Kernelization of Whitney Switches

Fedor V. Fomin
Department of Informatics, University of Bergen, Norway
Fedor.Fomin@uib.no

Petr A. Golovach
Department of Informatics, University of Bergen, Norway
Petr.Golovach@uib.no

Abstract

A fundamental theorem of Whitney from 1933 asserts that 2-connected graphs G and H are 2-isomorphic, or equivalently, their cycle matroids are isomorphic, if and only if G can be transformed into H by a series of operations called Whitney switches. In this paper we consider the quantitative question arising from Whitney's theorem: Given 2-isomorphic graphs, can we transform one into another by applying at most k Whitney switches? This problem is already NP-complete for cycles, and we investigate its parameterized complexity. We show that the problem admits a kernel of size $\mathcal{O}(k)$, and thus, is fixed-parameter tractable when parameterized by k.

2012 ACM Subject Classification Mathematics of computing → Graph algorithms; Theory of computation → Parameterized complexity and exact algorithms

Keywords and phrases Whitney switch, 2-isomorphism, Parameterized Complexity, kernelization

Digital Object Identifier 10.4230/LIPIcs.ESA.2020.48

Related Version The full version of the paper is available at `https://arxiv.org/abs/2006.13684`.

Funding The research leading to these results has received funding from the Research Council of Norway via the project "MULTIVAL" (grant no. 263317).

Acknowledgements We are grateful to Erlend Raa Vågset for fruitful discussions that initiated the research resulted in the paper.

1 Introduction

A fundamental result of Whitney from 1933 [35], asserts that every 2-connected graph is completely characterized, up to a series of Whitney switches (also known as 2-switches), by its edge set and cycles. This theorem is one of the cornerstones of Matroid Theory, since it provides an exact characterization of two graphs having isomorphic cycle matroids [32]. In graph drawing and graph embeddings, this theorem (applied to dual graphs) is used to characterize all drawings of a planar graph [8].

A *Whitney switch* is an operation, that from a 2-connected graph G, constructs graph G' as follows. Let $\{u, v\}$ be two vertices of G, whose removal separates G into two disjoint subgraphs G_1 and G_2. The graph G' is obtained by flipping the neighbors of u and v in the set of vertices of G_2. In other words, for every vertex $w \in V(G_2)$, if w was adjacent to u in G, in graph G' edge uw is replaced by vw. Similarly, if w was adjacent to v in G, we replace vw by uw. See Figure 1 for an example.

If we view the graph G as a graph with labelled edges, then a Whitney switch transforms G into a graph G' with the same set of labelled edges, however graphs G and G' are not necessarily isomorphic. On the other hand, graphs G and G' have the same set of cycles in the following sense: a set of (labelled) edges forms a cycle in G if and only if it forms a cycle in G'. (In other words, the cycle matroids of G and G' are isomorphic.) Whitney's

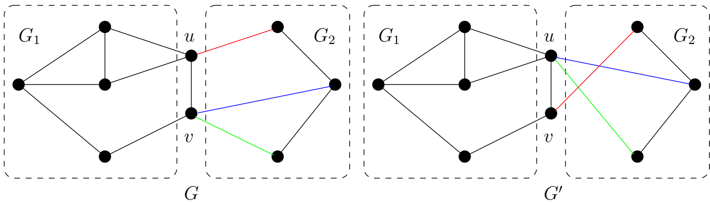

Figure 1 G' is obtained from G by the Whitney switch with respect to the partition of $G - \{u, v\}$ into G_1 and G_2.

theorem claims that the opposite is also true: if there is a cycle-preserving mapping between graphs G and G' then one graph can be transformed into another by a sequence of Whitney switches. To state the theorem of Whitney more precisely, we need to define 2-isomorphisms.

We say that 2-connected graphs G and H are 2-*isomorphic* if there is a bijection $\varphi\colon E(G) \to E(H)$ such that φ and φ^{-1} preserve cycles, that is, for every cycle C of G, C is mapped to a cycle of H by φ and, symmetrically, every cycle of H is mapped to a cycle of G by φ^{-1}. We refer to φ as to 2-*isomorphism* from G to H. An isomorphism $\psi\colon V(G) \to V(H)$ is a φ-*isomorphism* if for every edge $uv \in E(G)$, $\varphi(uv) = \psi(u)\psi(v)$, and G and H are φ-isomorphic if there is an isomorphism G to H that is a φ-isomorphism. Let us note that if G is 3-connected and 2-isomorphic to H under φ then G and H are φ-isomorphic [29, Lemma 1]. But for 2-connected graphs this is not true. For example, the graphs in Fig. 1 are not isomorphic but are 2-isomorphic. Moreover, even isomorphic graphs with 2-isomorphism φ not always have a φ-isomorphism. For example, for the 2-isomorphism φ in Fig. 2 mapping a cycle G into another cycle H (we view these cycles as labelled graphs), there is no φ-isomorphism. (For every φ-isomorphism edges $\varphi(a)$ and $\varphi(b)$ should have an endpoint in common.) On the other hand, graph G' obtained from G by Whitney switch (for vertices u and v) is φ-isomorphic to H.

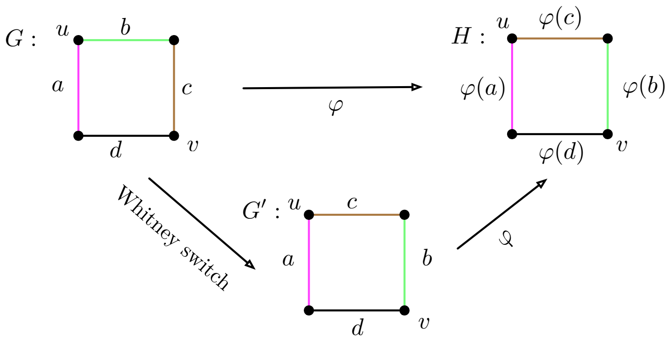

Figure 2 Graph G is not φ-isomorphic to H but its Whitney switch G' is.

▶ **Theorem 1** (Whitney's theorem [35]). *If there is a 2-isomorphism φ from graph G to graph H, then G can be transformed by a sequence of Whitney switches to a graph G' which is φ-isomorphic to H.*

However, Whitney's theorem does not provide an answer to the following question: Given a 2-isomorphism φ from graph G to graph H, *what is the minimum number of Whitney switches required to transform G to a graph φ-isomorphic to H?* Truemper in [29] proved that $n - 2$ switches always suffices, where n is the number of vertices in G. He also proved that this upper bound it tight, that is, there are graphs G and H for which $n-2$ switches are necessary. In this paper we study the algorithmic complexity of the following problem about Whitney switches.

WHITNEY SWITCHES

Input:	2-Isomorphic n-vertex graphs G and H with a 2-isomorphism $\varphi\colon E(G) \to E(H)$, and a nonnegative integer k.
Task:	Decide whether it is possible to obtain from G a graph G' that φ-isomorphic to H by at most k Whitney switches.

The departure point for our study is an easy reduction (Theorem 4) from SORTING BY REVERSALS that establishes NP-completeness of WHITNEY SWITCHES even when input graphs G and H are cycles. Our main algorithmic result is the following theorem (we postpone the definition of a kernel till Section 2). Informally, it means that the instance of the problem can be compressed in polynomial time to an equivalent instance with two graphs on $\mathcal{O}(k)$ vertices. It also implies that WHITNEY SWITCHES is fixed-parameter tractable parameterized by k.

▶ **Theorem 2.** *WHITNEY SWITCHES admits a kernel with $\mathcal{O}(k)$ vertices and is solvable in $2^{\mathcal{O}(k \log k)} \cdot n^{\mathcal{O}(1)}$ time.*

While Theorem 2 is not restricted to planar graphs, pipelined with the well-known connection of planar embeddings and Whitney switches, it can be used to obtain interesting algorithmic consequences about distance between planar embeddings of a graph. Recall that graphs G and G^* are called abstractly dual if there is a bijection $\pi\colon E(G) \to E(G^*)$ such that edge set $E \subseteq E(G)$ forms a cycle in G if and only if $\pi(E)$ is a minimal edge-cut in G^*. By another classical theorem of Whitney [34], a graph G has a dual graph if and only if G is planar. Moreover, an embedding of a planar graph into a sphere is uniquely defined by the planar graph G and edges of the faces, or equivalently, its dual graph G^*. While every 3-connected planar graph has a unique embedding into the sphere, a 2-connected graph can have several non-equivalent embeddings, and hence several non-isomorphic dual graphs. If G_1^* and G_2^* are dual graphs of graph G, then G_1^* is 2-isomorphic to G_2^*. Then by Theorem 1, by a sequence of Whitney switches G_1^* can be transformed into G_2^*, or equivalently, the embedding of G corresponding to G_1^* can be transformed to embedding of G corresponding to G_2^*. We refer to the survey of Carsten Thomassen [28, Section 2.2] for more details. By Theorem 2, we have that given two planar embeddings of a (labelled) 2-connected graph G, deciding whether one embedding can be transformed into another by making use of at most k Whitney switches, admits a kernel of size $\mathcal{O}(k)$ and is fixed-parameter tractable.

Related work. Whitney's theorem had a strong impact on the development of modern graph and matroid theories. While the original proof is long, a number of simpler proofs are known in the literature. In particular, the work of Truemper in [29], whose proof of Whitney's theorem is based on applications of Tutte decomposition [30, 31]. This is also the approach we adapt in our work.

