An SPQR-Tree Approach to Decide Special Cases of Simultaneous Embedding with Fixed Edges

J. Joseph Fowler[1], Carsten Gutwenger[2], Michael Jünger[3],
Petra Mutzel[2], and Michael Schulz[3]

[1] Department of Computer Science, University of Arizona, USA
jfowler@cs.arizona.edu

[2] Department of Computer Science, Technische Universität Dortmund, Germany
{petra.mutzel,carsten.gutwenger}@cs.tu-dortmund.de

[3] Department of Computer Science, University of Cologne, Germany[†]
{mjuenger,schulz}@informatik.uni-koeln.de

Abstract. We present a linear-time algorithm for solving the simultaneous embedding problem with fixed edges (SEFE) for a planar graph and a pseudoforest (a graph with at most one cycle) by reducing it to the following embedding problem: Given a planar graph G, a cycle C of G, and a partitioning of the remaining vertices of G, does there exist a planar embedding in which the induced subgraph on each vertex partite of $G \setminus C$ is contained entirely inside or outside C? For the latter problem, we present an algorithm that is based on SPQR-trees and has linear running time. We also show how we can employ SPQR-trees to decide SEFE for two planar graphs where one graph has at most two cycles and the intersection is a pseudoforest in linear time. These results give rise to our hope that our SPQR-tree approach might eventually lead to a polynomial-time algorithm for deciding the general SEFE problem for two planar graphs.

1 Introduction

Many practical graph drawing applications demand planar embeddings of a graph that yield additional constraints. One natural application is in obtaining simultaneous drawings of a set of related planar graphs. This is useful in the areas of bioinformatics, social sciences and software engineering. A single drawing can be insufficient in depicting complex interrelationships of different models of a system. Instead, multiple drawings may be required, each from a different perspective. The challenge is to preserve the "mental map" of the common structures in each layout so that the scientist can easily navigate between the different diagrams. To do this, common vertices and edges are placed and drawn equally in each drawing. This can be modeled via embedding constraints.

Various embedding constraints have already been studied in [2, 5, 6]; Gutwenger *et al.* [12] apply SPQR-trees to efficiently decide if a graph has a combinatorial embedding with respect to a set of hierarchical constraints modeling grouping

[†] Partially supported by the German Science Foundation (JU204/11-1).

and fixed orders of edges around a vertex. We instead address a problem that cannot be modeled by any of the previous approaches. Given a planar graph G, a cycle $C \subset G$, and a partition P of all vertices of $G \setminus C$, we ask whether there is a planar embedding of G where all vertices $v \in p$ for some part $p \in P$ lie completely inside or outside C. We give an efficient decision algorithm using SPQR-trees that can be used to solve a simultaneous embedding problem.

Given a set of planar graphs $\{G_1, G_2, \ldots, G_n\}$ on the same vertex set, a *simultaneous embedding with fixed edges* (SEFE) of $\{G_i\}$ are planar drawings Γ_i of G_i, $i \in [1..n]$, such that all vertices and all edges belonging to two graphs G_i and G_j are drawn identically in the corresponding drawings Γ_i and Γ_j. SEFE and its variant of simultaneous geometric embedding (SGE) with planar straight-line drawings as well as the other variations of simultaneous embedding have become an important branch within the field of graph drawing. It is known that deciding SEFE is NP-complete for three graphs [11] while deciding SGE is NP-hard for two graphs [8]. The complexity of deciding SEFE for two graphs is still open.

Many approaches have been made to decide the problem for some classes of graph pairs [4, 7, 9, 10]. Frati [10] showed that trees and planar graphs always have a SEFE. Fowler *et al.* [9] improved this result to show that forests, circular caterpillars (removal of all degree-1 vertices yields a cycle), K_4, and subgraphs of K_3-multiedges (an edge (x, y) with any number of edges with x or y as endpoints) are the only graphs to always have a SEFE with any planar graph. Their drawing algorithms are based upon using an optimal Euclidean shortest path algorithm [13]. We also apply this technique in our algorithms.

In this paper we examine the pairs of a planar graph G_1 with a pseudoforest G_2. A SEFE is not always guaranteed unless all non-cycle edges of G_2 are incident to the cycle, i.e., the pseudoforest happens to be a circular caterpillar. However, we show that SEFE for such pairs can be decided in polynomial time by presenting an efficient decision algorithm. We further discuss efficient decision algorithms for the case that G_2 contains two cycles and $G_1 \cap G_2$ is a pseudoforest. We think that our approach is promising in that it may eventually lead to a general polynomial time decision algorithm for testing SEFE of two graphs.

