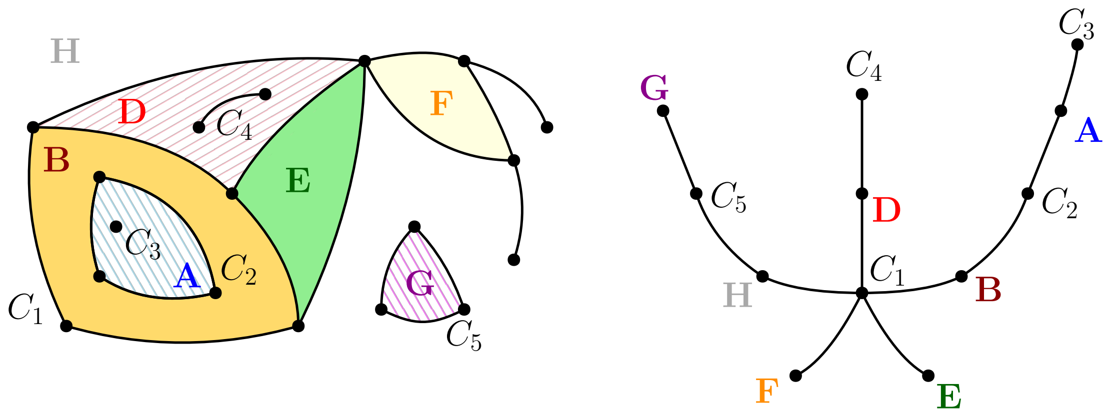


Figure 4.9: A connected labeled embedding of a disconnected graph and its component-label tree.

$L \cap L'$ in ERCS for biconnected graphs.)

The anchored border edges determine the distribution of labels around the cut vertex u. It means that we can find labels for u-blocks that are isolated in $BL_{\widehat{\mathcal{A}}}$ and have an edge in $\rho_{\widehat{\mathcal{A}}}(u)$ in time $\mathcal{O}(length(\rho_{\widehat{\mathcal{A}}}(u)))$. So in total the function `AERCS_connected`$(G, \widehat{\mathcal{A}})$ runs in time $\mathcal{O}(|V(G)| + |E(G)|)$. □

4.5 Disconnected graphs

The last step is to generalize the polynomial algorithm for the disconnected graphs. The idea is very similar to the connected case. Only this time we deal with the components of connectivity instead of the blocks.

We present an algorithm for the ERCS problem, but it can be trivially modified to solve also the AERCS instances. Let G be a graph and $\widehat{\mathcal{R}}$ an embedding restriction of G with labeled opaque edges. We define an analogy to the block-label tree and the block-label graph.

Definition 125. *Let $(\mathcal{G}, g)$ be a connected labeled embedding of G. Then, the* component-label tree *of (G, g) is the bipartite graph where the first part is the set of the components of G, the second part is the set of the labels of $(\mathcal{G}, g)$, and a component C and a label ℓ are connected by an edge iff there exists an edge $e \in E(C)$ such that ℓ is incident to e (formally $g(l_{(\mathcal{G},g)}(e)) = \ell$ or $g(r_{(\mathcal{G},g)}(e)) = \ell$).*

Definition 126. *The* component-label graph *of the embedding restriction $\widehat{\mathcal{R}}$ is the bipartite graph where the first part is the set of the components of G, the second part is the set of the labels appearing $\widehat{\mathcal{R}}$, and a component C and a label ℓ are connected by an edge iff there exists an edge $e \in E(C)$ such that $l_{\widehat{\mathcal{R}}}(e) = \ell$ or $r_{\widehat{\mathcal{R}}}(e) = \ell$.*

Let $CL_{\widehat{\mathcal{R}}}$ be the component-label graph of $\widehat{\mathcal{R}}$. The component-label tree and graph have the same property as their block-label alternatives.

