Reconfiguration of Digraph Homomorphisms

Benjamin Lévêque
Laboratoire G-SCOP, Grenoble INP, Université Grenoble-Alpes, France

Moritz Mühlenthaler
Laboratoire G-SCOP, Grenoble INP, Université Grenoble-Alpes, France

Thomas Suzan
Laboratoire G-SCOP, Grenoble INP, Université Grenoble-Alpes, France

Abstract

For a fixed graph H, the H-Recoloring problem asks whether, given two homomorphisms from a graph G to H, one homomorphism can be transformed into the other by changing the image of a single vertex in each step and maintaining a homomorphism to H throughout. The most general algorithmic result for H-Recoloring so far has been proposed by Wrochna in 2014, who introduced a topological approach to obtain a polynomial-time algorithm for any undirected loopless square-free graph H. We show that the topological approach can be used to recover essentially all previous algorithmic results for H-Recoloring and that it is applicable also in the more general setting of digraph homomorphisms. In particular, we show that H-Recoloring admits a polynomial-time algorithm i) if H is a loopless digraph that does not contain a 4-cycle of algebraic girth 0 and ii) if H is a reflexive digraph that contains no triangle of algebraic girth 1 and no 4-cycle of algebraic girth 0.

2012 ACM Subject Classification Mathematics of computing → Discrete mathematics

Keywords and phrases Digraph Homomorphisms, Combinatorial Reconfiguration

Digital Object Identifier 10.4230/LIPIcs.STACS.2023.43

Related Version *Full Version*: `https://arxiv.org/abs/2205.09210` [14]

1 Introduction

Reconfiguration problems have been introduced formally by Ito et al. in [11] and their complexity has been studied systematically since. Applications can be found in statistical physics, combinatorial games, and uniform sampling of objects such as colorings and matchings. The general setting is the following: Given two feasible solutions of an instance of a combinatorial problem, the goal is to decide whether one can be transformed into the other in a step-by-step manner, visiting only feasible configurations during the transformation. Related questions of interest are whether any two feasible solutions admit a transformation, and if there is a transformation of at most a certain length between two given solutions. We refer the reader to the surveys of Nishimura [15] and van den Heuvel [16] for a discussion of results and applications in this area.

A digraph homomorphism maps the vertex set of a digraph G to the vertex set of a digraph H such that each arc of G is mapped to an arc of H. The classical digraph homomorphism problem CSP(H) asks whether there is a digraph homomorphism from a given digraph G to a fixed "template" digraph H. Besides the fact that CSP(H) generalizes graph coloring, one of the motivations for studying its complexity is that it is polynomially equivalent to the seemingly richer constraint satisfaction problem CSP($\mathcal{H}$), where the template $\mathcal{H}$ can be any fixed finite relational structure [8]. The complexity of CSP(H) (and hence CSP($\mathcal{H}$)) is well understood in the sense that for any digraph H, the problem CSP(H) is known to be either polynomial-time solvable or NP-complete, a result that has been proved recently by Bulatov [5] and by Zhuk [18], settling in the affirmative a long-standing conjecture by

40th International Symposium on Theoretical Aspects of Computer Science (STACS 2023).
Editors: Petra Berenbrink, Patricia Bouyer, Anuj Dawar, and Mamadou Moustapha Kanté;
Article No. 43; pp. 43:1–43:21
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Feder and Vardi [8]. Motivated by these recent developments we study the complexity of the natural reconfiguration variant H-Recoloring associated with CSP(H), which is the following question: Given two digraph homomorphisms α and β from a G to H, is there a step-by-step transformation between α and β that changes the image of one vertex of G at a time and maintains a digraph homomorphism from G to H throughout?

The complexity of H-Recoloring is known for several special cases, most notably cases when H is from some class of undirected loopless graphs. Despite several positive [4, 6, 13, 17] and negative [1, 4, 12] results, a complete classification of the complexity of H-Recoloring even for undirected graphs H is not known. By replacing each edge of an undirected graph by two directed edges of opposite orientation, we see that digraphs homomorphisms are strictly more general than homomorphisms of undirected graphs in this context. For digraphs H, we are aware of only two results for H-Recoloring, which consider the case where H is a transitive tournament [7] and where H is some orientation of a reflexive digraph cycle [2] (a graph is *reflexive* if it has a loop on each vertex).

The *algebraic girth* of the orientation of a cycle is the absolute value of the number of forward arcs minus the number of backward arcs. In particular, a *4-cycle of algebraic girth 0* is either one of the two graphs shown in Figure 1a. We extend the topological approach developed by Wrochna [17] in the context of undirected graphs to digraphs and obtain a polynomial-time algorithm for CSP(H) for any graph H that contains no 4-cycle of algebraic girth 0 as a subgraph.

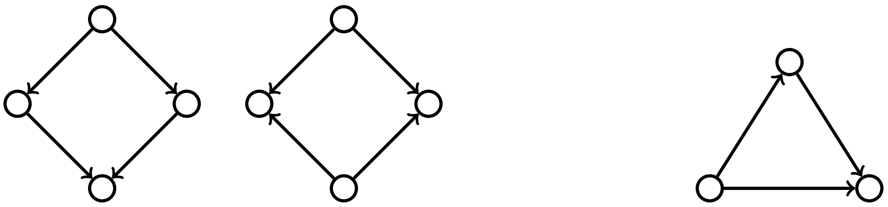
▶ **Theorem 1.** *Let H be a loopless digraph that contains no 4-cycle of algebraic girth 0 as a subgraph. Then H-Recoloring admits a polynomial-time algorithm.*

Theorem 1 generalizes the polynomial-time algorithm for CSP(H) given in [17] for undirected graphs H without a cycle on four vertices as subgraph, which in turn is a generalization of [6], where H is a complete graph on three vertices. The triangle of algebraic girth 1 is shown in Figure 1b. For reflexive digraphs we obtain the following result.

