Testing Mutual Duality of Planar Graphs*

Patrizio Angelini[1], Thomas Bläsius[2], and Ignaz Rutter[2]

[1] Università Roma Tre
angelini@dia.uniroma3.it
[2] Karlsruhe Institute of Technology (KIT)
{blaesius,rutter}@kit.edu

Abstract. We introduce and study the problem MUTUAL PLANAR DUALITY, which asks for planar graphs G_1 and G_2 whether G_1 can be embedded such that its dual is isomorphic to G_2. We show NP-completeness for general graphs and give a linear-time algorithm for biconnected graphs.

We consider the *common dual relation* $\sim$, where $G_1 \sim G_2$ if and only they admit embeddings that result in the same dual graph. We show that $\sim$ is an equivalence relation on the set of biconnected graphs and devise a succinct, SPQR-tree-like representation of its equivalence classes. To solve MUTUAL PLANAR DUALITY for biconnected graphs, we show how to do isomorphism testing for two such representations in linear time.

A special case of MUTUAL PLANAR DUALITY is testing whether a graph is self-dual. Our algorithm can handle the case of biconnected graphs in linear time and our NP-hardness proof extends to self-duality and also to map self-duality testing (which additionally requires to preserve the embedding).

1 Introduction

Let G be a planar graph with embedding $\mathcal{G}$ and let F be the set of faces of $\mathcal{G}$. The *dual* of G with respect to $\mathcal{G}$ is the graph $G^\star = (F, E^\star)$ with $E^\star = \{e^\star \mid e \in E\}$. The *dual edge* $e^\star$ of e connects the two faces incident to e in $\mathcal{G}$. Thus, $G^\star$ models the adjacencies of faces of G with respect to $\mathcal{G}$. We consider the problem MUTUAL PLANAR DUALITY. Given two planar graphs G_1 and G_2, is there an embedding $\mathcal{G}_1$ of G_1 such that the dual $G_1^\star$ of G_1 with respect to $\mathcal{G}_1$ is isomorphic to G_2? All graphs we consider are implicitly allowed to have multiple edges and loops. If G_1 is triconnected, it has a fixed planar embedding [13] and thus MUTUAL PLANAR DUALITY reduces to testing graph isomorphism for planar graphs, which is linear-time solvable due to Hopcroft and Wong [8]. Observe that bi- and triconnectivity of a planar graph is invariant under dualization [12]. For non-triconnected planar graphs MUTUAL PLANAR DUALITY is more complicated, since changing the embedding of G_1 influences the structure of its dual graph. In fact, we show that MUTUAL PLANAR DUALITY is NP-complete in general.

On the other hand, for biconnected planar graphs we provide a linear-time algorithm solving MUTUAL PLANAR DUALITY that is based on the definition of a new data structure that we call *dual SPQR-tree* in analogy with the SPQR-tree [5]. The dual SPQR-trees, together with a newly-defined set of operations, allows to succinctly represents and

* Partially supported by ESF project 10-EuroGIGA-OP-003 GraDR "Graph Drawings and Representations". This work began during a visit of Angelini at Karlsruhe Institute of Technology.

L. Cai, S.-W. Cheng, and T.-W. Lam (Eds.): ISAAC2013, LNCS 8283, pp. 350–360, 2013.

efficiently handle all the dual graphs of a biconnected planar graph. This data structure has an interesting implication on the structure of the dual graphs of a biconnected planar graph. Namely, consider the *common dual relation* $\sim$, where $G_1 \sim G_2$ if and only if they have a common dual graph. We show that $\sim$ is not transitive on the set of connected planar graphs. However, it follows from the dual SPQR-tree that $\sim$ is an equivalence relation on the set of biconnected planar graphs. In particular, the graphs represented by a dual SPQR-tree form an equivalence class. Thus, testing MUTUAL PLANAR DUALITY reduces to testing whether two dual SPQR-trees represent the same equivalence class. It is not hard to see that two biconnected graphs are related via the common dual relation if and only if they have the same graphic matroid (which again does not hold for general planar graphs). With this insight, one can use the one-to-many reduction from graphic matroid isomorphism testing to graph isomorphism testing by Rao and Sarma [9] to solve MUTUAL PLANAR DUALITY for biconnected planar graphs in polynomial time. We give a one-to-one reduction leading to a linear-time algorithm.

We note that the common-dual relation is closely related to 2-isomorphisms, studied by Whitney [14]. Two graphs are 2-isomorphic if and only if their cycle matroids are isomorphic. On biconnected graphs, the notions coincide, and our algorithm implies a linear-time isomorphism testing algorithm for graphic matroids of planar graphs.

We believe that the new data structure of dual SPQR-trees can be used to efficiently solve other related problems. In many applications it is desirable to find an embedding of a given planar graph that optimizes certain criteria, which can often be naturally described in terms of the dual graph with respect to the chosen embedding. For example, Bienstock and Monma [4], and Angelini et al. [1] seek an embedding of a planar graph minimizing the largest distance of internal faces to the external face. In terms of the dual graph this corresponds to minimizing the largest distance of a vertex to all other vertices. For problems of this kind it can be useful to work directly with a representation of all dual graphs, instead of taking the detour over a representation of all planar embeddings.