The well-studied problem similar in spirit to WHITNEY SWITCHES is the problem of computing the flip distance for triangulations of a set of points. The parameterized complexity of this problem was studied in [10, 23]. As we already have mentioned, WHITNEY SWITCHES for planar graphs is equivalent to the problem of computing the Whitney switch distance between planar embeddings. We refer to the survey of Bose and Hurtado [4] for the discussion of the relations between geometric and graph variants. The problem is known to be NP-complete [22] and FPT parameterized by the number of flips [19]. For the special case when the set of points defines a convex polygon, the problem of computing the flip distance between triangulations is equivalent to computing the rotation distance between two binary trees. For that case linear kernels are known [10, 23] but for the general case the existence of a polynomial kernel is open.

Whitney Switches also can be seen as a reconfiguration problem, study of reconfiguration problems is a popular trend in parameterized complexity, see e.g. [24, 21].

Overview of the proof of Theorem 2. The main tool in the construction of the kernel is the classical Tutte decompositions [30, 31] We postpone the formal definition till Section 2. Informally, the Tutte decomposition of a 2-connected graph represents the vertex separators of size two in a tree-like structure. Each node of this tree represents a part of the graph (or *bag*) that is either a 3-connected graph or a cycle, and each edge corresponds to a separator of size two. Then a 2-isomorphism of G and H allows to establish an isomorphism of the trees representing the Tutte decompositions of the input graphs. After that, potential Whitney switches can be divided into two types: the switches with respect to separators corresponding to the edges of the trees and the switches with respect to separators formed by nonadjacent vertices of a cycle-bag. The switches of the first type are relatively easy to analyze and we can identify necessary switches of this type. The "troublemakers" that make the problem hard are switches of the second type. To deal with them, we use the structural results about sorting of permutations by reversals of Hannenhalli and Pevzner [17] adapted for our purposes. This allows us to identify a set of vertices of size $\mathcal{O}(k)$ that potentially can be used for Whitney switches transforming G to H. Given such a set of crucial vertices, we simplify the structure of the input graphs and then reduce their size.

Organization of the paper. In Section 2, we give basic definitions. In Section 3, we discuss the Sorting by Reversals problem for permutations that is closely related to Whitney Switches. Section 4 contains structural results used by our kernelization algorithm, and in Section 5, we give the algorithm itself. We conclude in Section 6 by discussing further directions of research. Due to space constraints, the proofs are either omitted or just sketched. The details are given in the full version of the paper [15].

2 Preliminaries

Graphs. All graphs considered in this paper are finite undirected graphs without loops or multiple edges, unless it is specified explicitly that we consider directed graphs (in Section 6 we deal with tournaments). We follow the standard graph theoretic notation and terminology (see, e.g., [13]). For each of the graph problems considered in this paper, we let $n = |V(G)|$ and $m = |E(G)|$ denote the number of vertices and edges, respectively, of the input graph G if it does not create confusion. A pair (A, B), where $A, B \subseteq V(G)$, is a *separation* of G if $A \cup B = V(G)$, $A \setminus B \neq \emptyset$, $B \setminus A \neq \emptyset$ and G has no edge uv with $u \in A \setminus B$ and $v \in B \setminus A$; $|A \cap B|$ is the *order* of the separation. If the order is 2, then we say that (A, B) is a *Whitney separation*. A set $S \subseteq V(G)$ is a *separator* of G if there is a separation (A, B) of G with $S = A \cap B$.

Whitney switches. It is convenient to define Whitney switches using separations. Let G be a 2-connected graph. Let also (A, B) be a Whitney separation of G with $A \cap B = \{u, v\}$. The *Whitney switch* operation with respect to (A, B) transforms G as follows: take $G[A]$ and $G[B]$ and identify the vertex u of $G[A]$ with the vertex v of $G[B]$ and, symmetrically, v of $G[A]$ with

u of $G[B]$; if u and v are adjacent in G, then the edges uv of $G[A]$ and $G[B]$ are identified as well. Let G' be the obtained graph. We define the mapping $\sigma_{(A,B)}\colon E(G) \to E(G')$ that maps the edges of $G[A]$ and $G[B]$, respectively, to themselves. It is easy to see that $\sigma_{(A,B)}$ is a 2-isomorphism of G to G'. Therefore, if φ is a 2-isomorphism of G to a graph H, then $\varphi \circ \sigma^{-1}_{(A,B)}$ is a 2-isomorphism of G' to H. To simplify notation, we assume, if it does not create confusion, that the sets of edges of G and G' are identical and we only change incidences by switching. In particular, under this assumption, we have that $\varphi \circ \sigma^{-1}_{(A,B)} = \varphi$. We also assume that the graphs G and G' have the same sets of vertices.

Tutte decomposition. Our kernelization algorithm for WHITNEY SWITCHES is based on the classical result of Tutte [30, 31] about decomposing of 2-connected graphs via separators of size two. Following Courcelle [11], we define Tutte decompositions in the terms of tree decompositions.

A *tree decomposition* of a graph G is a pair $\mathcal{T} = (T, \{X_t\}_{t\in V(T)})$, where T is a tree whose every node t is assigned a vertex subset $X_t \subseteq V(G)$, called a bag, such that the following three conditions hold:

(T1) $\bigcup_{t\in V(T)} X_t = V(G)$, that is, every vertex of G is in at least one bag,
(T2) for every $uv \in E(G)$, there exists a node t of T such that the bag X_t contains both u and v,
(T3) for every $v \in V(G)$, the set $T_v = \{t \in V(T) \mid v \in X_t\}$, i.e., the set of nodes whose corresponding bags contain v, induces a connected subtree of T.

To distinguish between the vertices of the decomposition tree T and the vertices of the graph G, we will refer to the vertices of T as *nodes*.

Let $\mathcal{T} = (T, \{X_t\}_{t\in V(T)})$ be a tree decomposition of G. The *torso* of X_t for $t \in V(T)$ is the graph obtained from $G[X_t]$ by additionally making adjacent every two distinct vertices $u, v \in X_t$ such that there is $t' \in V(T)$ adjacent to t with $u, v \in X_t \cap X_{t'}$. For adjacent $t, t' \in V(T)$, $X_t \cap X_{t'}$ is the *adhesion set* of the bags X_t and $X_{t'}$ and $|X_t \cap X_{t'}|$ is the *adhesion* of the bags. The maximum adhesion of adjacent bags is called the *adhesion* of the tree decomposition.

Let G be a 2-connected graph. A tree decomposition $\mathcal{T} = (T, \{X_t\}_{t\in V(T)})$ is said to be a *Tutte decomposition* if $\mathcal{T}$ is a tree decomposition of adhesion 2 such that there is a partition $(W_2, W_{\geq 3})$ of $V(T)$ such that the following holds:

(T4) $|X_t| = 2$ for $t \in W_2$ and $|X_t| \geq 3$ for $t \in W_{\geq 3}$,
(T5) the torso of X_t is either a 3-connected graph or a cycle for every $t \in W_{\geq 3}$,
(T6) for every $t \in W_2$, $d_T(t) \geq 2$ and $t' \in W_{\geq 3}$ for each neighbor t' of t,
(T7) for every $t \in W_{\geq 3}$, $t' \in W_2$ for each neighbor t' of t,
(T8) if $t \in W_2$ and $d_T(t) = 2$, then for the neighbors t' and t'' of t, either the torso of t' or the torso of t'' is a 3-connected graph or the vertices of X_t are adjacent in G.

Notice that the bags X_t for $t \in W_2$ are distinct separators of G of size two, and $X_t \subseteq X_{t'}$ for $t \in W_2$ and $t' \in N_T(t)$. Observe also that if $\{u, v\}$ is a separator of G of size two, then either $\{u, v\} = X_t$ for some $t \in W_2$ or $u, v \in X_t$ for $t \in W_{\geq 3}$ such that the torso of X_t is a cycle and u and v are nonadjacent vertices of the torso.

Combining the results of Tutte [30, 31] and of Hopcroft and Tarjan [18], we state the following proposition.

▶ **Proposition 3** ([30, 31, 18]). *A 2-connected graph G has a unique Tutte decomposition that can be constructed in linear time.*

Parameterized Complexity and Kernelization. We refer to the books [12, 14, 16] for the detailed introduction to the field. Here we only give the most basic definitions. In the Parameterized Complexity theorey, the computational complexity is measured as a function of the input size n of a problem and an integer *parameter* k associated with the input. A parameterized problem is said to be *fixed parameter tractable* (or FPT) if it can be solved in time $f(k) \cdot n^{\mathcal{O}(1)}$ for some function f. A *kernelization* algorithm for a parameterized problem Π is a polynomial algorithm that maps each instance (I, k) of Π to an instance (I', k') of Π such that (i) (I, k) is a yes-instance of Π if and only if (I', k') is a yes-instance of Π, and (ii) $|I'| + k'$ is bounded by $f(k)$ for a computable function f. Respectively, (I', k') is a *kernel* and f is its *size*. A kernel is *polynomial* if f is polynomial. It is common to present a kernelization algorithm as a series of *reduction rules*. A reduction rule for a parameterized problem is an algorithm that takes an instance of the problem and computes in polynomial time another instance that is more "simple" in a certain way. A reduction rule is *safe* if the computed instance is equivalent to the input instance.