Lemma 127. *The component-label tree of connected labeled embedding $(\mathcal{G}, g)$ is a tree, i.e. it is connected and acyclic.*

Lemma 128. *If G has a connected labeled embedding satisfying $\widehat{\mathcal{R}}$, then at most one of the components of $CL_{\widehat{\mathcal{R}}}$ is a tree and the remaining components are isolated vertices.*

Definition 129. *Let C be a component of G. Then the* reduction *of $\widehat{\mathcal{R}}$ to C is the embedding restriction $\widehat{\mathcal{R}}[C]$ of C that is derived from $\widehat{\mathcal{R}}$ by removing conditions for vertices and edges not in C.*

Lemma 130. *The graph G has a connected labeled embedding satisfying $\widehat{\mathcal{R}}$ iff each component C of G has a connected labeled embedding satisfying $\widehat{\mathcal{R}}[C]$ and the components of $CL_{\widehat{\mathcal{R}}}$ consists of isolated vertices and at most one tree.*

Proof. If G has a connected labeled embedding $\mathcal{G}_L$ satisfying $\widehat{\mathcal{R}}$, then for each component C we can restrict $\mathcal{G}_L$ to C getting a connected labeled embedding of C satisfying $\widehat{\mathcal{R}}[C]$. The condition about the components of $CL_{\widehat{\mathcal{R}}}$ follows from Lemma 128.

If each component has a satisfying embedding and the components of $CL_{\widehat{\mathcal{R}}}$ are isolated vertices and at most one tree, then we can put together the embeddings of the components creating a connected labeled embedding of G satisfying $\widehat{\mathcal{R}}$. We start with a satisfying embedding of a component C of the tree component of $CL_{\widehat{\mathcal{R}}}$. Then, we do the depth-first search of $CL_{\widehat{\mathcal{R}}}$ placing the newly visited components into an arbitrary face of the constructed embedding that is tagged by the shared label. The components that are isolated in $CL_{\widehat{\mathcal{R}}}$ can be placed to any face of the embedding. □

Algorithm 9: A polynomial algorithm solving the ERCS problem.

input : A connected graph G, an embedding restriction $\widehat{\mathcal{R}}$ with labeled opaque edges.

```
function ERCS(G, R̂):
    if G is connected : return ERCS_connected(G, R̂) ;
    CL_R̂ ← the component-label graph of R̂;
    if CL_R̂ is not acyclic or it has at least 2 components that are not isolated vertices
        : return false ;
    foreach component C of G do
        if not ERCS_connected(C, R̂[C]) : return false ;
    return true;
```

Theorem 131. *Let G be a connected graph and let $\widehat{\mathcal{R}}$ be an embedding restriction of G with labeled opaque edges. Then ERCS$(G, \widehat{\mathcal{R}})$ can be solved in polynomial time with respect to the size of G.*

Proof. The function `ERCS` just verifies the conditions of Lemma 130. Therefore `ERCS(`$G, \widehat{\mathcal{R}}$`)` accepts iff G has a connected labeled embedding satisfying $\widehat{\mathcal{R}}$.

Except for the calls of the function `ERCS_connected` the function `ERCS(`$G, \widehat{\mathcal{R}}$`)` runs in time $\mathcal{O}(|V(G)|+|E(G)|)$ and the function `ERCS_connected` has polynomial time complexity. So in total `ERCS(`$G, \widehat{\mathcal{R}}$`)` runs in polynomial time with respect to the size of G. There is an implementation with time complexity $\mathcal{O}(|V(G)| + |E(G)|^6)$. □

Theorem 132. *Let G be a connected graph and let $\widehat{\mathcal{R}}$ be an embedding restriction of G with labeled opaque edges and anchored borders. Then ERCS$(G, \widehat{\mathcal{R}})$ can be solved in linear time with respect to the size of G.*

5. Conclusion

We have shown that the problem of Embedding Restriction Satisfiability (ERS) is NP-complete. It remains NP-hard even for the instances with labeled opaque edges or for the instances with anchored borders. On the other hand, we have observed that ERS for [SQR]-skeletons can be solved in linear time.

We also investigated the complexity of Embedding Restriction Continuous Satisfiability (ERCS). In this variation of the ERS problem, we require that for each label ℓ the faces of the satisfying labeled embedding tagged by ℓ form a connected region. ERCS is again NP-complete even for the instances with anchored borders. However, the instances with label opaque edges can be solved in polynomial time. We proposed an algorithm running in time $\mathcal{O}(|V(G)| + |E(G)|^6)$ for a graph G. In addition, there is a linear algorithm for ERCS with labeled opaque edges and anchored borders.