▶ **Theorem 2.** *Let H be a reflexive digraph that contains neither a triangle of algebraic girth 1 nor a 4-cycle of algebraic girth 0 as a subgraph. Then H-Recoloring admits a polynomial-time algorithm.*

Theorem 2 generalizes the algorithmic results from [2], where H is assumed to be a reflexive digraph cycle and from [13], where H is a triangle-free reflexive undirected graph. We remark that the algorithms of theorems 1 and 2 produce certificates for both Yes and No instances and that their running time is polynomial in the size of G and the size of H. The remaining known algorithmic results in [4, 7], while not implied directly by theorems 1 and 2, can be obtained in a straight-forward manner using the topological approach, see sections B.2 and B.3. In the light of our results, to our knowledge, all previous algorithmic results for H-Recoloring can be obtained by the topological approach. Therefore it seems natural to ask whether there are digraphs H such that the topological approach for H-Recoloring does not work, but there is nevertheless a polynomial-time algorithm. This question seems to be a stepping stone to a complete classification of the complexity of H-Recoloring.

One intriguing property of reconfiguration problems is that they can be easy even if the underlying decision problem is hard and vice versa. To illustrate this, consider the classical 3-Coloring problem, which can be formulated as follows: Given an undirected graph G, decide if there is a homomorphism from G to the complete graph on three vertices. While 3-Coloring is NP-complete, Cereceda et al. showed [6] that, given two such homomorphisms ("3-colorings"), there is a polynomial-time algorithm that decides whether there is a step-by-step transformation between them. Theorem 1 generalizes the result of Cereceda et al.



(a) The two non-isomorphic orientations of a 4-cycle of algebraic girth zero.

(b) The orientation of a 3-cycle of algebraic girth 1.

Figure 1 Orientations of a 4-cycle and a 3-cycle that appear in theorems 1 and 2.

and provides new examples with similar behavior. For instance, it is known that there are orientations H of a tree such that CSP(H) is NP-complete, but Theorem 1 implies that for any orientation H of any tree, the problem H-Recoloring admits a polynomial-time algorithm. The situation is different for reflexive graphs: Here, deciding if a given graph G admits a homomorphism to a fixed reflexive graph H is trivial, since a homomorphism may send all vertices of G to the same looped vertex of H. However, deciding if two given homomorphisms to a reflexive graph H admit a step-by-step transformation turns out to be non-trivial even for restricted graphs H (see [3, 13] and the proof of Theorem 2) and is PSPACE-complete in general [17].

1.1 Our results and their relation to Wrochna's algorithm

We show that the topological approach introduced by Wrochna [17] for reconfiguring homomorphisms of undirected graphs can be extended to the digraph homomorphisms. An undirected graph is *square-free* if it does not contain a cycle on four vertices as a subgraph. For a homomorphism $G \to H$ of directed or undirected graphs, we refer to the image of a vertex of G as its *color*. Two graph homomorphisms $\alpha, \beta : G \to H$ admit a step-by-step transformation if there is a sequence $\sigma_1, \sigma_2, \ldots, \sigma_\ell$ of homomorphisms $G \to H$, such that $\alpha = \sigma_1$, $\beta = \sigma_\ell$, and any two consecutive homomorphisms σ_i, σ_{i+1} differ with respect to the color of exactly one vertex. Such a sequence is called *H-recoloring sequence* (from α to β). One key observation of Wrochna [17] is that if H is an undirected square-free graph then, whenever the color of a vertex changes during a step-by-step transformation then all of its neighbors must have the same color[1]. This so-called *monochromatic neighborhood property* sets the stage for a topological approach to the H-Recoloring problem. Here, the graphs are considered to be continuous objects, obtained by gluing edges represented by unit intervals to their respective end-points, and graph homomorphism are continuous maps between the corresponding topological spaces. The monochromatic neighborhood property implies that if there is an H-recoloring sequence between two homomorphisms then they are homotopy-equivalent. This permits to represent H-recoloring sequences satisfying the monochromatic neighborhood property (up to re-ordering) by walks in H that correspond to the color changes of a single vertex of G. Wrochna [17] provides a characterization of all such walks for square-free graphs H, which leads to a polynomial-time algorithm for the corresponding H-Recoloring problem.

[1] It is easy to verify that if a vertex v of an undirected graph G has two neighbors with distinct colors then a color change of v implies that H contains a square.

At first sight, the crucial premise of Wrochna's algorithm seems to be the square-freeness of H. But in fact, the algorithm finds for any undirected graph H, square-free or not, walks that represent all H-recoloring sequences satisfying the monochromatic neighborhood property (possibly there is no such walk). In [17], Wrochna remarks the following.

▶ Remark 3 ([17]). We note that none of the proofs in this paper used any structural properties of H. If we consider H-Recoloring for any graph H, but only allow recoloring a vertex if all of its neighbors have one common color (in other words, a reconfiguration step is allowed only when the homotopy class of the mapping does not change), the same results will follow.