3 Sorting by reversals

Sorting by reversals is the classical problem with many applications including bioinformatics. We refer to the book of Pevzner [25] for the detailed survey of results and applications of this problem. This problem is also strongly related to WHITNEY SWITCHES– solving the problem for two cycles is basically the same as sorting circular permutations by reversals. First we use this relation to observe the NP-completeness. But we also need to establish some structural properties of sorting by reversals which will be used in kernelization algorithm.

Let $\pi = (\pi_1, \ldots, \pi_n)$ be a permutation of $\{1, \ldots, n\}$, that is, a bijective mapping of $\{1, \ldots, n\}$ to itself. Throughout this section, all considered permutations are permutations of $\{1, \ldots, n\}$. For $1 \leq i \leq j \leq n$, the *reversal* $\rho(i, j)$ reverse the order of elements $\pi_i, \ldots, \pi_j$ and transforms π into

$$\rho(i, j) \circ \pi = (\pi_1, \ldots, \pi_{i-1}, \pi_j, \pi_{j-1}, \ldots, \pi_i, \pi_{j+1}, \ldots, \pi_n).$$

The *reversal distance* $d(\pi, \sigma)$ between two permutations π and σ is the minimum number of reversals needed to transform π to σ. For a permutation π, $d(\pi) = d(\pi, \iota)$, where ι is the identity permutation; note that $d(\pi, \sigma) = d(\sigma^{-1} \circ \pi, \iota)$ and this means that computing the reversal distance can be reduced to sorting a permutation by the minimum number of reversals.

These definitions can be extended for circular permutations (further, we may refer to usual permutations as *linear* to avoid confusion). We say that $\pi^c = (\pi_1, \ldots, \pi_n)$ is a *circular* permutation if π^c is the class of the permutations that can be obtained from the linear permutation $(\pi_1, \ldots, \pi_n)$ by *rotations* and *reflections*, that is, all the permutations

$$(\pi_1, \ldots, \pi_n), (\pi_n, \pi_1, \ldots, \pi_{n-1}), \ldots, (\pi_2, \ldots, \pi_n, \pi_1),$$

and

$$(\pi_n, \ldots, \pi_1), (\pi_1, \pi_n, \ldots, \pi_2), \ldots, (\pi_{n-1}, \ldots, \pi_1, \pi_n)$$

composing one class are identified, meaning that we do not distinguish them when discussing circular permutations. The *circular reversals* $\rho^c(i, j)$ and *circular reversal distance* $d^c(\pi^c, \sigma^c)$ and $d^c(\pi^c)$ are defined in the same way as for linear permutations.

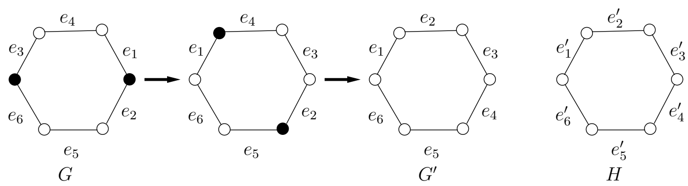

Figure 3 The construction of G' that is φ-isomorphic to H by the Whitney switches corresponding to the sorting by reversals $(3,4,1,2,5,6) \to (1,4,3,2,5,6) \to (1,2,3,4,5,6)$; $\varphi(e_i) = e'_i$ for $i \in \{1,\ldots,6\}$, the vertices of the separators for the switches are shown in black.

To see the connection between Whitney switches and circular reversals of permutations, consider a cycle G with the vertices $v_1,\ldots,v_n$ for $n \geq 4$ taken in the cycle order and the edges $e_i = v_{i-1}v_i$ for $i \in \{1,\ldots,r\}$ assuming that $v_0 = v_n$. Let $1 \leq i < j \leq n$ be such that v_i and v_j are not adjacent. Then the Whitney switch with respect to (A,B), where $A = \{v_1,\ldots,v_i\} \cup \{v_j,\ldots,v_n\}$ and $B = \{v_i,\ldots,v_j\}$ is equivalent to applying the reversal $\rho^c(i+1,j)$ to the circular permutation $(e_1,\ldots,e_n)$ of the edges of G. Moreover, let H be a cycle with n vertices and denote by $e'_1,\ldots,e'_n$ its edges in the cycle order. Notice that every bijection $\varphi\colon E(G) \to E(H)$ is a 2-isomorphism of G to H, and G and H are φ-isomorphic if and only if the circular permutation $\pi^c = (\varphi^{-1}(e'_1),\ldots,\varphi^{-1}(e'_n))$ is the same as $\sigma^c = (e_1,\ldots,e_n)$. Clearly, we can assume that π^c is a permutation of $\{1,\ldots,n\}$ and σ^c is the identity permutation. Then G can be transformed to a graph G' φ-isomorphic to H by at most k Whitney switches if and only if $d^c(\pi^c) \leq k$. An example is shown in Fig. 3.

In particular, the above observation implies the hardness of Whitney Switches, because the computing of the reversal distances is known to be NP-hard. For linear permutations, this was shown by Caprara in [7]. Then Solomon, Sutcliffe, and Lister [27] proved that it is NP-complete to decide, given a circular permutation π^c and a nonnegative integer k, whether $d^c(\pi^c) \leq k$. This brings us to the following result.

▶ **Theorem 4.** *Whitney Switches is NP-complete even when restricted to cycles.*

For our kernelization algorithm, we need some further structural results about reversals in an optimal sorting sequence.

Let $\pi = (\pi_1,\ldots,\pi_n)$ be a linear permutation. For $1 \leq i \leq j \leq n$, we say that $(\pi_i,\ldots,\pi_j)$ is an *interval* of π. An interval $(\pi_i,\ldots,\pi_j)$ is called a *block* if either $i = j$ or $i < j$ and for every $h \in \{i+1,\ldots,j\}$, $|\pi_{h-1} - \pi_h| = 1$, that is, a block is formed by consecutive integers in π in either the ascending or descending order. An inclusion maximal block is called a *strip*. In other words, a strip is an inclusion maximal interval that has no *breakpoint*, that is, a pair of elements π_{h-1},π_h with $|\pi_{h-1} - \pi_h| \geq 2$. It is said that a reversal $\rho(p,q)$ *cuts* a strip $(\pi_i,\ldots,\pi_j)$ if either $i < p \leq j$ or $i \leq q < j$, that is, the reversals separates elements that are consecutive in the identity permutation.

It is known that there are cases when every optimal sorting by reversal requires a reversal that cuts a strip. For example, as was pointed by Hannenhalli and Pevzner in [17], the permutation $(3,4,1,2)$ requires three reversals that do not cut strips, but the sorting can be done by two reversals: [1]

$$(3,4,1,2) \to (1,4,3,2) \to (1,2,3,4).$$

[1] This example can be extended for circular permutations: $(3,4,1,2,5,6) \to (1,4,3,2,5,6) \to (1,2,3,4,5,6)$.

However, it was conjectured by Kececioglu and Sankoff [20] that there is an optimal sorting that does not cut strips other than at their first or last elements. This conjecture was proved by Hannenhalli and Pevzner in [17]. More precisely, they proved that there is an optimal sorting that does not cut strips of length at least three.

It is common for bioinformatics applications, to consider *signed permutations* (see, e.g., [25]). In a signed permutation $\overrightarrow{\pi} = (\pi_1, \ldots, \pi_n)$, each element π_i has its *sign* "$-$" or "$+$". Then for $i, j \in \{1, \ldots, n\}$, the reversal reverse the sign of each element $\pi_i, \ldots, \pi_j$ besides reversing their order. We generalize this notion and define *partially signed circular permutations*, where each element has one of the sings: "$-$", "$+$" or "no sign". Formally, a *partially signed circular permutation* $\overrightarrow{\pi}^c = (\langle \pi_1, s_1\rangle, \ldots, \langle \pi_n, s_n\rangle)$, where $(\pi_1, \ldots, \pi_n)$ is a linear permutation and $s_i \in \{-1, +1, 0\}$ for $i \in \{1, \ldots, m\}$, as the class of the linear permutations that can be obtained from $(\langle \pi_1, s_1\rangle, \ldots, \langle \pi_n, s_n\rangle)$ by rotations and reflections such that every reflection reverse signs. For $i, j \in \{1, \ldots, n\}$, the reversal

$$\overrightarrow{\rho}^c(i,j) \circ \overrightarrow{\pi}^c = (\langle \pi_1, s_1\rangle, \ldots, \langle \pi_{i-1}, s_{i-1}\rangle, \langle \pi_j, -s_j\rangle, \ldots, \langle \pi_i, -s_i\rangle, \langle \pi_{j+1}, s_{j+1}\rangle \ldots, \langle \pi_n, s_n\rangle)$$

if $i \leq j$, and

$$\overrightarrow{\rho}^c(i,j) \circ \overrightarrow{\pi}^c = (\langle \pi_n, -s_n\rangle, \ldots, \langle \pi_i, -s_i\rangle, \langle \pi_{j+1}, s_{j+1}\rangle, \ldots, \langle \pi_{i-1}, s_{i-1}\rangle, \langle \pi_j, -s_j\rangle \ldots, \langle \pi_1, -s_1\rangle)$$

otherwise.