2 Preliminaries

Given some planar drawing Γ of a planar graph G, a cycle C in G forms a Jordan curve that splits the plane into two connected components. One is bounded by C and the other is unbounded as given by the Jordan curve theorem [14]. We say that some vertex $v \in G \setminus C$ lies in the *interior (exterior)* of C if it is mapped to a position in the bounded (unbounded) component.

A *combinatorial embedding* of a planar graph G is defined as a clockwise ordering of the incident edges for each vertex with respect to a crossing-free drawing of G in the Euclidean plane. A *planar embedding* is a combinatorial embedding together with a fixed *external face*.

A *block* is a maximal 2-connected subgraph of a graph G. If G is 2-connected, the *SPQR-tree* $\mathcal{T}$ of G represents its decomposition into 3-connected components comprising serial, parallel, and 3-connected structures [3]. The respective structure is given by a skeleton graph associated with each tree node which is either a cycle (S-node), a bundle of parallel edges (P-node), or a 3-connected simple graph (R-node); Q-nodes serve as representatives for the edges of G.

If G is 2-connected and planar, its SPQR-tree $\mathcal{T}$ represents all combinatorial embeddings of G. In particular, a combinatorial embedding of G uniquely defines a combinatorial embedding of each skeleton in $\mathcal{T}$, and fixing the combinatorial embedding of each skeleton uniquely defines a combinatorial embedding of G.

Given two planar graphs $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ on the same vertex set V, a *simultaneous embedding with fixed edges* (SEFE) consists of planar drawings Γ_i of G_i, $i \in [1, 2]$, such that each vertex is mapped to the same point in the plane for Γ_1 and Γ_2 and each edge in $G_1 \cap G_2$ is represented by the same simple curve in the plane for both drawings.

3 A Planar Graph, a Cycle, and a Partition

In this section, we consider the following graph embedding problem. Given a planar graph $G = (V, E)$, a cycle $C = (V_C, E_C) \subset G$, and a partition P of $V \setminus V_C$, decide whether G has a planar drawing such that all vertices of each part in P either lie completely inside or outside of C; see Algorithm 1.

The input partition P or the planar embeddings of the graph may force two vertices to be on the same side of the cycle (either both inside or both outside). We call this situation a *same-side constraint*. On the other hand, by examining all embeddings of the graph we may reveal that two vertices must be positioned on opposite sides of the cycle (one inside and one outside). We refer to this situation as an *opposite-side constraint*. The idea of the algorithm is to find all such constraints and then check whether all these constraints can be satisfied at once, i.e., whether a planar embedding with the required property exists.

The following algorithm uses an SPQR-tree $\mathcal{T}$ to examine all embeddings of the block of graph G containing the given cycle C. Each skeleton of a node of $\mathcal{T}$ may lead to constraints prohibiting some of the possible embeddings as discussed above. We use an auxiliary graph H containing all of the vertices of the original graph to maintain the occuring constraints. Same-side constraints are represented by green edges and opposite-side constraints by red edges.

We say that H is 2-colorable if its vertices can be colored with two colors, say red and green, in such a way that both endpoints of a green edge have the same color and both endpoints of a red edge have different color.

As cycles are 2-connected, the given cycle C is contained in a single block B of graph G. All other blocks are either completely inside or outside of C in all planar drawings of G. Hence, we get one same-side constraint for all vertices of each block $B' \neq B$. We can now assume to deal with a 2-connected graph G and its SPQR-tree $\mathcal{T}$ that represents all planar embeddings of G together with some cycle $C \subseteq G$. Let $\nu \in T$ be some node of the SPQR-tree, S be its skeleton and $e \in S$ be any skeleton edge. If the expansion graph of e includes any edge of C, we call e a *cycle edge*. We consider the different possibilities for ν in turn.