This remark implies immediately that his algorithm also works for undirected graphs H with loops allowed, whenever H does not contain C_4, K_3 with one loop added and K_2 with both loops added.

For a loopless digraph H, a structural property that enforces the monochromatic neighborhood property of any H-recoloring sequence is that H does not contain a 4-cycle of algebraic girth 0 (see Figure 1a). Following the discussion of Wrochna's algorithm above, we may apply it to the corresponding undirected graph $\bar{H}$ and it will return a description of all walks that represent $\bar{H}$-recoloring sequences satisfying the monochromatic neighborhood property. To obtain Theorem 1, it remains to determine whether or not one of the walks corresponds to an H-recoloring sequence that is compatible with the orientation of the arcs of H. For this purpose we introduce the so-called zigzag condition and show that the walks in H that represent H-recoloring sequences satisfying the monochromatic neighborhood property and the zigzag condition are precisely those that are compatible with the orientation of H (Theorem 16). Finally, to prove Theorem 1, we show that it can be checked in polynomial time whether any of the walks found by Wrochna's algorithm satisfies the zigzag condition. Using ideas from [17], a polynomial-time algorithm for finding *shortest* H-recoloring sequences can be obtained (see Section B.1).

In order to prove Theorem 2, the topological approach needs to be adapted to the setting of reflexive graphs. For this purpose, we introduce the *push-or-pull* property, which is similar to the monochromatic neighborhood property in a topological sense. We say that an H-recoloring sequence satisfies the *push-or-pull property* if, whenever a vertex of a graph G changes its color, say from a to b, then all its neighbors in G have either color a or b. This property ensures that if we consider the graphs G and H as topological spaces and homomorphisms $G \to H$ as continuous mappings between them (as described above), then all H-colorings of a given H-recoloring sequence satisfying the push-or-pull property are homotopy-equivalent. See Figure 2 for an example of two homomorphisms to a reflexive triangle that are not homotopy-equivalent: the first wraps around the triangle while the second does not.

It is not hard to see that for any triangle-free reflexive undirected graph H, any H-recoloring sequence satisfies the push-or-pull property. Similar to the loopless case we characterize those walks in H that correspond to H-recoloring sequences satisfying the push-or-pull property (see Theorem 20). From this characterization we obtain a polynomial-time algorithm for H-Recoloring for any undirected reflexive graph H of girth at least 5 (Corollary 24 in [14]). Recently, Lee et al. [13] have obtained the same result using other methods.

An advantage of the topological approach is that it allows for a generalization to digraphs H: we use the characterization of walks in Theorem 20 for the undirected graph $\bar{H}$ (almost) as a black box and then check whether any of the corresponding $\bar{H}$-recoloring sequences is compatible with the orientation of the arcs of H. Depending on which case of

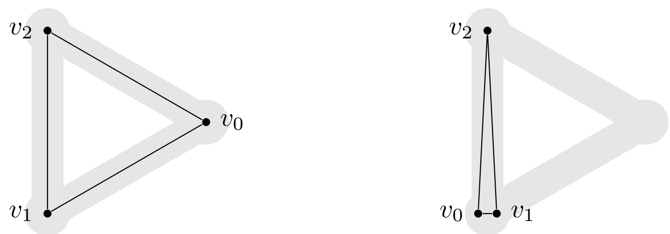


Figure 2 Two homomorphisms $G \to H$ that differ in the color of one vertex but which are not homotopy-equivalent. The graph G (in black) is a triangle and the graph H (in gray) is a triangle with a loop on each vertex.

Theorem 20 applies, different levels of sophistication are required, but in any case there is a polynomial-time algorithm. Notice that the push-or-pull property implies that when a vertex changes its color, say from a to b, then a and b are adjacent in H. Using ideas from [13], we show that we can get rid of this restriction in the case that H contains no 4-cycle of algebraic girth 0 which is the final ingredient of Theorem 2. The last step illustrates the connection between H-Recoloring and the so-called Hom-graph, which is discussed in Section 3 in [14].

1.2 Related work

The complexity of H-Recoloring for undirected graphs H has been studied systematically, in particular since the work of Cereceda et al. [6], who showed that if $H = K_3$, a complete graph on three vertices, then H-Recoloring admits a polynomial-time algorithm, despite $\mathrm{CSP}(K_3)$ ("3-Coloring") being NP-complete. Wrochna [17] generalized this result, showing that H-Recoloring admits a polynomial-time algorithm if H is loopless and square-free. Brewster et al. [4] gave a complexity classification of H-Recoloring for circular cliques $C_{p,q}$. Note that their polynomial-time algorithm for $2 \leq p/q < 4$ includes graphs H that are not square-free. Recently, Lee et al. [13] adapted Wrochna's algorithm to the case that H is reflexive and has girth at least 5. On the negative side, it is known that H-Recoloring is PSPACE-complete if H is a clique on at least four vertices [1], a circular clique $C_{p,q}$ where $p/q \geq 4$ [4], a wheel on a k-cycle, where $k \geq 3$ and $k \neq 4$ [12], or a quadrangulation of the 2-sphere with certain properties [12].