We say that $\overrightarrow{\pi}^c$ is *signed* if each s_i is either -1 or $+1$ and the *signed circular identity permutation* is $\overrightarrow{\iota}^c = (\langle 1, +1\rangle, \ldots, \langle n, +1\rangle)$. Also a partially signed circular permutation $\overrightarrow{\pi}^c = (\langle \pi_1, s_1\rangle, \ldots, \langle \pi_n, s_n\rangle)$ *agrees in signs* with a signed circular permutation $\overrightarrow{\pi}'^c = (\langle \pi_1, s'_1\rangle, \ldots, \langle \pi_n, s'_n\rangle)$ if $s_i = s'_i$ for $i \in \{1, \ldots, n\}$ such that $s_i \neq 0$, that is, the zero signs are replaced by either -1 or $+1$ in the signed permutation, and $\Sigma(\overrightarrow{\pi}^c)$ is used to denote the set of all signed circular permutations $\overrightarrow{\pi}'^c$ that agree in signs with $\overrightarrow{\pi}^c$. Then *reversal distance* $\overrightarrow{d}^c(\overrightarrow{\pi}^c, \sigma^c)$, where $\overrightarrow{\sigma}^c$ is a signed circular permutation, is the minimum number or reversal needed to obtain from $\overrightarrow{\pi}^c$ a partially signed circular permutation $\overrightarrow{\pi}'^c$ that agrees in signs with $\overrightarrow{\sigma}^c$, and $\overrightarrow{d}^c(\overrightarrow{\pi}^c) = \overrightarrow{d}^c(\overrightarrow{\pi}^c, \overrightarrow{\iota}^c)$. A a sequence of reversals of minimum length that result in a partially signed circular permutation that agrees in signs with $\overrightarrow{\iota}^c$ is an *optimal sorting* sequence.

Let $\overrightarrow{\pi}^c = (\langle \pi_1, s_1\rangle, \ldots, \langle \pi_n, s_n\rangle)$ be a partially signed circular permutation. For $1 \leq i \leq j \leq n$, we say that $(\langle \pi_i, s_i\rangle, \ldots, \langle \pi_j, s_j\rangle)$ and $(\langle \pi_{j+1}, s_{j+1}\rangle, \ldots, \langle \pi_n, s_n\rangle, \langle \pi_1, s_1\rangle, \ldots, \langle \pi_i, s_i\rangle)$ are *intervals* of $\overrightarrow{\pi}^c$. An interval is a *signed block* if it either has size one or for every two consecutive elements $\langle \pi_{i-1}, s_{i-1}\rangle, \langle \pi_i, s_i\rangle$, $|\pi_{i-1} - \pi_i| \leq 1$ and, moreover, if the elements of the interval are in the increasing order, then all the signs $s_i \in \{0, +1\}$, and if they are in the the decreasing order, then all the signs $s_i \in \{0, -1\}$. A *signed strip* is an inclusion maximal signed block. A reversal $\overrightarrow{\rho}^c(p, q)$ *cuts* an interval if the reversed part includes at least one element of the interval and excludes at least one element of the interval. We use the result of Hannenhalli and Pevzner [17] to show the following lemma.

▶ **Lemma 5.** *For a signed circular permutation $\overrightarrow{\pi}^c$, there is an optimal sorting sequence such that no reversal in the sequence cuts the interval formed by a signed strip of $\overrightarrow{\pi}^c$ of length at least* 5.

Notice that we do not claim that no reversal cuts a strip of length at least 5 that is obtained by performing the previous reversals; only the long strips of the initial permutation $\overrightarrow{\pi}^c$ are not cut by any reversal in the sorting sequence.

4 Tutte decomposition and 2-isomorphisms

In this section we provide a number of auxiliary results about 2-isomorphisms and Tutte decompositions.

We need the following folklore observation about φ-isomorphisms that we prove for completeness. For this, we extend φ on sets of edges in standard way, that is, $\varphi(A) = \{\varphi(e) \mid e \in A\}$ and $\varphi(\emptyset) = \emptyset$.

▶ **Lemma 6.** *Let G and H be n-vertex 2-connected 2-isomorphic graphs with a 2-isomorphism φ. Then G and H are φ-isomorphic if and only if there is a bijective mapping $\psi\colon V(G) \to V(H)$ such that for every $v \in V(G)$, $\varphi(E_G(v)) = E_H(\psi(v))$. Moreover, G and H are φ-isomorphic if and only if φ bijectively maps the family of the sets of edges $\{E_G(v) \mid v \in V(G)\}$ to the family $\{E_H(v) \mid v \in V(H)\}$, and this property can be checked in polynomial time.*

By Lemma 6, we can restate the task of Whitney Switches and ask whether it is possible to obtain a graph G' by performing at most k Whitney switches starting from G with the property that the extension of φ to the family of sets $\{E_{G'}(v) \mid v \in V(G')\}$ bijectively maps this family to $\{E_H(v) \mid v \in V(H)\}$.

We use Whitney's theorem [35](see also [29]).

▶ **Proposition 7** ([35])**.** *Let G and H be n-vertex graphs and let φ be a 2-isomorphism of G to H. Then there is a finite sequence of Whitney switches such that the graph G' obtained from G by these switches is φ-isomorphic to H.*

We also use the property of 3-connected graphs explicitly given by Truemper [29]. It also can be derived from Proposition 7.

▶ **Proposition 8** ([29])**.** *Let G and H be 3-connected n-vertex graphs and let φ be a 2-isomorphism of G to H. Then G and H are φ-isomorphic.*

Throughout this section we assume that G and H are n-vertex 2-connected graphs and let φ be a 2-isomorphism of G to H. Let also $\mathcal{T}^{(1)} = (T^{(1)}, \{X_t^{(1)}\}_{t\in V(T^{(1)})})$ and $\mathcal{T}^{(2)} = (T^{(2)}, \{X_t^{(2)}\}_{t\in V(T^{(2)})})$ be the Tutte decompositions of G and H, respectively, and denote by $(W_2^{(h)}, W_{\geq 3}^{(h)})$ the partition of $V(T^{(h)})$ satisfying (T4)–(T8) for $h = 1, 2$. We use Lemma 6 and Propositions 7 and 8 to show the following lemmata.

▶ **Lemma 9.** *There is an isomorphism α of $T^{(1)}$ to $T^{(2)}$ such that*

(i) *for every $t \in V(T^{(1)})$, $|X_t^{(1)}| = |X_{\alpha(t)}^{(2)}|$, in particular, $t \in W_2^{(1)}$ ($t \in W_{\geq 3}^{(1)}$, respectively) if and only if $\alpha(t) \in W_2^{(2)}$ ($\alpha(t) \in W_{\geq 3}^{(2)}$, respectively),*

(ii) *for every $t \in W_{\geq 3}^{(1)}$, the torso of $X_t^{(1)}$ is a 3-connected graph (a cycle, respectively) if and only if the torso of $X_{\alpha(t)}^{(2)}$ is a 3-connected graph (a cycle, respectively),*

(iii) *for every $t \in V(T^{(1)})$, $\varphi(E(G[X_t^{(1)}]) = E(H[X_{\alpha(t)}^{(2)}])$.*

Let F be a 2-connected graph. Let also $\mathcal{T} = (T, \{X_t\}_{t\in V(T)})$ be the Tutte decomposition of F and let $(W_2, W_{\geq 3})$ be the partition of $V(T)$ satisfying (T4)–(T8). We denote by $\widehat{F}$ the graph obtained from F by making the vertices of X_t adjacent for every $t \in W_2$. We say that $\widehat{F}$ is the *enhancement* of F. Note that $\mathcal{T}$ is the Tutte decomposition of $\widehat{F}$ and the torso of each bag X_t is $\widehat{F}[X_t]$. Notice also that (A, B) is a Whitney separation of F if and only if (A, B) is a Whitney separation of $\widehat{F}$. We also say that F is *enhanced* if $F = \widehat{F}$.

To simplify the arguments in our proofs, it is convenient for us to switch from 2-isomorphisms of graphs to 2-isomorphisms of their enhancements. By Lemma 9, there is an isomorphism α of $T^{(1)}$ to $T^{(2)}$ satisfying conditions (i) – (ii) of the lemma. We define the *enhanced* mapping $\widehat{\varphi}\colon E(\widehat{G}) \to E(\widehat{H})$ such that $\widehat{\varphi}(e) = \varphi(e)$ for $e \in E(G)$, and for each $e \in E(\widehat{G}) \setminus E(G)$ with its end-vertices in $X_t^{(1)}$ for some $t \in W_2^{(1)}$, we define $\widehat{\varphi}(e)$ be the edge with the end-vertices in $X_{\alpha(t)}^{(2)}$.

▶ **Lemma 10.** *The mapping $\widehat{\varphi}$ is a 2-isomorphism of $\widehat{G}$ to $\widehat{H}$. Moreover, a sequence of Whitney switches makes G φ-isomorphic to H if and only if the same sequence makes $\widehat{G}$ $\widehat{\varphi}$-isomorphic to $\widehat{H}$.*

Lemma 10 allows us to consider enhanced graph and this is useful, because we can strengthen the claim of Lemma 9.