Algorithm 1: Deciding the embeddability of parts respecting a cycle

```
Input: Planar graph $G = (V, E)$, cycle $C = (V_C, E_C) \subseteq G$, partition $P$ of $V \setminus V_C$
Output: Returns YES if and only if $G$ has a planar embedding such that all
        induced subgraphs of each $p \in P$ lie on one side of $C$

Let $H = (V, \emptyset)$
for all parts $p \in P$ do
    Construct path in $H$ with green edges of all vertices in $p$
Block $B$ := Biconnected component of $G$ containing $C$
for all blocks $B' \neq B$ do
    Construct path in $H$ with green edges of all vertices in $B'$
Tree $\mathcal{T}$ := SPQR-tree of $B$
for all nodes $\mu \in \mathcal{T}$ do
    if skeleton $S$ of $\mu$ has at least two cycle edges then
        Cycle $C'$ := cycle consisting of all cycle edges in $S$
        if $\mu$ is R-node then
            Expand all non-cycle edges in $S$
            Construct path in $H$ with green edges of all vertices inside $C'$
            Construct path in $H$ with green edges of all vertices outside $C'$
            if there exist vertex $v$ in the interior of $C'$ and vertex $w$ in the
            exterior of $C'$ then
                Add red edge to $H$ between $v$ and $w$
        if $\mu$ is P-node then
            for all edges $e$ in $S \setminus C'$ do
                Construct path in $H$ with green edges of all vertices in the
                expansion graph of $e$
if $H$ is 2-colorable then
    return YES
else
    return NO
```

If S contains exactly one cycle edge e, then the edges belonging to the skeleton of all the other vertices must lie on the same side of C. When regarding the node of $\mathcal{T}$ belonging to e, all these vertices are contained in the expansion graph of a single edge that is not a cycle edge. Repeating this process, if necessary, we get a $\mathcal{T}$-node that has more than one cycle edge but also has a single non-cycle edge containing all of the vertices from above. When dealing with this $\mathcal{T}$-node, the necessary auxiliary graph augmentation to handle this same-side constraint is performed.

If S contains two or more cycle edges, then these cycle edges comprise a cycle in S. If S also contains non-cycle edges, ν is a P-node or an R-node.

1. In an S-node this can only occur if all edges of the skeleton are cycle edges. In this situation there is nothing to be done as this does not lead to any same-side constraints or opposite-side constraints.

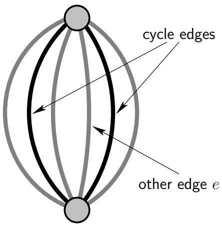

Fig. 1. Cycle edges in the skeleton of a P-node lead to same-side constraints: all vertices in the expansion graph of a non-cycle edge e are on the same side of the cycle.

2. Let ν be a P-node (see Fig. 1). All the vertices occurring in an expansion graph of any other edge in S are forced to be on one side of the cycle C.
3. Having ν as an R-node (see Fig. 2) is the most involved. The skeleton S of ν is a 3-connected graph and has hence a unique embedding (besides mirroring and choosing the outer face). The cycle edges split S into two halves: the interior and the exterior components of S. All vertices belonging to all expansion graphs of edges of one side must be on one side of the cycle in the final embedding. Neither pair of vertices w_1 and w_2 being the interior and the exterior components, respectively, may end up on the same side of the cycle. Hence, we get two same-side constraints (between all vertices in the interior and exterior components, respectively) and one opposite-side constraint (the edges from the interior and the exterior components must be separated).

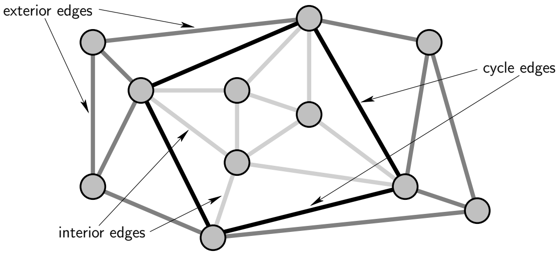

Fig. 2. Cycle edges in the skeleton of an R-node yield two same-side and one opposite-side constraints: All vertices in the expansion graphs of the interior component are on one side of the cycle while all vertices of the exterior component are on the other side.

Theorem 1. *Algorithm 1 has a runtime of $O(|V|)$ and works correctly, i.e., it returns* YES *if and only if the input graph G has a planar embedding $\mathcal{E}$ such that for each $p \in P$ all vertices in p lie on one side of C in $\mathcal{E}$.*

Proof. Obviously, the first two for-loops including the construction of $\mathcal{T}$ require only $O(|V|)$ time, thus add only $O(|V|)$ green edges to H. The third loop iterates over all nodes $\mu \in \mathcal{T}$ and expands some non-cycle edges. Observe that—for all nodes μ—the expansion graphs of these non-cycle edges do not share any edge, and thus no vertex except for vertices on the cycle C. Therefore, the whole for-loop takes $O(|V|)$ time, and we add only $O(|V|)$ green and red edges to H. Since the size of H is linear in $|V|$, we can check if H is 2-colorable using breadth-first-search in $O(|V|)$ time.