We are aware of two results for H-Recoloring for digraphs H. The first one is by Brewster et al. [2], who showed that H-Recoloring admits a polynomial-time algorithm if H is a reflexive digraph cycle that does not contain a 4-cycle of algebraic girth 0. In spirit this algorithm uses the topological approach similar to that of Wrochna (but more involved) that reduces the task of finding H-recoloring sequences to finding vertex walks in H. Secondly, Dochterman and Singh [7] study the Hom-complex for digraphs G and H and show that it is connected (in the topological sense) if H is the transitive tournament T_n on n vertices. From this they conclude that any instance of T_n-Recoloring is a Yes-instance and give a polynomial-time algorithm that finds a T_n-recoloring sequence. The algorithm is simple and does not need any topological tools. It boils down to the fact that a homomorphism of an acyclic digraph into a tournament corresponds to a linear extension of a partial order. To reconfigure one linear extension into another, we may greedily take the last element where the two linear extensions disagree and assign to the vertex with the smaller image the larger image (w.r.t. the total order). The same result can be obtained using the topological approach (see Section B.3).

Further results are known for $\mathcal{H}$-Recoloring for a relational structure $\mathcal{H}$ on a Boolean domain. This problem corresponds to the reconfiguration of satisfying assignments of Boolean formulas. In [9], Gopalan et al. provide a complexity dichotomy for this problem by characterizing the relations for which Boolean satisfiability reconfiguration admits a polynomial-time algorithm and showing that the problem is PSPACE-complete otherwise. Another popular theme with a different flavor is the reconfiguration of subgraphs, which may be considered to be homomorphisms (injective or not) from a fixed graph H to a given graph G. In this context, Ito et al. showed that reconfiguring directed paths is PSPACE-complete. Often however the graph H is not fixed, but any subgraph of a certain “shape”, e.g., any tree, would be acceptable. We refer the readers to the survey of Nishimura [15] for an overview of known results in this direction, in particular those on the reconfiguration of independent sets, as well as [10] for results on spanning and induced subgraphs. A general introduction to reconfiguration problems that also discusses their relation to combinatorial games and puzzles can be found in [16].

1.3 Organization

Section 2 contains notation and basic definitions that are needed to prove Theorems 1 and 2. Section 3 gives an overview of the proof of Theorem 1, which implies a polynomial-time algorithm for H-Recoloring if H is a loopless digraph that contains no 4-cycle of algebraic girth 0. This section also serves as a blueprint for the more involved proof of Theorem 2, which is sketched in Section 4. Due to space limitations the full proofs are deferred to the appendix.

2 Preliminaries

A directed graph (digraph) is a pair $(V(G), A(G))$ where $V(G)$ is a finite set of vertices and $A(G) \subseteq V(G) \times V(G)$ are *arcs*. We write $u \to v$ when $uv \in A(G)$. We say that a digraph G is *symmetric* if $vu \in A(G)$ whenever $uv \in A(G)$. A digraph G is *reflexive* if $uu \in A(G)$ for each vertex $u \in V(G)$. We interpret a symmetric digraph as *undirected graph* and think of two edges $\{uv, vu\}$ as undirected edge, which we also write as uv since it should be clear from the context whether we refer to a directed or undirected edge. We write $E(G)$ for the set of undirected edges of a symmetric graph G. For any digraph G, we canonically associate to G an undirected graph $\bar{G}$ where $V(\bar{G}) = V(G)$ and $uv \in E(\bar{G})$ if $u \to v$ or $v \to u$. Let G be a digraph. The *in-neighborhood* (resp., *out-neighborhood*) of a vertex $v \in V(G)$ is given by $N^-_G(v) := \{w \in V(G) \mid w \to v\}$ (resp., $N^+_G(v) := \{w \in V(G) \mid v \to w\}$). If G is symmetric (undirected), the neighborhood $N_G(v)$ of a vertex $v \in V(G)$ is the set of vertices adjacent to v in G, that is, $N_G(v) := \{w \in V(G) \mid vw \in E(G)\}$. Let G and H be digraphs. A homomorphism $\phi : G \to H$ or (H-coloring of G) is a map $V(G) \to V(H)$ that preserves arcs, that is, for each $u \to v$, we have $\phi(u) \to \phi(v)$. Similarly, for undirected graphs G and H, a homomorphism $\phi\colon G \to H$ is a map $V(G) \to V(H)$ that preserves edges (but not necessarily non-edges). A homomorphism $\alpha\colon G \to H$ also induces a homomorphism $\bar{\alpha}\colon \bar{G} \to \bar{H}$.

A sequence $W = (v_1v_2)(v_2v_3)\dots(v_{n-1}v_n)$ of consecutive edges of an undirected graph G is a *walk*. The reverse walk W^{-1} of a walk $W = (v_1v_2)(v_2v_3)\dots(v_{n-1}v_n)$ is the walk $W^{-1} = (v_nv_{n-1})\dots(v_2\,v_1)$. The length $|W|$ of W the number of edges of W. A *cycle* C is a closed walk, i.e., a walk such that $v_1 = v_n$ on $n-1$ distinct vertices. A walk in digraph G is a walk in $\bar{G}$. The *algebraic girth* of a cycle C in a digraph G is the absolute value of the number of forward arcs minus the number of backward arcs. We say that a graph or a digraph G is *connected* if for any two vertices $u, v \in V(G)$ there is a walk from u to v in G. A walk $W = (v_1v_2)\dots(v_{n-1}v_n)$ in a digraph is *directed* if $v_i \to v_{i+1}$ for all $1 \le i \le n-1$. The walk W is *symmetric* if both W and W^{-1} are directed. We denote the empty walk by ε.