We finally remark that MUTUAL PLANAR DUALITY is a generalization of the self-duality of planar graphs [10]. A graph G is *graph self-dual* if it admits an embedding such that its dual $G^\star$ is isomorphic to G. We call the corresponding decision problem GRAPH SELF-DUALITY. A stronger form of self-duality is defined as follows. A graph G is *map self-dual* [11] if and only if G has an embedding $\mathcal{G}$ such that there exists an isomorphism from G to its dual graph $G^\star$ that preserves embedding $\mathcal{G}$. The corresponding decision problem is called MAP SELF-DUALITY. Since triconnected planar graphs have a unique planar embedding, GRAPH SELF-DUALITY and MAP SELF-DUALITY are equivalent for them. Servatius and Servatius [11] show the existence of biconnected planar graphs that are graph self-dual but not map self-dual. Servatius and Christopher [10] show how to construct self-dual graphs from given planar graphs. Archdeacon and Richter [3] give a set of constructions for triconnected self-dual graphs and show that every such graph can be constructed in this way. To the best of our knowledge the computational complexity of testing MAP or GRAPH SELF-DUALITY is open. Since GRAPH SELF-DUALITY is a special case of MUTUAL PLANAR DUALITY, our algorithm can be used to solve GRAPH SELF-DUALITY in linear time when G is biconnected. Moreover, our NP-hardness proof for general instances extends to MAP and GRAPH SELF-DUALITY.

Outline. In Section 2 we show that MUTUAL PLANAR DUALITY is NP-complete, even if both input graphs are required to be simple. The proof can be extended to show that MAP SELF-DUALITY and GRAPH SELF-DUALITY are NP-complete in general. To solve MUTUAL PLANAR DUALITY efficiently for biconnected graphs, we first describe decomposition trees as a generalization of SPQR-trees in Section 3. In Section 4 we describe the dual SPQR-tree and show that it succinctly represents all dual graphs of a biconnected planar graph. We consider the common dual relation in Section 5 and show that $\sim$ is not transitive on the set of connected planar graphs. On the other hand, we show that it follows from the dual SPQR-tree that $\sim$ is an equivalence relation on the set of biconnected planar graphs. This implies that solving MUTUAL PLANAR DUALITY is equivalent to testing whether two dual SPQR-trees represent the same equivalence class. In Section 6 we show that this reduces to testing graph isomorphism of two planar graphs, which leads to a linear-time algorithm for MUTUAL PLANAR DUALITY, including GRAPH SELF-DUALITY as a special case. Omitted proofs can be found in the full version of this paper[2].

2 Complexity

In this section we first show that MUTUAL PLANAR DUALITY is NP-complete by a reduction from 3-PARTITION. Then we show that the resulting instances of MUTUAL PLANAR DUALITY can be further reduced to equivalent instances of MAP and GRAPH SELF-DUALITY. An instance (A, B) of 3-PARTITION consists of a positive integer B and a set $A = \{a_1, \dots, a_{3m}\}$ of $3m$ integers with $B/4 < a_i < B/2$ for $i = 1, \dots, 3m$. The question is whether A admits a partition $\mathcal{A}$ into a set of triplets such that for each triplet $X \in \mathcal{A}$ we have $\sum_{x \in X} x = B$. The problem 3-PARTITION is strongly NP-hard [6], i.e., it remains NP-hard even if B is bounded by a polynomial in m.

Theorem 1. MUTUAL PLANAR DUALITY *is NP-complete, even for simple graphs.*

Proof. Clearly, MUTUAL PLANAR DUALITY is in NP, as we can guess an embedding for graph G_1 and then check in polynomial time whether $G_1^\star$ is isomorphic to G_2.

To show NP-hardness we give a reduction from 3-PARTITION. Our construction first contains loops, later we show how to get rid of them. Let (A, B) be an instance of 3-PARTITION with $|A| = 3m$. The graph G_1 contains a wheel of size m, i.e., a cycle $v_1, \dots, v_m$ together with a *center* u connected to each v_i. Additionally, for each element $a_i \in A$ we create a star T_i with $a_i - 1$ leaves and connect its center to u; see Figure 1(a). The graph G_2 is a wheel of size m along with B loops at every vertex except for the center; see Figure 1(b). We claim that G_1 and G_2 form a YES-instance of MUTUAL PLANAR DUALITY if and only if (A, B) is a YES-instance of 3-PARTITION.

Suppose that there exists a partition $\mathcal{A}$ of A. The embedding of the wheel in G_1 is fixed and it has exactly m faces incident to the center u. The remaining degree of freedom is to decide the embedding of the trees T_i into these m faces. For each triplet $X = \{a_i, a_j, a_k\} \in \mathcal{A}$ we pick a distinct such face and embed T_i, T_j and T_k into it. Call the resulting embedding $\mathcal{G}_1$ and consider the dual $G_1^\star$ with respect to $\mathcal{G}_1$. The wheel of G_1 determines a wheel of size m in $G_1^\star$. Consider a tree T_i that is embedded in a face f. Since T_i contains a_i bridges, which are all embedded in f, the corresponding

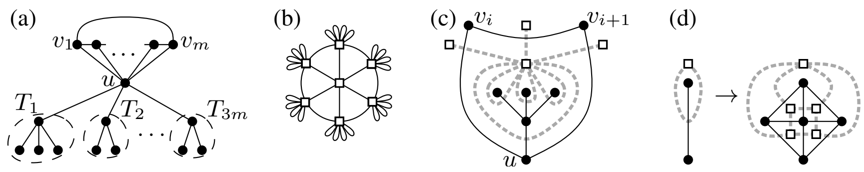

Fig. 1. The graphs G_1 (a) and G_2 (b) of the reduction from 3-PARTITION. (c) Embedding a tree T_i inside a face f creates a_i loops at the corresponding dual vertex. (d) Bridges and corresponding loops can be replaced by small graphs.

vertex of $G_1^\star$ has a_i loops; see Figure 1(c). Due to the construction each face contains exactly three trees with a total of B bridges. Thus, $G_1^\star$ is isomorphic to G_2.