▶ **Lemma 11.** *Let G and H be enhanced graphs. Then there is an isomorphism α of $T^{(1)}$ to $T^{(2)}$ such that conditions (i)–(iii) of Lemma 9 are fulfilled and, moreover,*
(iv) *for every $t \in V(T^{(1)})$, $G[X_t^{(1)}]$ is isomorphic to $H[X_{\alpha(t)}^{(2)}]$.*
Moreover, if $G[X_t^{(1)}]$ is 3-connected, then $G[X_t^{(1)}]$ is φ-isomorphic to $H[X_{\alpha(t)}^{(2)}]$.

For the remaining part of the sections, we assume that G and H are enhanced graphs and α is the isomorphism of $T^{(1)}$ to $T^{(2)}$ satisfying conditions (i)–(iv) of Lemmas 9 and 11.

Our next aim is to investigate properties of the sequences of Whitney switches that are used in solutions for Whitney Switches. For a sequence $\mathcal{S}$ of Whitney switches such that the graph G' obtained from G by applying this sequence is ϕ-isomorphic to H, we say that $\mathcal{S}$ is an *H-sequence*. We also say that $\mathcal{S}$ is *minimum* if $\mathcal{S}$ has minimum length.

For $t \in W_{\geq 3}^{(1)}$, we say that $X_t^{(1)}$ is *φ-good* if $G[X_t^{(1)}]$ is φ-isomorphic to $H[X_{\alpha(t)}^{(2)}]$, and $X_t^{(1)}$ is *φ-bad* otherwise. Notice that if $G[X_t^{(1)}]$ is 3-connected, then $X_t^{(1)}$ is φ-good but this not always so if $G[X_t^{(1)}]$ is a cycle.

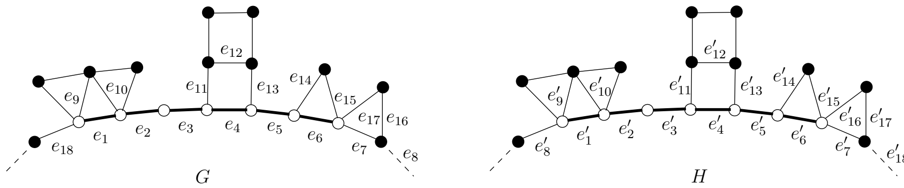

Figure 4 An example of a φ-good segment; $\varphi(e_i) = e'_i$ for $i \in \{1, \ldots, 18\}$, the vertices of the segment are white.

Let $t \in W_{\geq 3}^{(1)}$ such that $X_t^{(1)}$ is φ-bad. Clearly, $G[X_t^{(1)}]$ is a cycle. Let $\{t_1, \ldots, t_s\} = N_{T^{(1)}}^2(t)$ and denote $G_t = G[X_t^{(1)} \cup \bigcup_{i=1}^s X_{t_i}^{(1)}]$ and $H_{\alpha(t)} = H[X_{\alpha(t)}^{(2)} \cup \bigcup_{i=1}^s X_{\alpha(t_i)}^{(2)}]$. Let $P = v_0 \cdots v_r$ be a path in $G[X_t^{(1)}]$ and $e_i = v_{i-1}v_i$ for $i \in \{1, \ldots, r\}$. We say that P a *φ-good segment* of $X_t^{(1)}$ if the following holds (see Fig. 4 for an example):
(i) the length of P is at least 5,
(ii) there is a path $P' = u_0 \cdots u_r$ in $H[X_{\alpha(t)}^{(2)}]$ such that $u_{i-1}u_i = \varphi(e_i)$ for all $i \in \{1, \ldots, r\}$,
(iii) for every $i \in \{1, \ldots, r\}$ and for every $t' \in W_{\geq 3}^{(1)}$ such that $X_t^{(1)} \cap X_{t'}^{(1)} = \{v_{i-1}, v_i\}$, $X_{t'}^{(1)}$ is φ-good,
(iv) for every $i \in \{1, \ldots, r-1\}$, $\varphi(E_{G_t}(v_i)) = E_{H_{\alpha(t)}}(u_i)$.

For distinct $t_1, t_2 \in W_{\geq 3}^{(1)}$ with a common neighbor in $T^{(1)}$, we say that $X_{t_1}^{(1)}$ and $X_{t_2}^{(1)}$ are *mutually φ-good* (see Fig. 5) if they are φ-good and $G[X_{t_1}^{(1)} \cup X_{t_2}^{(1)}]$ is φ-isomorphic to $H[X_{\alpha(t_1)}^{(2)} \cup X_{\alpha(t_2)}^{(2)}]$.

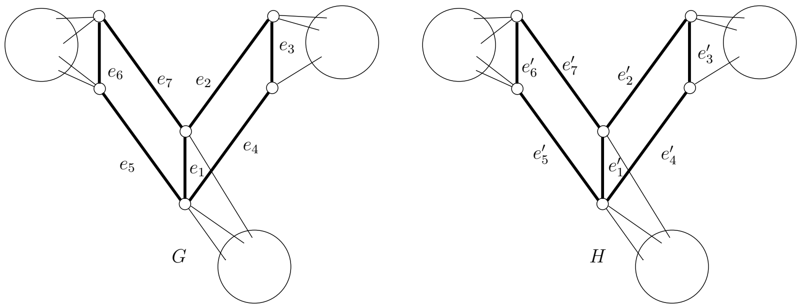

Figure 5 Mutually φ-good bags; $\varphi(e_i) = e'_i$ for $i \in \{1, \dots, 7\}$, the vertices of the mutually φ-good bags of G and the corresponding bags of H are white.

We say that an H-sequence is *φ-good* if no Whitney switch of $\mathcal{S}$ splits (mutually) φ-good bags and segments. Formally, for every switch with respect to some Whitney separation (A, B) in $\mathcal{S}$,

(i) $X_t^{(1)} \subseteq A$ or $X_t^{(1)} \subseteq B$ for every φ-good bag $X_t^{(1)}$,

(ii) $V(P) \subseteq A$ or $V(P) \subseteq B$ for every φ-good segment P,

(iii) $X_{t_1}^{(1)} \cup X_{t_2}^{(1)} \subseteq A$ or $X_{t_1}^{(1)} \cup X_{t_2}^{(1)} \subseteq B$ for every two distinct mutually φ-good bags $X_{t_1}^{(1)}$ and $X_{t_2}^{(1)}$.

We prove that it is sufficient to consider φ-good H-sequences. The crucial tool for our kernelization algorithm is the following lemma that is proved by using Lemmata 5 and 11. In particular, Lemma 5 is crucial for showing that we can avoid splitting φ-good segments.

▶ **Lemma 12.** *There is a minimum H-sequence of Whitney switches $\mathcal{S}$ that is φ-good.*

Let $t \in W_{\geq 3}^{(1)}$ be such that $X_t^{(1)}$ is φ-bad. Denote by $t_1, \dots, t_s \neq t$ the nodes of $N^2_{T^{(1)}}(t)$. Let $G_t = G[X_t^{(1)} \cup \bigcup_{i=1}^{s} X_{t_i}^{(1)}]$ and $H_{\alpha(t)} = G[X_{\alpha(t)}^{(2)} \cup \bigcup_{i=1}^{s} X_{\alpha(t_i)}^{(2)}]$. In words, G_t is the subgraphs of G induced by the vertices of $X_t^{(1)}$ and the vertices of the bags at distance two in $T^{(1)}$ from t, and $H_{\alpha(t)}$ the subgraph of H induced by the vertices of the bags that are images of the bags composing G_t according to α.

We say that a vertex $v \in X_t^{(1)}$ is a *crucial breakpoint* if $\varphi(E_{G_t}(v)) \neq E_{H_{\alpha(t)}}(u)$ for every $u \in V(H_{\alpha(t)})$. We denote by $b(G)$ the total number of crucial breakpoints in the φ-bad bags and say that $b(G)$ is the *breakpoint number* of G. Recall that by our convention, G and H are enhanced graphs, but we extend this definition for the general case needed in the next section. For (not necessarily enhanced) 2-isomorphic graphs G and H, and a 2-isomorphism φ, we construct their enhancements $\widehat{G}$ and $\widehat{H}$, and consider the enhanced mapping $\widehat{\varphi}$. Then $b(G)$ is defined as $b(\widehat{G})$.

Observe that if G and H are φ-isomorphic, then $b(t) = 0$ by Lemma 6, but not the other way around.

We conclude the section by giving a lower bound for the length of an H-sequence.

▶ **Lemma 13.** *Let $\mathcal{S}$ be an H-sequence of Whitney switches. Then $b(G)/2 \leq |\mathcal{S}|$.*

5 Kernelization for Whitney Switches

In this section, we show that Whitney Switches parameterized by k admits a polynomial kernel. To do it, we obtain a more general result by proving that the problem has a polynomial kernel when parameterized by the breakpoint number of the first input graph.

▶ **Theorem 14.** *WHITNEY SWITCHES has a kernel such that each graph in the obtained instance has at most* $\max\{52 \cdot b - 36, 3\}$ *vertices, where b is the breakpoint number of the input graph.*

Proof sketch. Let (G, H, φ, k) be an instance of WHITNEY SWITCHES, where G and H are n-vertex 2-connected 2-isomorphic graphs, $\varphi\colon E(G) \to E(H)$ is a 2-isomorphism, and k is a nonnegative integer.