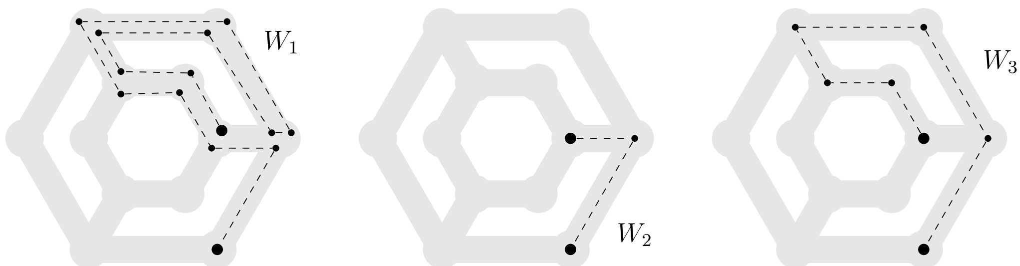


Figure 3 Consider the gray graph H and the three walks W_1, W_2 and W_3 in H (their successive vertices are drawn in black). The walk W_1 reduces to W_2. The walk W_3 is already reduced.

Fundamental groupoid

Let H be an undirected or directed graph. Given a walk $W = (v_1\, v_2)(v_2\, v_3)\dots(v_{n-1}\, v_n)$ in H, we call *reduction* the two following operations (see Figure 3):

- The operation of deleting $(v_i\, v_{i+1})(v_{i+1}\, v_{i+2})$ from W if $v_i = v_{i+2}$ and $1 \le i \le n-2$
- The operation of deleting $(v_i\, v_{i+1})$ from W if $v_i = v_{i+1}$ and $1 \le i \le n-1$. Note that this operation requires a loop on v_i, so it applies in particular if H is reflexive.

We say that W is *reduced* if none of the two operations above is applicable. That is, for $1 \le i \le n-2$, we have $v_{i+2} \ne v_i$ and for $1 \le i \le n-1$, we have $v_{i+1} \ne v_i$. We can *reduce* a walk W by iteratively applying reductions on it; we can easily see that by doing so we obtain a unique reduced walk. By considering two walks to be equivalent if they reduce to the same walk, we obtain an equivalence relation $\sim$ on the walks in H. The fundamental groupoid $\pi(H)$ is the set of all equivalence classes of walks in H under $\sim$. Its groupoid operation is the concatenation $\cdot$ of walks and its neutral element is the empty walk ε. For any walk $W = (v_1\, v_2)\dots(v_{n-1}\, v_n)$, the inverse of (the class of) W in $\pi(H)$ is (the class of) the reversed walk $W^{-1} = (v_n\, v_{n-1})\dots(v_2\, v_1)$ since both WW^{-1} and $W^{-1}W$ reduce to ϵ. In the next sections of this paper, we will write $W_1 = W_2$ *in* $\pi(H)$ if $W_1 \sim W_2$, that is, W_1 and W_1 reduce to the same walk.

Cyclic reduction

We say that closed walk $C = (v_1 v_2)\dots(v_{n-1} v_1)$ is *cyclically reduced* if it is reduced and additionally $v_2 \ne v_{n-1}$. We can *cyclically reduce* any reduced closed walk C by iteratively deleting from it both its first and last edges while one is the inverse of the other. This operation leads to a unique decomposition $C = A^{-1} C_0 A$ where A is the sequence of deleted edges of C and C_0 is cyclically reduced.

H-recoloring

Let G and H be digraphs. Recall that two digraph homomorphisms $\alpha, \beta : G \to H$ admit an *H-recoloring sequence* if for some ℓ there are digraph homomorphisms $\sigma_1, \sigma_2, \dots, \sigma_\ell : G \to H$, such that $\alpha = \sigma_1$, $\beta = \sigma_\ell$ and for $1 \le i < \ell$, the two homomorphisms σ_i and σ_{i+1} differ with respect to the color of a single vertex of G. The problem H-Recoloring asks whether, given a graph G and two digraph homomorphisms $\alpha, \beta : G \to H$, the homomorphisms α and β admit an H-recoloring sequence. An H-recoloring sequence satisfies the *monochromatic neighborhood property* if, whenever a vertex v of G changes its color then all neighbors of v in G have the same color.

In this entire paper we assume that G and H are directed or undirected, (weakly) connected graphs, with at least two vertices

One can see that assuming the connectivity of G and H imposes no restrictions, since if G is not connected, we may consider the recoloring of each connected component of G separately. If H is not connected, observe that any connected component of G maps to a connected component of H.

3 Loopless digraphs

In this section we prove Theorem 1. To do so, we extend the polynomial-time algorithm from [17] for H-Recoloring for the case that H is symmetric and square-free to the case where H is any loopless digraph that contains no 4-cycle of algebraic girth 0. We assume in the following that G and H are simple weakly connected digraphs with at least two vertices. Observe that since H is a subgraph of the symmetric graph $\bar{H}$, any H-recoloring sequence is also a $\bar{H}$-recoloring sequence. Furthermore, if H contains no 4-cycle of algebraic girth 0 then any H-recoloring sequence satisfies the monochromatic neighborhood property. To see this, consider a step of any H-recoloring sequence where the color of a vertex $u \in V(G)$ changes from a to b ($a, b \in V(H)$). Let v be any neighbor of u and let h be the current color of v. If a color different from h appears in the neighborhood of u then H contains a cycle of algebraic girth 0, that is, one of the two orientations of the 4-cycle shown in Figure 1a. To prove Theorem 1, the main idea is to run Wrochna's algorithm on the symmetric graphs $\bar{G}$ and $\bar{H}$ to obtain a description of those $\bar{H}$-recoloring sequences that satisfy the monochromatic neighborhood property. We then check whether one of these sequences is compatible with the orientation of the edges of H.