Conversely, assume to have embeddings G_1 and G_2 such that the dual $G_1^\star$ of G_1 is isomorphic to G_2. Again, the wheel in G_1 forms m faces incident to u, and since $G_1^\star$ is isomorphic to G_2, the trees must be embedded such that each face contains exactly B bridges. Since embedding T_i inside a face f places a_i bridges into f and since $B/4 < a_i < B/2$, each face contains exactly three trees. Thus, the set of triplets determined by trees that are embedded in the same face form a solution to 3-PARTITION.

Clearly, the transformation can be computed in polynomial time, and thus MUTUAL PLANAR DUALITY is NP-hard. Moreover, the graph G_2 can be made simple (G_1 is already simple) by replacing each bridge in G_1 and each loop in G_2 with a 4-wheel as shown in Figure 1(d). The resulting graphs G_1' and G_2' are obviously dual to each other if and only if G_1 and G_2 are dual to each other. Moreover, G_1' and G_2' are simple. □

In the following we show how the above construction can be used to show NP-completeness for MAP and GRAPH SELF-DUALITY. To this end, we use the *adhesion* operation introduced by Servatius and Christopher [10]. Let v be a vertex of G incident to a face f. Then the adhesion of G and its dual $G^\star$ (with respect to v and f) is obtained by identifying v in G and $f^\star$ in $G^\star$. Servatius and Christopher [10] show that the adhesion of a plane graph and its dual is graph self-dual. Moreover, although not explicitly mentioned, they show that this adhesion is even map self-dual. To show the following theorem we essentially transform the instance of MUTUAL PLANAR DUALITY consisting of the two graphs G_1 and G_2 described in the proof of Theorem 1 into an equivalent instance of MAP and GRAPH SELF-DUALITY by forming the adhesion of G_1 and G_2.

Theorem 2. GRAPH SELF-DUALITY *and* MAP SELF-DUALITY *are NP-complete.*

Sketch of Proof. Both problems are in NP. Let G_1 and G_2 form an instance of MUTUAL PLANAR DUALITY obtained from an instance of 3-PARTITION as in the proof of Theorem 1. Let G be the graph obtained by identifying a vertex that is not the center of the wheel in G_2 with the vertex u in G_1. The theorem is implied by the following claims.

Claim 1. If G is a YES-instance of MAP SELF-DUALITY, it is a YES-instance of GRAPH SELF-DUALITY.

Claim 2. If G_1 and G_2 form a YES-instance of MUTUAL PLANAR DUALITY, then G is a YES-instance of MAP SELF-DUALITY.

Claim 3. If G is a YES-instance of GRAPH SELF-DUALITY, then G_1 and G_2 form a YES-instance of MUTUAL PLANAR DUALITY. □

3 Decomposition Trees and the SPQR-Tree

A graph is *connected* if there exists a path between any pair of vertices. A *separating k-set* is a set of k vertices whose removal disconnects the graph. Separating 1-sets and 2-sets are *cutvertices* and *separation pairs*, respectively. A connected graph is *biconnected* if it does not have a cut vertex and *triconnected* if it does not have a separation pair. The maximal biconnected components of a graph are called *blocks*.

In the following we consider *decomposition trees* of biconnected planar graphs. *SPQR-trees* [5], which can be computed in linear time [7], are a special case of decomposition trees. Let G be a planar biconnected graph and let $\{s, t\}$ be a *split pair*, that is either a separation pair or a pair of adjacent vertices. Let further H_1 and H_2 be two subgraphs of G such that $H_1 \cup H_2 = G$ and $H_1 \cap H_2 = \{s, t\}$. Consider the tree $\mathcal{T}$ consisting of two nodes μ_1 and μ_2 associated with the graphs $H_1 + (s, t)$ and $H_2 + (s, t)$, respectively. For each node μ_i, the graph $H_i + (s, t)$ associated with it is the *skeleton* of μ_i, denoted by $\mathrm{skel}(\mu_i)$, and the special directed edge (s, t) is called *virtual edge*. The edge connecting the nodes μ_1 and μ_2 in $\mathcal{T}$ associates the virtual edge $\varepsilon_1 = (s, t)$ in $\mathrm{skel}(\mu_1)$ with the virtual edge $\varepsilon_2 = (s, t)$ in $\mathrm{skel}(\mu_2)$; we say that ε_1 is the *twin* of ε_2 and vice versa. Moreover, we say that ε_1 in $\mathrm{skel}(\mu_1)$ *corresponds* to the neighbor μ_2 of μ_1. This can be expressed as a bijective map $\mathrm{corr}_\mu \colon E(\mathrm{skel}(\mu)) \to N(\mu)$ for each node μ, where $E(\mathrm{skel}(\mu))$ and $N(\mu)$ denote the set of edges in $\mathrm{skel}(\mu)$ and the set of neighbors of μ in $\mathcal{T}$, respectively. In the example above we have $\mathrm{corr}(\varepsilon_1) = \mu_2$ and $\mathrm{corr}(\varepsilon_2) = \mu_1$ (the subscript of corr is omitted as it is clear from the context).