We next show that the algorithm works correctly. First, assume that the algorithm returns NO. Then the constructed auxiliary graph H is not 2-colorable. This means that two vertices v and w in H are connected by two paths: one containing an odd number of red edges and one containing an even number. This implies that v and w must lie on the same side of C (due to the path with even number of red edges), as well as on opposite sides (path with odd number of red edges). Hence, G has no such embedding.

Next assume that the algorithm returns YES in which H is 2-colorable. We pick one of the two colors to lie in the interior of C and one to lie on the outside. The choice of embeddings for every P-node and every R-node implies an embedding $\mathcal{E}$ for G. For each such node in $\mathcal{T}$, we can choose an embedding that satisfies the given choice of interior and exterior of C. In each P-node, the vertices that belong to the expansion graph of one of the parallel edges are connected by green edges in H, thus, they lie on the same side in $\mathcal{E}$. In each R-node, the vertices on both sides of the cycle are connected by green edges, respectively, while a single edge between these sets forbids both parts to lie on the same side. Finally, green edges between the vertices of the input partition yield that these vertices lie on the same side of C. □

4 A Planar Graph, a Pseudoforest, and a Decision

In this section, we apply Algorithm 1 to solve the following open problem in simultaneous embedding: Given a planar graph G_1 and a pseudoforest G_2, find an efficient algorithm to decide whether the pair $\{G_1, G_2\}$ has an SEFE; see Algorithm 2. For a few special cases of G_2 the situation becomes trivial as described by the next theorem.

Theorem 2 (Fowler et al. [9]). *Let G_1 be a planar graph and G_2 be a forest or a circular caterpillar. Then G_1 and G_2 have a* SEFE.

Next, we consider the more general case of a pseudoforest containing a cycle C in which not all non-cycle edges are incident to C. We see by the next theorem that the case is also trivial if C is not in the intersection of G_1 and G_2.

Theorem 3. *Let $G_1 = (V, E_1)$ be a planar graph and $G_2 = (V, E_2)$ be a pseudoforest with a cycle C. If C is not in $G_1 \cap G_2$, then the pair has a* SEFE.

Proof. Let edge $e \in C \setminus G_1$. Create a planar drawing of $\varGamma_1$ of G_1 in the plane using any suitable graph drawing algorithm (e.g. [1]). We construct a planar drawing $\varGamma_2$ of G_2 that, together with $\varGamma_1$, creates a SEFE of G_1 and G_2.

Draw all vertices and all edges of $G_1 \cap G_2$ in $\varGamma_2$ in the same way as in $\varGamma_1$ guaranteeing a simultaneous drawing. We still must draw all edges of $G_2 \setminus G_1$ without introducing any crossings in $\varGamma_2$. As e is not part of G_1, it has not been drawn in $\varGamma_2$ yet. We draw all edges of $G_2 \setminus G_1$ in $\varGamma_2$ one after another with e as the last edge. The order of the other edges can be chosen arbitrarily.

To do this we use an optimal Euclidean shortest path algorithm [13]. We apply the modification as done by Fowler *et al.* [9] in their drawing algorithms. A distance ε is always maintained between the shortest path and any line segment corresponding to previous part of $\varGamma_2$. This allows subsequent edges to be routed as need be in between any pair of non-incident edges that would otherwise be touching. Applying this algorithm adds at most $O(|V|)$ edge bends for each new edge (as new bends hide old bends as argued in [9]) so that the final complexity of the drawing is $O(|V|^2)$ giving an overall running time of $O(|V|^2 \log |V|)$.

As G_2 has only one cycle C and e is part of C, $G_2 \setminus \{e\}$ is a forest. Any drawing of any subgraph of $G_2 \setminus \{e\}$ has exactly one face. Hence, starting with the partial drawing of $\varGamma_2$ it is always possible to insert a route for the edges not yet drawn maintaining planarity. Even, in the last step, when edge e is inserted, the partial drawing of $\varGamma_2$ has exactly one face and thus, e can be safely inserted into $\varGamma_2$. Then $\varGamma_2$ is completed and $\{\varGamma_1, \varGamma_2\}$ is a SEFE of $\{G_1, G_2\}$. □

Due to Theorems 2 and 3 we assume G_2 to have exactly one cycle C in the intersection $G_1 \cap G_2$. By construction G_1 is planar. However, to ensure a SEFE of G_1 and G_2 we must embed G_1 in such a way that the cycle C does not separate any pair of vertices that are adjacent in G_2. On the other hand, as $G_2 \setminus C$ is a forest, this condition suffices to guarantee a SEFE of the pair $\{G_1, G_2\}$.