3.1 Realizable walks in $\bar{H}$

We recall several results and definitions from [17] that will be useful later on, stating them in a slightly different manner for the sake of better integration in the context of digraph homomorphisms. Largely the same proofs apply however, as pointed out in Remark 3.

Let $S = \sigma_0, \dots, \sigma_\ell$ be a $\bar{H}$-recoloring sequence satisfying the monochromatic neighborhood property and let v be a vertex of G. For each $0 \le i < \ell$, let $S_i(v)$ be given by

$$S_i(v) = \begin{cases} \epsilon & \text{if } \sigma_i(v) = \sigma_{i+1}(v), \text{ and} \\ (\sigma_i(v)\, h)(h\, \sigma_{i+1}(v)) & \text{otherwise,} \end{cases}$$

where h is the unique color of the neighbors of v with respect to $\sigma_i(v)$ and $\sigma_{i+1}(v)$. We associate with S and v the walk $S(v) := S_0(v)S_1(v)\cdots S_\ell(v)$ in H. Suppose that $S(v) = (a_1\, a_2)(a_2\, a_3)\cdots(a_{n-2}\, a_{n-1})(a_{n-1}\, a_n)$. Then according to S the vertex v changes its color from a_1 to a_3 while its neighbors have color a_2, then it changes color from a_3 to a_5 while its neighbors all have color a_4 (so all the neighbors must change their color from a_2 to a_4 before), and so on until v changes its color from a_{n-2} to a_n while its neighbors all have color a_{n-1}.

Let us pick any vertex $q \in V(G)$ and let Q be a reduced walk from $\alpha(q)$ to $\beta(q)$. Furthermore, let $(W_v)_{v \in V(G)}$ be a set of walks from q to each $v \in V(G)$. We say that Q and $(W_v)_v$ *generate* the walks $S_v := \alpha(W_v)^{-1} \cdot Q \cdot \beta(W_v) \in \pi(H)$. We will say that Q is *$\bar{H}$-realizable* for α, β, q if there is a $\bar{H}$-recoloring sequence S satisfying the monochromatic neighborhood property such that $Q = S(q)$. Hence, by Lemma 4, if Q is $\bar{H}$-realizable then S_v

does not depend on W_v. We say that Q is *H-realizable* for α, β, q if there is a H-recoloring sequence S satisfying the monochromatic neighborhood property such that $Q = S(q)$. Clearly, if Q is H-realizable then it is $\bar{H}$-realizable[2].

According to Remark 3, we obtain the next results by following the proofs in [17] nearly word by word.

▶ **Lemma 4** (($*$) see [17, Lemma 4.1]). *Let $S = \sigma_0, \ldots, \sigma_\ell$ be an $\bar{H}$-recoloring sequence from $\alpha = \sigma_0$ to $\beta = \sigma_\ell$ satisfying the monochromatic neighborhood property and let W be any walk in G connecting two vertices u and v. Then $S(v) = \alpha(W)^{-1} \cdot S(u) \cdot \beta(W)$ in $\pi(H)$. In particular, for any set $(W_v)_{v \in V(G)}$ of walks from q to all vertices $v \in V(G)$, if a walk Q from $\alpha(q)$ to $\beta(q)$ is $\bar{H}$-realizable then the other vertex walks in any associated $\bar{H}$-recoloring sequence are the walks $\alpha(W_v)^{-1} \cdot Q \cdot \beta(W_v) \in \pi(H)$.*

The next theorem gives a description of all $\bar{H}$-realizable walks.

▶ **Theorem 5** (see [17, Theorem 8.1]). *Let $\alpha, \beta \colon G \to H$ and $q \in V(G)$. Let $\bar{\Pi}$ be the set of all reduced walks that are $\bar{H}$-realizable for α, β, q. Then one of the following holds:*

1. $\bar{\Pi} = \emptyset$.
2. $\bar{\Pi} = \{Q\}$ *for some* $Q \in \pi(H)$.
3. $\bar{\Pi} = \{R^n P \mid n \in \mathbb{Z}\}$, *for some* $R, P \in \pi(H)$.
4. $\bar{\Pi}$ *contains all reduced walks of even length from $\alpha(q)$ to $\beta(q)$.*

Furthermore, there is an algorithm that determines in time $O(|V(G)| \cdot |E(G)| + |E(H)|)$ which case holds and outputs Q or R, P in cases 2 3 such that $|Q|, |R|, |P|$ are bounded by the total running time $O(|V(G)| \cdot |E(G)| + |E(H)|)$.

The following lemma tells us that from any $\bar{H}$-realizable walk we can obtain in polynomial time an $\bar{H}$-recoloring sequence. We will later generalize it to digraphs, which allows us to construct in polynomial time an H-recoloring sequence from a given H-realizable walk (see Lemma 8).