The above-described procedure is called *decomposition* and can be applied further to the skeletons of the nodes of $\mathcal{T}$, leading to a larger tree with smaller skeletons. The decomposition can be undone by *contracting* an edge in $\mathcal{T}$. Let $\{\mu, \mu'\}$ be an edge in $\mathcal{T}$ and let ε be the virtual edge in $\mathrm{skel}(\mu)$ with $\mathrm{corr}(\varepsilon) = \mu'$ having ε' in $\mathrm{skel}(\mu')$ as twin. The contraction of $\{\mu, \mu'\}$ collapses μ and μ' into a single node with the following skeleton. The skeletons of μ and μ' are *glued* together at the twins ε and ε' according to their orientation, i.e., the sources and targets of ε and ε' are identified with each other, respectively. The resulting virtual edge is removed. Iteratively applying the contraction in $\mathcal{T}$ leads to a tree consisting of a single node μ, whose skeleton is independent from the contraction order. The graph *represented* by $\mathcal{T}$ is $\mathrm{skel}(\mu)$.

A *reversed decomposition tree* is defined as a decomposition tree with the only difference that in the decomposition step one of the two twin edges is reversed and in the contraction step they are glued together oppositely. Note that a reversed decomposition tree can be transformed into an equivalent normal decomposition tree representing the same graph by reversing one virtual edge in each pair of twin edges.

A special decomposition tree is the SPQR-tree. A decomposition tree is an SPQR-tree if each inner node is either an S-, a P-, or an R-node whose skeletons contain only virtual edges forming a cycle, a bunch of at least three parallel edges or a triconnected planar graph, respectively, such that no two S-nodes and no two P-nodes are adjacent. Each leaf is a Q-node whose skeleton consists of two vertices connected by one virtual and one normal edge. The SPQR-tree of a biconnected planar graph is unique up to reversal of twins. We assume that all virtual edges in P-node skeletons are oriented in the same direction and those in S-node skeletons form a directed cycle.

There is a bijection between the embeddings of a biconnected graph G and the set of all combinations of embeddings of the skeletons in its SPQR-tree $\mathcal{T}$. The embedding choices for the skeletons consist of reordering the parallel edges in a P-node and flipping the skeleton of an R-node. Fixing the embeddings of skeletons in an arbitrary decomposition tree $\mathcal{T}$ also determines a planar embedding of the represented graph G. However, there may be planar embeddings that are not represented by $\mathcal{T}$.

We assume the skeletons of the SPQR-tree of a graph to be embedded graphs if and only if the graph itself is embedded.

4 Succinct Representation of all Duals of a Biconnected Graph

Let G be a biconnected graph with SPQR-tree $\mathcal{T}$ and planar embedding $\mathcal{G}$. In the following we study the effects of changing the embedding of G on its dual graph $G^\star$. To this end, we do not consider the graphs themselves but their SPQR-trees. We first show how the SPQR-tree of $G^\star$ can be directly obtained from the SPQR-tree of G.

We first define the *dual decomposition tree* $\mathcal{T}^\star$ of a decomposition tree $\mathcal{T}$ representing G (with respect to a fixed embedding $\mathcal{G}$ of G represented by $\mathcal{T}$). Essentially, $\mathcal{T}^\star$ is obtained from $\mathcal{T}$ by replacing each skeleton with its directed dual and interpreting the resulting tree as a reversed decomposition tree. More precisely, for each node μ in $\mathcal{T}$, the dual decomposition tree $\mathcal{T}^\star$ contains a *dual node* $\mu^\star$ having the dual of $\mathrm{skel}(\mu)$ as skeleton. An edge $\varepsilon^\star$ in $\mathrm{skel}(\mu^\star)$ dual to a virtual edge ε in $\mathrm{skel}(\mu)$ is again virtual and oriented from right to left with respect to the orientation of ε. Two virtual edges in $\mathcal{T}^\star$ are twins if and only if their primal edges are twins. This has the effect that $\mathrm{corr}(\varepsilon)^\star = \mathrm{corr}(\varepsilon^\star)$ holds. In case $\mathcal{T}$ is the SPQR-tree of G, the dual of a triconnected skeleton is triconnected, the dual of a (directed) cycle is a bunch of parallel edges (all directed in the same direction), and the dual of a normal edge with a parallel virtual edge is a normal edge with a parallel virtual edge. Thus, if a node μ in $\mathcal{T}$ is an S-, P-, Q-, or R-node, its dual node $\mu^\star$ in $\mathcal{T}^\star$ is a P-, S-, Q-, or R-node, respectively. Thus, the dual SPQR-tree is again an SPQR-tree and not just an arbitrary decomposition tree.