First, we use Proposition 3 to construct the Tutte decompositions of G and H. Denote by $\mathcal{T}^{(1)} = (T^{(1)}, \{X_t^{(1)}\}_{t\in V(T^{(1)})})$ and $\mathcal{T}^{(2)} = (T^{(2)}, \{X_t^{(2)}\}_{t\in V(T^{(2)})})$ the constructed Tutte decompositions of G and H respectively, and let $(W_2^{(h)}, W_{\geq 3}^{(h)})$ be the partition of $V(T^{(h)})$ satisfying (T4)–(T8) for $h = 1, 2$.

In the next step, we construct the isomorphism $\alpha\colon V(T^{(1)}) \to V(T^{(2)})$ satisfying conditions (i)–(iii) of Lemma 9. Recall that Lemma 9 claims that such an isomorphism always exists.

Given α, we compute the enhancements $\widehat{G}$ and $\widehat{H}$ of G and H respectively, and then define the enhanced mapping $\widehat{\varphi}\colon E(\widehat{G}) \to E(\widehat{H})$. Note that α satisfies the conditions of Lemma 11. Observe also that we can verify in polynomial time whether a bag $X_t^{(1)}$ for $t \in W_{\geq 3}^{(1)}$ is φ-good or not.

To simplify notation, let $G := \widehat{G}$, $H := \widehat{H}$ and $\varphi := \hat{\varphi}$.

Now we apply a series of reduction rules that are applied for G, H, φ, and the Tutte decompositions of G and H.

The aim of the first rule is to decrease the total size of bags that are φ-bad (see Fig. 6 for an example).

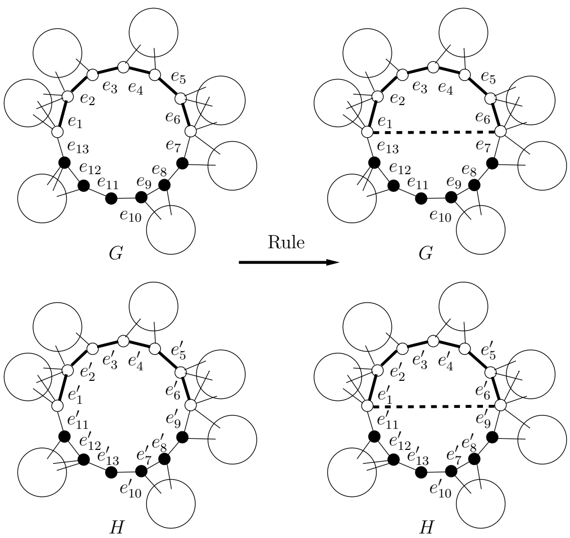

Figure 6 An example of an application of Reduction Rule 1; $\varphi(e_i) = e_i'$ for $i \in \{1, \ldots, 13\}$, the vertices of the φ-good segment in G and the corresponding segment in H are white, and the added edges are shown by dashed lines.

▶ **Reduction Rule 1.** *If for* $t \in W_{\geq 3}^{(1)}$ *such that* $X_t^{(1)}$ *is* φ*-bad, there is an inclusion maximal* φ*-good segment* $P = v_0 \cdots v_r$*, then do the following:*

- *find the path* $P' = u_0 \cdots u_r$ *in* $H[X^{(2)}_{\alpha(t)}]$ *composed by the edges* $u_{i-1}u_i = \varphi(v_{i-1}v_i)$ *for* $i \in \{1, \ldots, r\}$*,*
- *add the edge* v_0v_r *to G and* u_0u_r *to H,*
- *extend* φ *by setting* $\varphi(v_0u_r) = u_0u_r$*,*
- *recompute the Tutte decompositions of the obtained graphs and the isomorphism* α*.*

The safeness of the rule is proved by using Lemma 12. The crucial observation is that there is an optimal sequence of Whitney switches such that every φ-good segment remains in one part of every Whitney separation in the sequence, i.e., they are not split. Reduction Rule 1 is applied exhaustively while we are able to find φ-good segments. To simplify notation, we use G, H and φ to denote the obtained graphs and the obtained 2-isomorphism. We also keep the notation used for the Tutte decompositions.

Our next reduction rule is used to simplify the structure of φ-good bags by turning them into cliques (see Fig. 7 for an example).

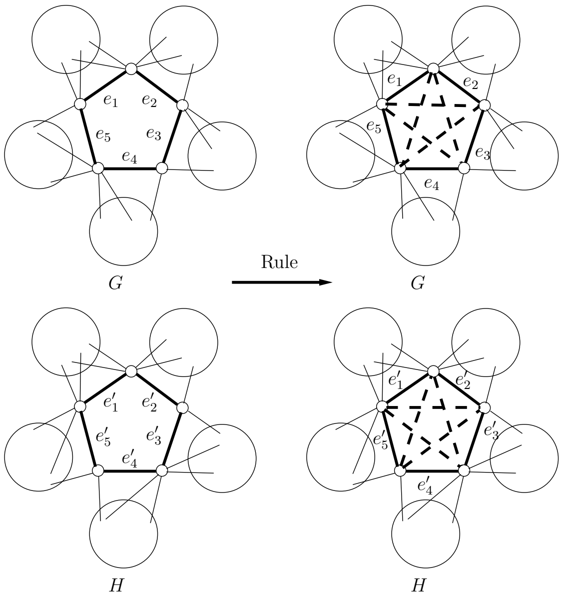

Figure 7 An example of an application of Reduction Rule 2; $\varphi(e_i) = e'_i$ for $i \in \{1, \ldots, 5\}$, the vertices of the φ-good bag of G and the corresponding bag of H are white, and the added edges are shown by dashed lines.

▶ **Reduction Rule 2.** *If for $t \in W^{(1)}_{\geq 3}$ such that $X^{(1)}_t$ is a φ-good, there are nonadjacent vertices in $X^{(1)}_t$, then compute the φ-isomorphism ψ of $G[X^{(1)}_t]$ to $H[X^{(2)}_{\alpha(t)}]$ and for every nonadjacent $u, v \in X^{(1)}_t$, do the following:*
- *add the edge uv to G and $\psi(u)\psi(v)$ to H,*
- *extend φ by setting $\varphi(uv) = \psi(u)\psi(v)$.*

The safeness of the rule follows from Lemma 12. We use that there is an optimal sequence of Whitney switches such that no Whitney separation splits φ-good bags. We apply Reduction Rule 2 for all bags of G that are not cliques. We use the same convention as for the first rule, and keep the old notation for the obtained graphs, their Tutte decompositions, and the obtained 2-isomorphism.

The next aim is to reduce the number of mutually φ-good bags by “gluing” them into cliques (see Fig. 8 for an example).

▶ **Reduction Rule 3.** *For distinct $t_1, t_2 \in W^{(1)}_{\geq 3}$ such that $X^{(1)}_{t_1}$ and $X^{(1)}_{t_2}$ are mutually φ-good,*
- *compute the φ-isomorphism ψ of $G[X^{(1)}_{t_1} \cup X^{(1)}_{t_2}]$ to $H[X^{(2)}_{\alpha(t_1)} \cup X^{(2)}_{\alpha(t_2)}]$,*
- *for every $u \in X^{(1)}_{t_1} \setminus X^{(1)}_{t_2}$ and every $v \in X^{(1)}_{t_2} \setminus X^{(1)}_{t_1}$, do the following:*
  - *add the edge uv to G and $\psi(u)\psi(v)$ to H,*
  - *extend φ by setting $\varphi(uv) = \psi(u)\psi(v)$,*
- *recompute the Tutte decompositions of the obtained graphs and the isomorphism α.*

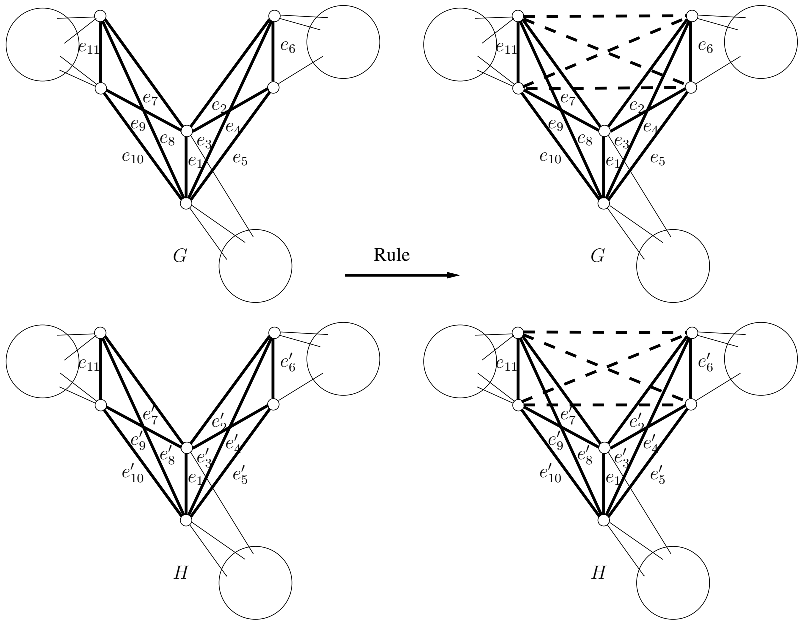

Figure 8 An example of an application of Reduction Rule 3; $\varphi(e_i) = e'_i$ for $i \in \{1, \ldots, 11\}$, the vertices of the mutually φ-good bags of G and the corresponding bags of H are white, and the added edges are shown by dashed lines.