▶ **Lemma 6** (see [17, Theorem 6.1]). *Given an $\bar{H}$-realizable walk Q we can construct an associated $\bar{H}$-recoloring sequence in time $O(|V(G)|^2 + |V(G)| \cdot |Q|)$.*

3.2 Orientation compatibility

We now characterize $\bar{H}$-recoloring sequences that both satisfy the monochromatic neighborhood property and are compatible with the orientation of H. To this end we give a simple condition, the *zigzag condition*, on the reduced walks of the vertices of G obtained from a given $\bar{H}$-recoloring sequence. We show that it suffices to check the zigzag condition for each vertex of G in order to determine whether a given $\bar{H}$-recoloring sequence is also an H-recoloring sequence. From this we obtain a polynomial-time algorithm that, given two H-colorings $\alpha, \beta : G \to H$ and a vertex v of G, finds a walk from $\alpha(v)$ to $\beta(v)$ in H that is compatible with the orientation of H or reports correctly that no such walk exists. For the remainder of this section let us fix two digraph homomorphisms $\alpha, \beta \colon G \to H$.

[2] This definition generalizes the definition of realizability from [17] such that $\bar{H}$-realizability becomes a necessary condition for H-realizability. When $\bar{H}$ is square-free our definition coincides with the one from [17].

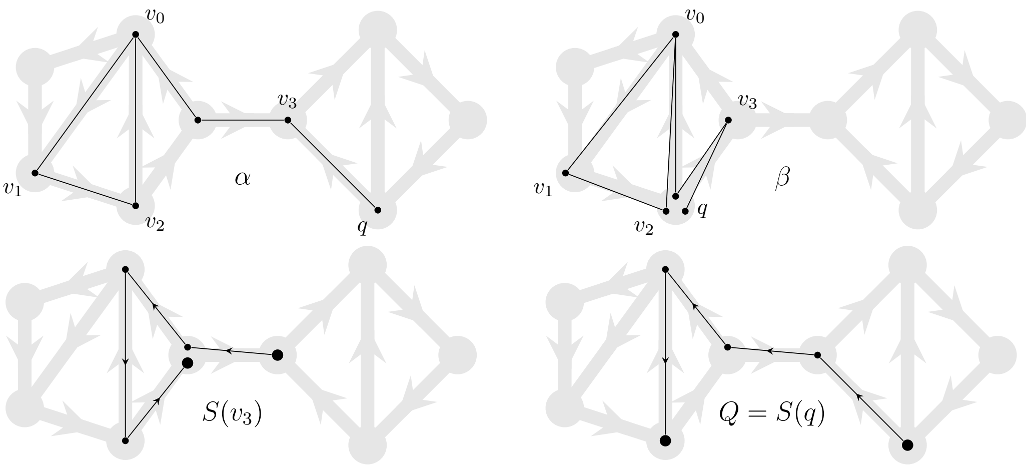


Figure 4 Homomorphisms α and β map a graph G on 6 vertices to the gray graph H. The walk $Q = S(q)$ satisfies the zigzag condition and generates walks that also satisfy the zigzag condition like $S(v_3)$, so it is orientation compatible. We will see soon that the vertices of the triangle $v_0\, v_1\, v_2$ must have symmetric vertex walks, so they cannot change colors for the lack of a symmetric edge in H.

3.2.1 The zigzag condition

Let $v \in V(G)$ and $S_v = (a_1a_2)\dots(a_{n-1}a_n)$ be a walk of even length from $\alpha(v)$ to $\beta(v)$ (think of S_v as the walk of v induced by some $\bar{H}$-recoloring sequence S that satisfies the monochromatic neighborhood property; in particular, S and S_v could have been obtained by combining Theorem 5 and Lemma 6). We say that S_v satisfies the *zigzag condition* if $a_1 \leftarrow a_2 \rightarrow a_3 \leftarrow \dots \leftarrow a_{n-1} \rightarrow a_n$ is a walk in H whenever $N^-_G(v) \neq \emptyset$ (then we also say that S_v zigzags correctly with respect to its *in*-neighborhood) and $a_1 \rightarrow a_2 \leftarrow a_3 \rightarrow \dots \rightarrow a_{n-1} \leftarrow a_n$ is a walk in H whenever $N^+_G(v) \neq \emptyset$. Notice that at least one of $N^-_G(v)$ and $N^+_G(v)$ is non-empty by our assumptions on G. Also, observe that if S_v is not reduced and satisfies the zigzag condition then it will still satisfy that condition after reduction.

Based on Lemma 4 we can state a definition of orientation compatibility for walks: given $\alpha, \beta \colon G \to H$, $q \in V(G)$ and a set of walks (W_v) from q to v. We say that a reduced walk of even length $Q \in \pi(H)$ from $\alpha(q)$ to $\beta(q)$ is *orientation compatible* for the set $(W_v)_{v \in V(G)}$ if for each vertex $v \in V(G)$ and any walk W_v from q to v, the walk $\alpha(W_v)^{-1} \cdot Q \cdot \beta(W_v) \in \pi(H)$ satisfies the zigzag condition. See Figure 4 for illustrations of the zigzag condition and of orientation compatibility. Observe that by Lemma 4, if Q is $\bar{H}$-realizable, then the orientation compatibility of Q does not depend of the choice of the walks $(W_v)_{v \in V(G)}$.