Theorem 4. *Let $G_1 = (V, E_1)$ be a planar graph and $G_2 = (V, E_2)$ be a pseudo-forest each on n vertices with a cycle $C \subseteq G_1 \cap G_2$. G_1 and G_2 have a* SEFE *if and only if there exists a planar drawing of G_1 such that for all edges $e = \{v, w\} \in G_2 \setminus G_1$ either both v and w lie inside or both lie outside of C.*

Proof. Assume first that G_1 has a planar drawing $\varGamma_1$ with the described property. We create a planar drawing $\varGamma_2$ of G_2 that, together with $\varGamma_1$, yields a SEFE of G_1 and G_2. Draw all vertices and all edges of $G_1 \cap G_2$ in $\varGamma_2$ in the same way as in $\varGamma_1$. As $C \subseteq G_1 \cap G_2$, the cycle is now present in $\varGamma_2$. We draw all remaining edges of $G_2 \setminus G_1$ next by using the same approach of the proof of Theorem 3.

We start with the edges e that have one endpoint in the exterior of C in $\varGamma_1$. Due to the condition on $\varGamma_1$, both endpoints of e are in the exterior of C or one endpoint is on C. As we have just drawn C and all vertices in the same way as in $\varGamma_1$, this condition also holds for the partial drawing of $\varGamma_2$. As $G_2 \setminus G_1$ is a forest there is a way to route e without introducing crossings: Imagine C and its interior as one big vertex. The partial drawing $\varGamma_2$ then has exactly one face. This also holds for edges connecting the exterior of C with C itself. The same argument holds for all the edges in the interior of C as well as the edges

Algorithm 2: Deciding SEFE for planar graph and pseudoforest pair

```
Input: Planar graph $G_1$ and pseudoforest $G_2$.
Output: YES if and only if $\{G_1, G_2\}$ has a SEFE.
if $G_2$ contains no cycle then
    Return YES.
Cycle $C :=$ the only cycle of $G_2$
if $C \nsubseteq G_1$ then
    Return YES.
Partition $P = \{P_v \mid v \in G_1 \setminus C\} :=$ trivial partition of $G_1 \setminus C$
for all edges $\{v, w\}$ of $G_2 \setminus C$ do
    UNION $P_v$ and $P_w$.
Run Algorithm 1 with input $(G_1, C, P)$.
Return output of Algorithm 1.
```

connecting the interior with C. Hence, by construction we have a planar drawing Γ_2 of G_2 that, together with Γ_1, yields a SEFE of G_1 and G_2.

Now let G_1 be without a planar drawing with the described property. Assume G_1 and G_2 have a SEFE. By definition there exist planar drawings Γ_i of G_i, $i \in [1, 2]$, such that the intersection $G_1 \cap G_2$ is drawn in the same way in both Γ_1 and Γ_2. As G_1 has no planar drawing with the described property, there exists an edge $e = \{v, w\} \in G_2 \setminus C$ such that v lies in the interior of C and w lies in the exterior of C in Γ_1. As vertices v and w and cycle C are part of $G_1 \cap G_2$, the same condition holds for Γ_2. But this means that e cannot be routed in Γ_2 without introducing a crossing in Γ_2, which is a contradiction to our assumption. Hence, G_1 and G_2 have no SEFE. □

We use the previously discussed results to create an efficient algorithm deciding the problem mentioned in the beginning of this section.

Theorem 5. *Algorithm 2 works correctly, i.e., it returns* YES *if and only if* $\{G_1, G_2\}$ *has a* SEFE. *Moreover, it has a linear runtime.*

Proof. Assume first that the algorithm returns YES, which is by one of three statements. The first returns YES if G_2 contains no cycle. But then Theorem 2 states that $\{G_1, G_2\}$ has a SEFE. The second statement returns YES if cycle C is not completely part of G_1. Theorem 3 guarantees that G_1 and G_2 have a SEFE in this situation. The last instruction is that the run of Algorithm 1 returns YES. Algorithm 1 checks whether graph G_1 can be embedded in the plane such that all partition sets of P lie completely inside or outside C. By the construction of P, this is equivalent to saying that both endpoints of every edge of $G_2 \setminus C$ lie both inside or both outside C. Then Theorem 4 yields a SEFE of G_1 and G_2.