Lemma 1. *Let G be a biconnected planar graph with SPQR-tree $\mathcal{T}$ and embedding $\mathcal{G}$. The dual SPQR-tree $\mathcal{T}^\star$ with respect to $\mathcal{G}$ is the reversed SPQR-tree of the dual $G^\star$.*

Sketch of Proof. We show the claim for general decomposition trees. As illustrated in Figure 2, first contracting an edge $\{\mu, \mu'\}$ in a decomposition tree $\mathcal{T}$ and then taking the dual decomposition tree is equivalent to first taking the dual decomposition tree $\mathcal{T}^\star$ and then contracting $\{\mu^\star, \mu'^\star\}$. Applying this operation iteratively until the trees $\mathcal{T}$ and $\mathcal{T}^\star$ consist of single nodes directly shows that the reversed decomposition tree $\mathcal{T}^\star$ represents the graph $G^\star$ dual to the graph G represented by $\mathcal{T}$. $\square$

In the following we consider how the dual SPQR-tree changes when the embedding of skeletons in the SPQR-tree change. Flipping the skeleton of an R-node and reordering the virtual edges in a P-node give rise to the following two operations: *reversal* of R-nodes and *restacking* of S-nodes. A reversal applied on an R-node μ reverses the direction of all edges in $\mathrm{skel}(\mu)$. Note that this only affects how $\mathrm{skel}(\mu)$ is glued to the skeletons of its adjacent nodes. Let μ be an S-node with virtual edges $\varepsilon_1, \dots, \varepsilon_k$. A restacking of μ picks an arbitrary ordering of $\varepsilon_1, \dots, \varepsilon_k$ and glues their end-points such that they create a directed cycle C in that order. Then, $\mathrm{skel}(\mu)$ is replaced by C.

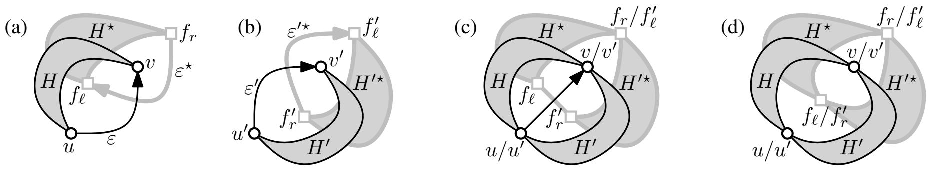

Fig. 2. (a)–(c) Glueing the virtual edge ε and ε'. (d) Removing the resulting edge.

Lemma 2. *Let $\mathcal{T}$ and $\mathcal{T}^\star$ be the SPQR-trees of an embedded biconnected planar graph and of its dual, respectively. Flipping an R-node and reordering a P-node in $\mathcal{T}$ corresponds to reversing its dual R-node and restacking its dual S-node, respectively.*

Proof. Due to Lemma 1 we can work with the dual SPQR-tree instead of the SPQR-tree of the dual. Obviously, flipping an R-node μ in $\mathcal{T}$ exchanges left and right in $\mathrm{skel}(\mu)$ and thus reverses the orientation of each virtual edge in $\mathrm{skel}(\mu^\star)$, where $\mu^\star$ is the node in $\mathcal{T}^\star$ dual to μ. Thus, flipping μ corresponds to a reversal of $\mu^\star$. Similarly, reordering the virtual edges in the skeleton of a P-node μ has the effect that the virtual edges in its dual S-node $\mu^\star$ are restacked, yielding a different cycle. Note that this cycle is again directed since the virtual edges in μ are still all oriented in the same direction. □

This shows that the SPQR-tree of the dual graph with respect to a fixed embedding can be used to represent the dual graphs with respect to all possible planar embeddings by allowing reversal and restacking operations. We say that an SPQR-tree *represents a dual graph* if it can be obtained by applying reversal and restacking operations.

Theorem 3. *The dual SPQR-tree of a biconnected planar graph G represents exactly the dual graphs of G.*

When we are not interested in the embedding of the dual graph but only in its structure, we may also allow the usual SPQR-tree operations, that is flipping R-nodes and reordering the edges in P-nodes. Note that the reversal operation applied to P-nodes only changes the embedding of the graph and not its structure. Moreover, reversing a Q-node does not change anything and the reversal of an S-node can be seen as a special way of restacking it. This observation can be used to show the following lemma.

Lemma 3. *Let G be a biconnected planar graph with embedding $\mathcal{G}$ and let $G^\star$ be its dual graph with SPQR-tree $\mathcal{T}^\star$. Let $\mathcal{T}^\star_\varepsilon$ be the SPQR-tree obtained from $\mathcal{T}^\star$ by reversing the orientation of the virtual edge ε in $\mathcal{T}^\star$ and let $G^\star_\varepsilon$ be the graph it represents. There exists an embedding $\mathcal{G}_\varepsilon$ of G such that $G^\star_\varepsilon$ is the dual graph of G with respect to $\mathcal{G}_\varepsilon$.*

Sketch of Proof. Let μ be the node in $\mathcal{T}^\star$ containing the virtual edge ε and let $\mathrm{corr}(\varepsilon) = \mu'$ be the neighbor of μ corresponding to ε. Removing the edge $\{\mu, \mu'\}$ splits $\mathcal{T}^\star$ into two subtrees $\mathcal{T}^\star_\mu$ and $\mathcal{T}^\star_{\mu'}$. One can show that the reversal of all nodes in one of these subtrees (no matter which one) yields an SPQR-tree $\mathcal{T}^\star_{\mu\mu'}$ representing $G^\star_\varepsilon$. Then it follows by Lemma 2 and the observation above, that $G^\star_\varepsilon$ is a dual graph of G. □
Lemma 2 and Lemma 3 together yield the following theorem.