To show the safeness, we use that, by Lemma 12, we can find an optimal sequence of Whitney switches such that the corresponding Whitney separations do not split mutually φ-good bags. Reduction Rule 3 is applied exhaustively whenever it is possible. As before, we do not change the notation for the obtained graphs, their Tutte decompositions, and the obtained 2-isomorphism.

Our next rule is used to perform the Whitney switches that are unavoidable (see Fig. 9 for an example).

▶ **Reduction Rule 4.** *If there is $t \in W_2^{(1)}$ such that $d_{T^{(1)}}(t) = 2$ and for the neighbors t_1 and t_2 of t, it holds that $X_{t_1}^{(1)}$ and $X_{t_2}^{(2)}$ are φ-good but not mutually φ-good, then do the following:*

- *find the connected components T_1 and T_2 of $T^{(1)} - t$, and construct $A = \bigcup_{t' \in V(T_1)} X_{t'}^{(1)}$ and $B = \bigcup_{t' \in V(T_2)} X_{t'}^{(1)}$,*
- *perform the Whitney switch with respect to the separation (A, B),*
- *set $k := k - 1$, and if $k < 0$, then return the trivial no-instance and stop.*

Reduction Rule 4 is applied exhaustively whenever it is possible. Note that after applying this rule, we are able to apply Reduction Rule 3 and we do it.

Suppose that the algorithm did not stop while executing Reduction Rule 4. In the same way as with previous rules, we maintain the initial notation for the obtained graphs, their Tutte decompositions, and the obtained 2-isomorphism.

Our final rule deletes simplicial vertices of degree at least 3.

▶ **Reduction Rule 5.** *If there is a simplicial vertex $v \in V(G)$ with $d_G(v) \geq 3$, then do the following:*

- *find the vertex $u \in V(H)$ such that $E_H(u) = \varphi(E_G(v))$,*
- *set $G := G - v$ and $H := H - u$,*
- *set $\varphi := \varphi|_{E(G) \setminus E_G(v)}$.*

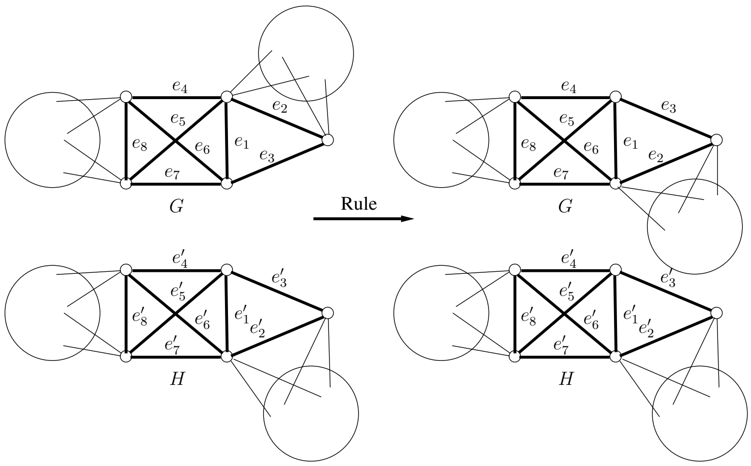

Figure 9 An example of an application of Reduction Rule 4; $\varphi(e_i) = e'_i$ for $i \in \{1, \dots, 8\}$, the vertices of the switched φ-good bags of G and the corresponding bags of H are white.

Reduction Rule 5 is applied exhaustively. Let G, H and φ be the resulting graphs. We also keep the same notation for the Tutte decompositions of G and H and the isomorphism α following the previous convention. This completes the description of our kernelization algorithm as the graphs G and H have bounded size. We show that $|V(G)| = |V(H)| \leq \max\{52 \cdot b(G) - 36, 3\}$.

It can be shown that Reduction Rules 1–5 do not increase the breakpoint number. Therefore, for the the obtained instance (G, H, φ, k) of Whitney Switches, $|V(G)| = |V(H)| \leq \min\{52 \cdot b - 36, 3\}$, where b is the breakpoint number of the initial input graph G. ◀

Theorem 14 together with Lemma 13 imply that Whitney Switches has a polynomial kernel when parameterized by k. Thus, we can show Theorem 2 that we restate.

▶ **Theorem 2.** *Whitney Switches admits a kernel with $\mathcal{O}(k)$ vertices and is solvable in $2^{\mathcal{O}(k \log k)} \cdot n^{\mathcal{O}(1)}$ time.*

In Corollary 4, we proved that Whitney Switches is NP-hard when the input graphs are constrained to be cycles. Theorem 14 indicates that it is the presence of bags in the Tutte decompositions that are cycles of length at least 4 that makes Whitney Switches difficult, because only such cycles may contain crucial breakpoint. In particular, we can derive the following straightforward corollary.

▶ **Corollary 15.** *Let (G, H, φ, k) be an instance of Whitney Switches such that $b(G) = 0$. Then Whitney Switches for this instance can be solved in polynomial time.*

For example, the condition that $b(G) = 0$ holds when G and H have no induced cycles of length at least 4, that is, when G and H are chordal graphs.

▶ **Corollary 16.** *Whitney Switches can be solved in polynomial time on chordal graphs.*

6 Conclusion

We proved that Whitney Switches admits a polynomial kernel when parameterized by the breakpoint number of the input graphs and this implies that the problem has a polynomial kernel when parameterized by k. More precisely, we obtain a kernel, where the graphs have $\mathcal{O}(k)$ vertices. Using this kernel, we can solve Whitney Switches in $2^{\mathcal{O}(k \log k)} \cdot n^{\mathcal{O}(1)}$ time. It is natural to ask whether the problem can be solved in a single-exponential in k time.

Another interesting direction of research is to investigate approximability for Whitney Switches. In [3], Berman and Karpinski proved that for every $\varepsilon > 0$, it is NP-hard to approximate the reversal distance $d(\pi)$ for a linear permutation π within factor $\frac{1237}{1236} - \varepsilon$. This result can be translated for circular permutations and this allows to obtain inapproximability lower bound for Whitney Switches on cycles similarly to Corollary 4. From the positive side, the currently best 1.375-approximation for $d(\pi)$ was given by Berman, Hannenhalli, and Karpinski [2]. Due to the close relations between Whitney Switches and the sorting by reversal problem, it is interesting to check whether the same approximation ratio can be achieved for Whitney Switches.

In Whitney Switches, we are given two graphs G and H together with a 2-isomorphism and the task is to decide whether we can apply at most k Whitney switches to obtain a graph G' from G such that G' is φ-isomorphic to H. We can relax the task and ask whether we can obtain G' that is isomorphic to H, that is, we do not require an isomorphism of G to H be a φ-isomorphism. Formally, we define the following problem.

Unlabeled Whitney Switches

Input:	2-Isomorphic graphs G and H, and a nonnegative integer k.
Task:	Decide whether it is possible to obtain a graph G' from G by at most k Whitney switches such that G' is isomorphic to H.

Note that if φ is a 2-isomorphism of G to H, then the minimum number of Whitney switches needed to obtain G' that is φ-isomorphic to H gives an upper bound for the number of Whitney switches required to obtain from G a graph that isomorphic to G. However, these values can be arbitrary far apart. Consider two cycles G and H with the same number of vertices. Clearly, G and H are isomorphic but for a given 2-isomorphism φ of G to H, we may need many Whitney switches to obtain G' that is φ-isomorphic to H and the number of switches is not bounded by any constant.

Using the result of Solomon, Sutcliffe, and Lister [27], we can show that Unlabeled Whitney Switches is NP-hard for very restricted instances.

▶ **Proposition 17.** *Unlabeled Whitney Switches is NP-complete when restricted to 2-connected series-parallel graphs even if the input graphs are given together with their 2-isomorphism.*

Proposition 17 lead to the question about the parameterized complexity of Unlabeled Whitney Switches. In particular, does the problem admit a polynomial kernel when parameterized by k?

Notice that to deal with Unlabeled Whitney Switches, we should be able to check whether the input graphs G and H are isomorphic. If we are given a 2-isomorphism φ of G to H, then checking whether G and H are φ-isomorphic can be done in polynomial time by Lemma 6. However, checking whether G and H are isomorphic, even if a 2-isomorphism φ is given, is a complicated task. For example, it can be observed that this is at least as difficult as solving Graph Isomorphism on tournaments (recall that a *tournament* is a directed graph such that for every two distinct vertices u and v, either uv or vu is an arc). While Graph Isomorphism on tournaments may be easier than the general problem (we refer to [26, 33] for the details), still it is unknown whether this special case can be solved in polynomial time and the best known algorithm is the quasi-polynomial algorithm of Babai [1]. Given this observation, it is natural to consider Unlabeled Whitney Switches on graph classes for which Graph Isomorphism is polynomially solvable. For example, what can be said about Unlabeled Whitney Switches on planar graphs?