Assume next, that the algorithm returns NO, which implies Algorithm 1 returned NO. Hence, G_1 has no planar drawing with the property of Theorem 4, which implies that G_1 and G_2 are without a SEFE.

The proposed runtime $O(|V|)$ follows directly from the complexity analysis of Algorithm 1. □

5 A Planar Graph, a Path, and a Cyclic Edge Order

In this section, we consider two embedding problems with requirements on the cyclic order of some of the edges around a vertex x or two vertices x and y that can be used to decide some special SEFE problems in Section 6.

In the first problem, x and y are two distinct vertices connected by a path p. Let e_p and e'_p be the first and last edges on p incident to x and y, respectively, where $\{e_a, e_b\}$ and $\{e'_a, e'_b\}$ are distinct edges also incident to x and y. We want to ensure that the order of these edges around x and y (amongst other possible incident edges) in a combinatorial embedding $\mathit{\Gamma}$ of G is consistent with an embedding of a graph in which x and y are connected by the three edge-disjoint paths p, $p_a = e_a, \ldots, e'_a$, and $p_b = e_b, \ldots, e'_b$. This implies that either the cyclic order around x is e_p, e_a, e_b and around y is e'_p, e'_b, e'_a or both orders are reversed. It suffices to test only one possibility, since we can generate a combinatorial embedding with the reversed orders simply by mirroring the embedding.

Let $E_x = \{e_p, e_a, e_b\}$ and $E_y = \{e'_p, e'_a, e'_b\}$. We observe that—if not all edges in $E_x \cup E_y$ are in the same block—such a required combinatorial embedding always exists; in this case, x or y is a cut vertex. We can insert the embedding of one block B' into a face of an embedding of the other block B (mirroring the embedding of B' if necessary) so that the requirements on the embedding are met. On the other hand, if all the edges in $E_x \cup E_y$ are contained in a single block B, it is sufficient to test a few simple conditions in the SPQR-tree $\mathcal{T}$ of B. The necessary and sufficient conditions are given in the lemma below.

Lemma 1. *G has a combinatorial embedding $\mathit{\Gamma}$ such that the cyclic order induced by $\mathit{\Gamma}$ on E_x is e_p, e_a, e_b and the cyclic order induced on E_y is e'_p, e'_b, e'_a if and only if*

1. *there is no block B of G containing all edges in $E_x \cup E_y$; or*
2. *there is a block B containing $E_x \cup E_y$, and its SPQR-tree $\mathcal{T}$ has neither*
   - *(a) a P-node whose skeleton contains three distinct edges e_1, e_2, e_3 such that e_p and e'_p are contained in the expansion graph of e_1, e_a and e'_b in the expansion graph of e_2, and e_b and e'_a in the expansion graph of e_3; nor*
   - *(b) an R-node whose skeleton has a combinatorial embedding such that e_p, e_a, e_b are in the expansion graphs of three distinct skeleton edges $\tilde{e}_p, \tilde{e}_a, \tilde{e}_b$ in this cyclic order, and e'_p, e'_a, e'_b are in the expansion graphs of three distinct skeleton edges $\tilde{e}'_p, \tilde{e}'_a, \tilde{e}'_b$ in this cyclic order.*

These conditions can be checked in linear time, since constructing an SQPR-tree and determining for each edge $e \in E_x \cup E_y$ in the expansion graphs of which skeleton edges it is contained, requires only linear time, and there are only two combinatorial embeddings of each R-node's skeleton.

In the second embedding problem, we consider a planar graph G with a vertex x and four distinct edges e_a, e'_a, e_b, e'_b incident to x. We want to decide if there exists an embedding $\mathit{\Gamma}$ of G that induces a cyclic order on these four edges in which e_a and e'_a (and thus also e_b and e'_b) are consecutive. The motivation for this problem is similar as for the first problem, where p is an empty path and thus x and y are identical. In this case, deciding if a feasible combinatorial embedding of G exists is even easier. We only need to consider only R-node skeletons containing x in which x is incident to at least four skeleton edges. This gives the following lemma whose conditions can be verified in linear time:

Lemma 2. *G has a combinatorial embedding Γ such that the cyclic order induced by Γ on $E_x = \{e_a, e'_a, e_b, e'_b\}$ is such that e_a and e'_a are consecutive, if and only if either*

1. *no block of G contains all edges in E_x; or*
2. *there is a block B containing all edges in E_x, and its SPQR-tree contains no R-node whose skeleton S contains x and the edges in E_x are in the expansion graphs of four distinct skeleton edges $\tilde{E}_x = \tilde{e}_a, \tilde{e}'_a, \tilde{e}_b, \tilde{e}'_b$ such that there exists a combinatorial embedding of S that induces a cyclic order on the edges in $\tilde{E}_x$ in which $\tilde{e}_a$ and $\tilde{e}'_a$ are not consecutive.*

6 Two Planar Graphs with Restrictions and a Decision

We now consider how this approach of using SPQR-trees might be extended to address more general decision problems for deciding whether a pair of graphs has a SEFE. We examine pairs of planar graphs G_1 and G_2 where we restrict both the number and the arrangement of cycles in G_2 and in $G_1 \cap G_2$.