Theorem 4. *For two SPQR-trees $\mathcal{T}_1$ and $\mathcal{T}_2$, the following three statements are equivalent. 1. $\mathcal{T}_1$ and $\mathcal{T}_2$ represent the same set of dual graphs. 2. $\mathcal{T}_1$ and $\mathcal{T}_2$ can be transformed into each other using reversal and restacking operations. 3. $\mathcal{T}_1$ and $\mathcal{T}_2$ can be transformed one into the other by choosing orientations for the virtual edges and by restacking S-node skeletons.*

5 Equivalence Relation

We define the relation $\sim$ on the set of planar graphs as follows. Two graphs G_1 and G_2 are related, i.e., $G_1 \sim G_2$, if and only if G_1 and G_2 can be embedded such that they have the same dual graph $G_1^\star = G_2^\star$. We call $\sim$ the *common dual relation*.

Theorem 5. *The common dual relation $\sim$ is an equivalence relation on the set of biconnected planar graphs. For a biconnected planar graph G, the set of dual graphs of G is an equivalence class with respect to $\sim$.*

Proof. Clearly, $\sim$ is symmetric and reflexive. For the transitivity let G_1, G_2 and G_3 be three biconnected planar graphs such that $G_1 \sim G_2$ and $G_2 \sim G_3$. Let further $\mathcal{T}_1^\star$, $\mathcal{T}_2^\star$ and $\mathcal{T}_3^\star$ be the dual SPQR-trees representing all duals of G_1, G_2 and G_3, respectively. Due to $G_1 \sim G_2$ there exists a graph G that is represented by $\mathcal{T}_1^\star$ and $\mathcal{T}_2^\star$. Thus, $\mathcal{T}_1^\star$ and $\mathcal{T}_2^\star$ can both be transformed into the SPQR-tree representing G using reversal and restacking operations, which shows that they represent the same set of duals (Theorem 4). The same argument shows that G_2 and G_3 have the same set of dual graphs. Thus, also G_1 and G_3 have exactly the same set of dual graphs, which yields $G_1 \sim G_3$.

For the second statement, let $C^\star$ be the set of dual graphs of G. Clearly, for $G_1^\star, G_2^\star \in C^\star$ the graph G is a common dual, thus $G_1^\star \sim G_2^\star$. On the other hand, let $G_1^\star \in C^\star$ and $G_1^\star \sim G_2^\star$. By the above argument, $G_1^\star$ and $G_2^\star$ have the same set of dual graphs. Thus G is a dual graph of $G_2^\star$ yielding $G_2^\star \in C^\star$. □

Theorem 5 shows that the equivalence class C of a biconnected planar graph G with respect to the common dual relation is exactly the set of dual graphs that is represented by the SPQR-tree $\mathcal{T}$ of G. The dual SPQR-tree $\mathcal{T}^\star$ of G also represents a set of dual graphs forming the equivalence class $C^\star$. We say that $C^\star$ is the *dual equivalence class* of C. Given an arbitrary graph $G \in C$ and an arbitrary graph $G^\star \in C^\star$, graphs G and $G^\star$ can be embedded such that they are dual to each other, since $C^\star$ contains exactly the graphs that are dual to G. The problems MUTUAL PLANAR DUALITY and GRAPH SELF-DUALITY can be reformulated in terms of the equivalence classes of the common dual relation. Two biconnected planar graphs are a YES-instance of MUTUAL PLANAR DUALITY if and only if their equivalence classes are dual to each other. A biconnected planar graph is graph self-dual if and only if its equivalence class is dual to itself. This in particular means that either each or no graph in an equivalence class is graph self-dual.

Although it might seem quite natural that the common dual relation is an equivalence relation, this is not true for general planar graphs, see Figure 3.

Theorem 6. *The common dual relation $\sim$ is not transitive on the set of planar graphs.*

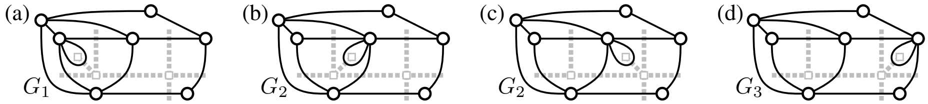

Fig. 3. The graphs G_1 (a) and G_2 (b) have a common dual and the graphs G_2 (c) and G_3 (d) have a common dual. The graphs G_1 and G_3 do not have a common dual.

6 Solving Mutual Planar Duality for Biconnected Graphs

The problem Mutual Planar Duality can be rephrased as follows.

Corollary 1. *Two biconnected planar graphs G_1 and G_2 with SPQR-trees $\mathcal{T}_1$ and $\mathcal{T}_2$ form a* Yes*-instance of* Mutual Planar Duality *if and only if $\mathcal{T}_2$ and the dual SPQR-tree $\mathcal{T}_1^\star$ represent the same dual graphs.*

In the following we show that two SPQR-trees represent the same set of dual graphs if and only if they are *dual isomorphic* (we define this in a moment). Then we show that testing the existence of such an isomorphism reduces to testing graph isomorphism for planar graphs. Figure 4(a) sketches this strategy.