The relation between Whitney switches and sorting by reversals together with the reduction in the proof of Proposition 17 indicates that as the first step, it could be reasonable to investigate the following problem for sequences that generalizes Sorting by Reversals for permutations. Let $\pi = (\pi_1, \ldots, \pi_n)$ be a sequence of positive integers; note that now some elements of π may be the same. For $1 \leq i < j \leq n$, we define the *reversal* $\rho(i, j)$ in exactly the same way as for permutations. Then we can define the *reversal distance* between two n-element sequences such that the multisets of their elements are the same; we assume that the distance is $+\infty$ if the multisets of elements are distinct.

Sequence Reversal Distance

Input:	Two n-element sequences π and σ of positive integers and a nonnegative integer k.
Task:	Decide whether the reversal distance between π and σ is at most k.

By the result of Caprara in [7], this problem is NP-complete even if the input sequences are permutations. It is also known that the problem is NP–complete if the input sequences contain only two distinct elements [9]. The question, whether Sequence Reversal Distance is FPT when parameterized by k, was explicitly stated in the survey of Bulteau et. al [6] (in terms of strings) and is open and only some partial results are known [5]. We also can define the version of Sequence Reversal Distance for circular sequences and ask the same question about parameterized complexity. Using the idea behind the reduction in the proof of Proposition 17, it is easy to observe that Unlabeled Whitney Switches on 2-connected series-parallel graphs is at least as hard as the circular variant of Sequence Reversal Distance.

References

1 László Babai. Graph isomorphism in quasipolynomial time [extended abstract]. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 684–697. ACM, 2016. doi:10.1145/2897518.2897542.

2 Piotr Berman, Sridhar Hannenhalli, and Marek Karpinski. 1.375-approximation algorithm for sorting by reversals. In *Algorithms - ESA 2002, 10th Annual European Symposium, Rome, Italy, September 17-21, 2002, Proceedings*, volume 2461 of *Lecture Notes in Computer Science*, pages 200–210. Springer, 2002. doi:10.1007/3-540-45749-6_21.

3 Piotr Berman and Marek Karpinski. On some tighter inapproximability results (extended abstract). In *Automata, Languages and Programming, 26th International Colloquium, ICALP'99, Prague, Czech Republic, July 11-15, 1999, Proceedings*, volume 1644 of *Lecture Notes in Computer Science*, pages 200–209. Springer, 1999. doi:10.1007/3-540-48523-6_17.

4 Prosenjit Bose and Ferran Hurtado. Flips in planar graphs. *Comput. Geom.*, 42(1):60–80, 2009. doi:10.1016/j.comgeo.2008.04.001.

5 Laurent Bulteau, Guillaume Fertin, and Christian Komusiewicz. (Prefix) reversal distance for (signed) strings with few blocks or small alphabets. *J. Discrete Algorithms*, 37:44–55, 2016. doi:10.1016/j.jda.2016.05.002.

6 Laurent Bulteau, Falk Hüffner, Christian Komusiewicz, and Rolf Niedermeier. Multivariate algorithmics for NP-hard string problems. *Bulletin of the EATCS*, 114, 2014. URL: http://eatcs.org/beatcs/index.php/beatcs/article/view/310.

7 Alberto Caprara. Sorting by reversals is difficult. In *Proceedings of the First Annual International Conference on Research in Computational Molecular Biology, RECOMB 1997, Santa Fe, NM, USA, January 20-23, 1997*, pages 75–83. ACM, 1997. doi:10.1145/267521.267531.

8 Markus Chimani, Petr Hlinený, and Petra Mutzel. Vertex insertion approximates the crossing number of apex graphs. *Eur. J. Comb.*, 33(3):326–335, 2012. doi:10.1016/j.ejc.2011.09.009.

9 David A. Christie and Robert W. Irving. Sorting strings by reversals and by transpositions. *SIAM J. Discrete Math.*, 14(2):193–206, 2001. doi:10.1137/S0895480197331995.

10 Sean Cleary and Katherine St. John. Rotation distance is fixed-parameter tractable. *Inform. Process. Lett.*, 109(16):918–922, 2009. doi:10.1016/j.ipl.2009.04.023.

11 Bruno Courcelle. The monadic second-order logic of graphs XI: hierarchical decompositions of connected graphs. *Theor. Comput. Sci.*, 224(1-2):35–58, 1999. doi:10.1016/S0304-3975(98)00306-5.

12 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. doi:10.1007/978-3-319-21275-3.

13 Reinhard Diestel. *Graph Theory, 4th Edition*, volume 173 of *Graduate texts in mathematics*. Springer, 2012.

14 Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013. doi:10.1007/978-1-4471-5559-1.

15 Fedor V. Fomin and Petr A. Golovach. Kernelization of whitney switches. *CoRR*, abs/2006.13684, 2020. arXiv:2006.13684.

16 Fedor V. Fomin, Daniel Lokshtanov, Saket Saurabh, and Meirav Zehavi. *Kernelization*. Cambridge University Press, Cambridge, 2019. Theory of parameterized preprocessing.

17 Sridhar Hannenhalli and Pavel A. Pevzner. To cut... or not to cut (applications of comparative physical maps in molecular evolution). In *Proceedings of the Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, 28-30 January 1996, Atlanta, Georgia, USA*, pages 304–313. ACM/SIAM, 1996. URL: http://dl.acm.org/citation.cfm?id=313852.314077.

18 John E. Hopcroft and Robert Endre Tarjan. Dividing a graph into triconnected components. *SIAM J. Comput.*, 2(3):135–158, 1973. doi:10.1137/0202012.

19 Iyad A. Kanj, Eric Sedgwick, and Ge Xia. Computing the flip distance between triangulations. *Discrete & Computational Geometry*, 58(2):313–344, 2017. doi:10.1007/s00454-017-9867-x.

20 John D. Kececioglu and David Sankoff. Exact and approximation algorithms for sorting by reversals, with application to genome rearrangement. *Algorithmica*, 13(1/2):180–210, 1995. doi:10.1007/BF01188586.

21 Daniel Lokshtanov, Amer E Mouawad, Fahad Panolan, MS Ramanujan, and Saket Saurabh. Reconfiguration on sparse graphs. *Journal of Computer and System Sciences*, 95:122–131, 2018.

22 Anna Lubiw and Vinayak Pathak. Flip distance between two triangulations of a point set is NP-complete. *Comput. Geom.*, 49:17–23, 2015. doi:10.1016/j.comgeo.2014.11.001.

23 Joan M. Lucas. An improved kernel size for rotation distance in binary trees. *Inform. Process. Lett.*, 110(12-13):481–484, 2010. doi:10.1016/j.ipl.2010.04.022.

24 Amer E. Mouawad, Naomi Nishimura, Venkatesh Raman, Narges Simjour, and Akira Suzuki. On the parameterized complexity of reconfiguration problems. *Algorithmica*, 78(1):274–297, 2017. doi:10.1007/s00453-016-0159-2.

25 Pavel A. Pevzner. *Computational molecular biology - an algorithmic approach*. MIT Press, 2000.

26 Pascal Schweitzer. A polynomial-time randomized reduction from tournament isomorphism to tournament asymmetry. In *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10-14, 2017, Warsaw, Poland*, volume 80 of *LIPIcs*, pages 66:1–66:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017. doi:10.4230/LIPIcs.ICALP.2017.66.

27 Andrew Solomon, Paul J. Sutcliffe, and Raymond Lister. Sorting circular permutations by reversal. In *Algorithms and Data Structures, 8th International Workshop, WADS 2003, Ottawa, Ontario, Canada, July 30 - August 1, 2003, Proceedings*, volume 2748 of *Lecture Notes in Computer Science*, pages 319–328. Springer, 2003. doi:10.1007/978-3-540-45078-8_28.

28 Carsten Thomassen. Embeddings and minors. In *Handbook of combinatorics, Vol. 1, 2*, pages 301–349. Elsevier Sci. B. V., Amsterdam, 1995.

29 Klaus Truemper. On whitney's 2-isomorphism theorem for graphs. *Journal of Graph Theory*, 4(1):43–49, 1980. `doi:10.1002/jgt.3190040106`.

30 W. T. Tutte. *Connectivity in graphs.* Mathematical Expositions, No. 15. University of Toronto Press, Toronto, Ont.; Oxford University Press, London, 1966.

31 W. T. Tutte. *Graph theory*, volume 21 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 2001. With a foreword by Crispin St. J. A. Nash-Williams, Reprint of the 1984 original.

32 Dirk L. Vertigan and Geoffrey P. Whittle. A 2-isomorphism theorem for hypergraphs. *J. Comb. Theory, Ser. B*, 71(2):215–230, 1997. `doi:10.1006/jctb.1997.1789`.

33 Fabian Wagner. Hardness results for tournament isomorphism and automorphism. In *Mathematical Foundations of Computer Science 2007, 32nd International Symposium, MFCS 2007, Ceský Krumlov, Czech Republic, August 26-31, 2007, Proceedings*, volume 4708 of *Lecture Notes in Computer Science*, pages 572–583. Springer, 2007. `doi:10.1007/978-3-540-74456-6_51`.

34 Hassler Whitney. Non-separable and planar graphs. *Trans. Amer. Math. Soc.*, 34(2):339–362, 1932. `doi:10.2307/1989545`.

35 Hassler Whitney. 2-Isomorphic Graphs. *Amer. J. Math.*, 55(1-4):245–254, 1933. `doi:10.2307/2371127`.