$G_1 \cap G_2$ is a forest: We start with a more general version of Theorem 3 where we have a larger number of cycles in G_2 but still the intersection is a forest.

Theorem 6. *Let $G_1 = (V, E_1)$ be a planar graph and $G_2 = (V, E_2)$ be a planar graph where all cycles $C_i \subseteq G_2, i \in [1..k]$, are pairwise disjoint. If no C_i is contained in $G_1 \cap G_2$, then the pair $\{G_1, G_2\}$ has a* SEFE.

Proof (sketch). We adapt the proof of Theorem 3. When drawing G_2, remove one edge e_i from each $C_i \setminus G_1$ and draw the rest of G_2, which is a forest. Then insert one edge e_i after another in the same way as done with edge e in the proof of Theorem 3. As all cycles are disjoint and no further cycles exist, this method can be applied without introducing any crossings in the drawing of G_2. □

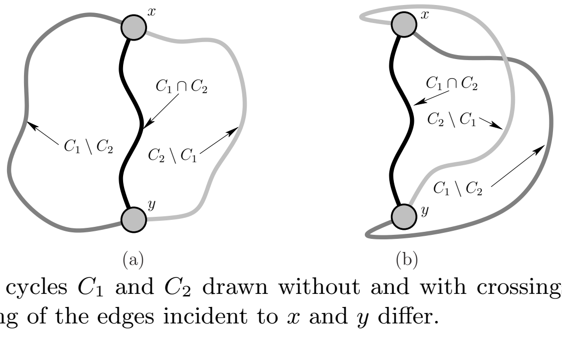

Fig. 3. The two cycles C_1 and C_2 drawn without and with crossings. The respective clockwise ordering of the edges incident to x and y differ.

Algorithm 3: Deciding SEFE for restricted planar graph pair

```
Input: Planar graphs $G_1$ and $G_2$ where $G_2$ contains exactly two cycles and
       $G_1 \cap G_2$ is a pseudoforest but not a forest.
Output: YES if and only if $\{G_1, G_2\}$ has a SEFE.
Cycle $C$ := the only cycle of $G_1 \cap G_2$
Partition $P = \{P_v \mid v \in G_1 \setminus C\}$ := trivial partition of $G_1 \setminus C$
for all edges $\{v, w\}$ of $G_2 \setminus C$ do
    UNION $P_v$ and $P_w$.
Run Algorithm 1 with input $(G_1, C, P)$.
Return output of Algorithm 1.
```

Next, we discuss the case where G_2 contains exactly two cycles that either touch in exactly one point or share a common path. With the ideas developed in Section 5 we can handle this situation efficiently.

Theorem 7. *The* SEFE *decision problem for two planar graphs G_1 and G_2 where G_2 contains exactly two cycles and $G_1 \cap G_2$ is a forest can be decided in linear time.*

Proof (sketch). Let C_1 and C_2 be the two cycles of G_2. If $C_1 \cap C_2 = \emptyset$, the case is trivial as given by Theorem 6. As G_2 contains no more cycles, $C_1 \cap C_2$ is a path p with endpoints x and y; see Fig. 3. A planar embedding of G_1 can force the outgoing edges of x and y to have a specific order leading to the situation in Fig. 3(b) in G_2 that prevents a SEFE of G_1 and G_2. However, if G_1 has an embedding that allows the right cyclic order for both x and y as in Fig. 3(a), then a SEFE can be achieved. All other edges of G_2 can be drawn without introducing crossings as in the proof of Theorem 4. Lemma 1 gives a linear time check to determine whether G_1 has an embedding such that the cyclic order for the three outgoing edges corresponds to the paths shown in Fig. 3. Lemma 2 handles the degenerate case for $x = y$, also determinable in linear time. □

$G_1 \cap G_2$ is a pseudoforest: Assume now that both G_1 and G_2 are planar graphs in which G_2 contains exactly two cycles C_1 and C_2 of which only one, say C_1, is contained in $G_1 \cap G_2$. When removing one edge of $C_2 \setminus G_1$ we are in the situation described in Section 4. This correlation allows us to construct a new decision algorithm based on Algorithm 2. We start by generalizing Theorem 4, which we use as the key ingredient to Algorithm 3.