For two graphs G and G' with vertices $V(G)$ and $V(G')$ and edges $E(G)$ and $E(G')$, respectively, a map $\varphi\colon V(G) \to V(G')$ is a *graph isomorphism* if it is bijective and $\{u, v\} \in E(G)$ if and only if $\{\varphi(u), \varphi(v)\} \in E(G')$ (for directed graphs, the direction of the edges is disregarded). A graph isomorphism φ induces a bijection between $E(G)$ and $E(G')$ and we use $\varphi(e)$ for $e \in E(G)$ to express this bijection. As we consider undirected edges, fixing $\varphi(\cdot)$ only for the edges is not sufficient. A *dual SPQR-tree isomorphism* between two SPQR-trees $\mathcal{T}$ and $\mathcal{T}'$ consists of several maps. First, a map $\varphi\colon V(\mathcal{T}) \longrightarrow V(\mathcal{T}')$ such that

(I) φ is a graph isomorphism between $\mathcal{T}$ and $\mathcal{T}'$; and
(II) for each node $\mu \in V(\mathcal{T})$, the node $\varphi(\mu) \in V(\mathcal{T}')$ is of the same type.

Second, a map $\varphi_\mu\colon V(\mathrm{skel}(\mu)) \longrightarrow V(\mathrm{skel}(\varphi(\mu)))$ for every R-node μ in $\mathcal{T}$ such that

(III) φ_μ is a graph isomorphism between $\mathrm{skel}(\mu)$ and $\mathrm{skel}(\varphi(\mu))$; and
(IV) $\mathrm{corr}(\varphi_\mu(\varepsilon)) = \varphi(\mathrm{corr}(\varepsilon))$ holds for every virtual edge ε in $\mathrm{skel}(\mu)$.

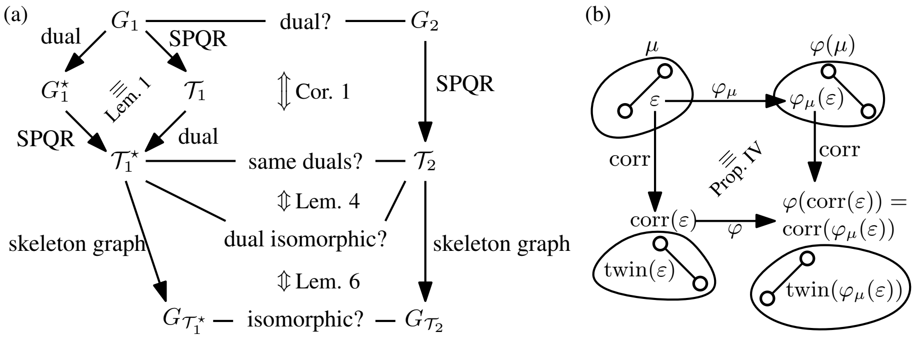

Fig. 4. (a) Overview of our strategy. (b) Commutative diagram illustrating Property IV

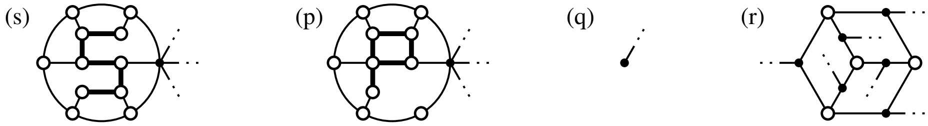

Fig. 5. The subgraphs H_μ of the skeleton graph depending on the type of the node μ. The small black vertices are the attachment vertices.

If there is a dual SPQR-tree isomorphism between $\mathcal{T}$ and $\mathcal{T}'$, then we say that $\mathcal{T}$ and $\mathcal{T}'$ are *dual isomorphic*. Note that Property IV (illustrated in Figure 4(b)) is a natural requirement and one would usually require it also for S-nodes (for P-nodes it does not make sense since every permutation is an isomorphism on its skeleton). However, not requiring it for S-nodes implicitly allows restacking their skeletons. As the graph isomorphisms $\varphi_\mu(\cdot)$ do not care about the orientation of virtual edges, it is also implicitly allowed to reverse them. We get the following lemma showing that this definition of dual SPQR-tree isomorphism is well suited for our purpose.

Lemma 4. *Two SPQR-trees represent the same set of dual graphs if and only if they are dual isomorphic.*

We reduce dual SPQR-tree isomorphism testing to graph isomorphism testing for planar graphs, which can be solved in linear time [8]. We define the *skeleton graph* $G_\mathcal{T}$ of an SPQR-tree $\mathcal{T}$ as follows. For each node μ in $\mathcal{T}$ there is a subgraph H_μ in $G_\mathcal{T}$ and for each edge $\{\mu, \mu'\}$ in $\mathcal{T}$ there is an edge connecting H_μ and $H_{\mu'}$. In the following we describe the subgraphs H_μ for the cases that μ is an S-, P-, Q-, or R-node and define *attachment vertices* that are incident to the edges connecting H_μ to other subgraphs.

If μ is an S- or P-node, the subgraph H_μ contains only one attachment vertex v_μ and all subgraphs representing neighbors of μ are attached to v_μ. To distinguish between S- and P-nodes, small non-isomorphic subgraphs called *tags* are attached to v_μ, see Figure 5(s) and (p). If μ is a Q-node, then H_μ is a single attachment vertex, see Figure 5(q). Note that μ is a leaf in $\mathcal{T}$ and thus H_μ is also a leaf in $G_\mathcal{T}$. If μ is an R-node, H_μ is the skeleton $\mathrm{skel}(\mu)$, where additionally every virtual edge ε is subdivided by an attachment vertex v_ε, see Figure 5(r) for an example. The subgraph $H_{\mathrm{corr}(\varepsilon)}$ stemming from the neighbor $\mathrm{corr}(\varepsilon)$ of μ is attached to H_μ over the attachment vertex v_ε.