The time complexity $\mathcal{O}(|V(G)|+|E(G)|^6)$ is so big that it does not make sense to implement the algorithm in practice. But, there is probably an asymptotically faster algorithm. Ideally, we would like to find a linear one. The P-skeletons would be a good place to start because they are the bottleneck of the entire procedure.

The gap between the polynomial and NP-complete variants of the problem is still relatively wide. For example, we do not know the complexity of ERS with labeled opaque edges and biconnected graphs. Also, the constraint of labeled opaque edges seems unnecessarily strong. A weaker condition may be sufficient for a polynomial algorithm for ERCS. So, there is still a lot of space for further research.

Another possible approach would be to look for minimal unsatisfiable instances. Something similar was done for the problem of Partially Embedded Planarity by Jelínek et al. [12]. These minimal structures could be used as certificates proving that an ERS instance has no satisfying labeled embedding. None of the presented polynomial algorithms provides such negative certificates. They can only construct a satisfying embedding that serves as a positive one.

Bibliography

[1] P. Angelini, G. Di Battista, F. Frati, V. Jelínek, J. Kratochvíl, M. Patrignani, and I. Rutter. Testing Planarity of Partially Embedded Graphs. *ACM Transactions on Algorithms*, 11(4):1–42, 2015.

[2] J. Boyer and W. Myrvold. On the cutting edge: Simplified O(n) planarity by edge addition. *Journal of Graph Algorithms and Applications*, 8(3):241–273, 2004.

[3] M. Chrobak and T. H. Payne. A linear-time algorithm for drawing a planar graph on a grid. *Information Processing Letters*, 54(4):241–246, 1995.

[4] G. Di Battista and R. Tamassia. Incremental Planarity Testing. *Proceedings 30th IEEE Symposium on Foundations of Computer Science*, pages 436–441, 1989.

[5] Q.-W. Feng, R. F. Cohen, and P. Eades. Planarity for Clustered Graphs. In *Algorithms - ESA '95, Third Annual European Symposium, Corfu, Greece, September 25-27, 1995, Proceedings*, pages 213–226, 1995.

[6] H. Fleischner. The uniquely embeddable planar graphs. *Discrete Mathematics*, 4(4):347–358, 1973.

[7] Z. Galil and N. Megiddo. Cyclic Ordering is NP-Complete. *Theoretical Computer Science*, 5(2):179–182, 1977.

[8] C. Gutwenger and P. Mutzel. A Linear Time Implementation of SPQR-Trees. *Graph Drawing, volume 1984 of LNCS*, pages 77–90, 2000.

[9] J. Hopcroft and R. Tarjan. Algorithm 447: efficient algorithms for graph manipulation. *Communications of the ACM*, 16(6):372–378, 1973.

[10] J. Hopcroft and R. Tarjan. Efficient planarity testing. *Journal of the ACM*, 21(4):549–568, 1974.

[11] M. Hora. An algorithm for extending partial planar drawings. Bachelor's thesis, Charles University, 2017.

[12] V. Jelínek, J. Kratochvíl, and I. Rutter. A Kuratowski–Type Theorem for Planarity of Partially Embedded Graphs. *Computational Geometry – Theory and Applications*, 46(4):466–492, 2013.

[13] A.B. Kahn. Topological sorting of large networks. *Communications of the ACM*, 5(11):558–562, 1962.

[14] K. Kuratowski. Sur le problème des courbes gauches en Topologie. *Fundamenta Mathematicae*, 11:217–283, 1930.

[15] D. Lichtenstein. Planar Formulae and Their Uses. *SIAM Journal on Computing*, 11(2):329–343, 1982.

[16] M. Middendorf and F. Pfeiffer. On the complexity of the disjoint paths problem. *Combinatorica*, 13(1):97–107, 1993.

[17] H. Whitney. Congruent Graphs and the Connectivity of Graphs. *American Journal of Mathematics*, 54(1):150–168, 1932.

List of Algorithms