Theorem 8. *Let $G_1 = (V, E_1)$ be a planar graph and $G_2 = (V, E_2)$ be a planar graph with exactly two cycles C_1 and C_2 where $C_1 \subseteq G_1 \cap G_2$ and $C_2 \not\subseteq G_1 \cap G_2$. G_1 and G_2 have a* SEFE *if and only if there exists a planar drawing of G_1 such that for all edges $e = \{v, w\} \in G_2 \setminus G_1$ either both v and w lie inside or both lie outside of C_1.*

Theorem 8 can be proved by using Theorem 4 to determine whether $\{G_1, G_2 \setminus \{e\}\}$ has a SEFE. In an SEFE of this smaller pair, edge $e = \{v, w\}$ can be inserted if and only if both endpoints v and w lie on the same side of C_1.

It is easy to see that Algorithm 3 works correctly. We can imitate the proof of correctness of Algorithm 2 (see Theorem 5) where this time Theorem 8 plays the role of Theorem 4.

7 Concluding Remarks and Future Applications

We have shown how to use SPQR-trees in the context of simultaneous embedding with fixed edges by presenting several new decision algorithms for some classes of graph pairs. Clearly, much future works remains, but overall this approach of using SPQR-trees seems promising in potentially yielding a polynomial-time decision algorithm for deciding whether two graphs have a SEFE, if one exists.

References

1. de Fraysseix, H., Pach, J., Pollack, R.: How to draw a planar graph on a grid. Combinatorica 10(1), 41–51 (1990)
2. Di Battista, G., Didimo, W., Patrignani, M., Pizzonia, M.: Drawing database schemas. Software: Practice and Experience 32(11), 1065–1098 (2002)
3. Di Battista, G., Tamassia, R.: On-line planarity testing. SIAM Journal on Computing 25(5), 956–997 (1996)
4. Di Giacomo, E., Liotta, G.: A note on simultaneous embedding of planar graphs. In: EWCG 2005. pp. 207–210 (2005)
5. Dornheim, C.: Planar graphs with topological constraints. Journal on Graph Algorithms and Applications 6(1), 27–66 (2002)
6. Eiglsperger, M., Fößmeier, U., Kaufmann, M.: Orthogonal graph drawing with constraints. In: Proc. SODA 2000. pp. 3–11 (2000)
7. Erten, C., Kobourov, S.G.: Simultaneous embedding of planar graphs with few bends. In: Pach, J. (ed.) GD 2004. LNCS, vol. 3383, pp. 195–205 (2005)
8. Estrella-Balderrama, A., Gassner, E., Jünger, M., Percan, M., Schaefer, M., Schulz, M.: Simultaneous geometric graph embeddings. In: Hong, S.H., Nishizeki, T., Quan, W. (eds.) GD 2007. LNCS, vol. 4875, pp. 280–290 (2008)
9. Fowler, J.J., Jünger, M., Kobourov, S.G., Schulz, M.: Characterizations of restricted pairs of planar graphs allowing simultaneous embedding with fixed edges. In: WG 2008 (To appear)
10. Frati, F.: Embedding graphs simultaneously with fixed edges. In: Kaufmann, M., Wagner, D. (eds.) GD 2006. LNCS, vol. 4372, pp. 108–113 (2007)
11. Gassner, E., Jünger, M., Percan, M., Schaefer, M., Schulz, M.: Simultaneous graph embeddings with fixed edges. In: Fomin, F.V. (ed.) WG 2006. LNCS, vol. 4271, pp. 325–335 (2006)
12. Gutwenger, C., Klein, K., Mutzel, P.: Planarity testing and optimal edge insertion with embedding constraints. In: Kaufmann, M., Wagner, D. (eds.) GD 2006. LNCS, vol. 4372, pp. 126–137 (2007)
13. Hershberger, J., Suri, S.: An optimal algorithm for Euclidean shortest paths in the plane. SIAM Journal on Computing 28(6), 2215–2256 (1999)
14. Veblen, O.: Theory on plane curves in non-metrical analysis situs. Transactions of the American Mathematical Society 6, 83–98 (1905)