Lemma 5. *The skeleton graph is planar and can be computed in linear time.*

Lemma 6. *Two SPQR-trees are dual isomorphic if and only if their skeleton graphs are isomorphic.*

Sketch of Proof. Let φ together with $\varphi_{\mu_1}, \dots, \varphi_{\mu_k}$ be a dual SPQR-tree isomorphism between the SPQR-trees $\mathcal{T}$ and $\mathcal{T}'$. We show how this induces a graph isomorphism φ_G between the skeleton graphs $G_\mathcal{T}$ and $G_{\mathcal{T}'}$. If μ is an S-, P- or Q-node, then its corresponding subgraph in H_μ only contains a single attachment vertex v_μ. Since $\varphi(\mu)$ is of the same type (Property II), the subgraph $H_{\varphi(\mu)}$ also contains a single attachment vertex $v_{\varphi(\mu)}$ and we set $\varphi_G(v_\mu) = v_{\varphi(\mu)}$. For S- and P-nodes we additionally map their tags to each other. If μ is an R-node, the map φ_μ is a graph isomorphism between $\mathrm{skel}(\mu)$ and

$\mathrm{skel}(\varphi(\mu))$ (Property III). Thus, it induces a graph isomorphism between H_μ and $H_{\varphi(\mu)}$ since these subgraphs are obtained from $\mathrm{skel}(\mu)$ and $\mathrm{skel}(\varphi(\mu))$, respectively, by subdividing each virtual edge. Finally, φ_G respects the edges between attachment vertices of different subgraphs, since φ maps adjacent nodes to each other (Property I) and since these edges connect the correct attachment vertices of the subgraphs (Property IV).

We only sketch the opposite direction. Assume φ_G is a graph isomorphism between $G_{\mathcal{T}}$ and $G_{\mathcal{T}'}$. As bridges are mapped to bridges, we directly get an isomorphism φ between the trees $\mathcal{T}$ and $\mathcal{T}'$. As leaves have to be mapped to leaves, Q-nodes are mapped to Q-nodes. Moreover, the tags ensure that other nodes are mapped to nodes of the same type. Thus, Properties I and II are satisfied. Moreover, for every R-node μ in $\mathcal{T}$, φ_G induces an isomorphism φ_μ between μ and $\varphi(\mu)$ satisfying Properties III and IV. □

Following the outline given in Fig. 4(a) thus reduces MUTUAL PLANAR DUALITY for biconnected graphs to planarity testing for planar graph, which is linear-time solvable [8].

Theorem 7. MUTUAL PLANAR DUALITY *is linear-time solvable for biconnected graphs.*

Corollary 2. GRAPH SELF-DUALITY *is linear-time solvable for biconnected graphs.*

References

1. Angelini, P., Di Battista, G., Patrignani, M.: Finding a minimum-depth embedding of a planar graph in $O(n^4)$ time. Algorithmica 60, 890–937 (2011)
2. Angelini, P., Bläsius, T., Rutter, I.: Testing mutual duality of planar graphs. CoRR abs/1303.1640 (2013)
3. Archdeacon, D., Richter, R.B.: The construction and classification of self-dual spherical polyhedra. Journal of Combinatorial Theory, Series B 54(1), 37–63 (1992)
4. Bienstock, D., Monma, C.: On the complexity of embedding planar graphs to minimize certain distance measures. Algorithmica 5, 93–109 (1990)
5. Di Battista, G., Tamassia, R.: On-line maintenance of triconnected components with SPQR-trees. Algorithmica 15(4), 302–318 (1996)
6. Garey, M.R., Johnson, D.S.: Computers and Intractability; A Guide to the Theory of NP-Completeness. W. H. Freeman & Co., New York (1979)
7. Gutwenger, C., Mutzel, P.: A linear time implementation of SPQR-trees. In: Marks, J. (ed.) GD 2000. LNCS, vol. 1984, pp. 77–90. Springer, Heidelberg (2001)
8. Hopcroft, J.E., Wong, J.K.: Linear time algorithm for isomorphism of planar graphs (preliminary report). In: Proceedings of the 6th Annual ACM Symposium on Theory of Computing (STOC 1974), pp. 172–184. ACM (1974)
9. Raghavendra Rao, B., Jayalal Sarma, M.: On the complexity of matroid isomorphism problem. Theory of Computing Systems 49(2), 246–272 (2011)
10. Servatius, B., Christopher, P.R.: Construction of self-dual graphs. Am. Math. Monthly 99(2), 153–158 (1992)
11. Servatius, B., Servatius, H.: Self-dual graphs. Discrete Math. 149(1-3), 223–232 (1996)
12. Tutte, W.T.: Connectivity in matroids. Canad. J. Math. 18, 1301–1324 (1966)
13. Whitney, H.: Congruent graphs and the connectivity of graphs. Amer. J. Math. 54(1), 150–168 (1932)
14. Whitney, H.: 2-isomorphic graphs. Amer. J. Math. 55, 245–254 (1933)