▶ **Lemma 7.** *Let $q \in V(G)$. Let Q be an $\bar{H}$-realizable walk for α, β, q. Then Q is H-realizable if and only if Q is orientation compatible.*

Proof. Let Q be an $\bar{H}$-realizable walk. First assume that Q is H-realizable. Then let S be a H-recoloring sequence from α to β such that $Q = S(q)$. Let $v \in V(G)$ and W_v a walk from q to v in G. Then $\alpha(W_v)^{-1}Q\beta(W_v) = S(v)$ and we can write $S(v)$ as a reduced walk $(a_1a_2)(a_2a_3)\dots(a_{n-1}a_n)$. If $N^-_G(v) \neq \emptyset$ then there is an arc $w \rightarrow v$ for some $w \in V(G)$. At each color change of v from a_i to a_{i+2}, the vertex w must have color a_{i+1} because S satisfies the monochromatic neighborhood property. Since S is a H-recoloring sequence, its homomorphisms all preserve the arc $w \rightarrow v$. Hence $a_1 \leftarrow a_2 \rightarrow \dots \leftarrow a_{n-1} \rightarrow a_n$ is a path in H. Similarly, if $N^+_G(v) \neq \emptyset$ then we have the same path with all arcs reversed. As this holds for each vertex $v \in V(G)$, Q is orientation compatible.

Conversely, if Q is orientation compatible, then let S be the $\bar{H}$-recoloring sequence constructed via Lemma 6. Consider any step σ_i, σ_{i+1} of S, say when a vertex v changes color from a to b while its neighbors have color c. This color change is recorded in $S(u)$ which satisfies the zigzag condition so if σ_i induces a homomorphism $G \to H$, then σ_{i+1} too. By induction, each homomorphism of S induces a homomorphism $G \to H$ so we eventually have a H-recoloring sequence. ◀

The proof of Lemma 7 implies the following generalization of Lemma 6.

▶ **Lemma 8** (∗). *Given an H-realizable walk Q, we can construct an associated H-recoloring sequence in time $O(|V(G)|^2 + |V(G)| \cdot |Q|)$.*

Furthermore, the zigzag condition can be exploited in order to decide efficiently whether a given walk from $\alpha(q)$ to $\beta(q)$ is H-realizable. To show this, we need the following lemma, which is contained in Wrochna's proof of Lemma 6.

▶ **Lemma 9** ([17]). *Given any walk Q from $\alpha(q)$ to $\beta(q)$, we can decide in time $O(|E(G)| \cdot (|Q| + |V(G)|))$ if Q is $\bar{H}$-realizable.*

▶ **Lemma 10** (∗). *There is a polynomial-time algorithm that, given a walk Q from $\alpha(q)$ to $\beta(q)$, decides in time $O(|E(G)| \cdot (|Q| + |V(G)|))$ if Q is H-realizable for α, β, q.*

3.2.2 Characterizing orientation-compatible walks

The following three technical observations (see Definition 11) on the zigzag condition help us to find all orientation compatible walks (for any set of walks $(W_v)_{v \in V(G)}$). More precisely:

1. For any arc $u \to v$, if S_u satisfies the zigzag condition then S_v satisfies the part of the zigzag condition for the *in*-neighborhood (See Lemma 12).
2. It turns out that a walk Q from $\alpha(q)$ to $\beta(q)$ is H-realizable if and only if it satisfies the zigzag condition and generates symmetric walks on a certain set of vertices of G (See Lemma 13).
3. By checking possible reductions for the walks on that set of vertices (see Lemma 15), we obtain the lemma, which provides a description of all orientation compatible walks.

▶ **Definition 11.** *Let Q be an even walk from $\alpha(q)$ to $\beta(q)$. Furthermore, let $v \in V(G)$ and let W_v be a walk from q to v. Suppose that $S_v := \alpha(W_v)^{-1} \cdot Q \cdot \beta(W_v) = (a_1\, a_2) \ldots (a_{n-1}\, a_n)$ is the reduced walk generated on v by Q and W_v. We say that v is of type* IN *if $N_G^-(v) \neq \emptyset$ and it is of type* OUT *if $N_G^+(v) \neq \emptyset$. Furthermore, we say that S_v is* IN-compatible *(resp.,* OUT-compatible*) if v is of type* IN *and additionally $a_1 \leftarrow a_2 \to a_3 \leftarrow \ldots \leftarrow a_{n-1} \to a_n$ is a path in H (resp., v is of type* OUT *and additionally $a_1 \to a_2 \leftarrow a_3 \to \ldots \to a_{n-1} \leftarrow a_n$ is a path of H). Finally, if v is of type* IN *and of type* OUT*, we say that v is of type* SYM *and that S_v is* SYM*-compatible if it is* IN*-compatible and* OUT*-compatible, this means in particular that S_v has only symmetric edges.*

By Lemma 7, we have that an even walk Q is orientation compatible for the set $(W_v)_{v \in V(G)}$ if and only if for every vertex v, if v is of type IN, then S_v is IN-compatible and if v is of type OUT, then S_v is OUT-compatible.

▶ **Lemma 12.** *For any arc $u \to v$ of G, the walk S_u is* OUT*-compatible if and only if S_v is* IN*-compatible.*

Proof. Note that u is of type OUT and v is of type IN. By the monochromatic neighborhood property, if S_u is OUT-compatible then since α and β are homomorphisms, $(\alpha(v)\,\alpha(u))S_u(\beta(u)\,\beta(v))$, is IN-compatible. As S_v is precisely this walk after reduction, it is then IN-compatible. Similarly, if S_v is IN-compatible, then S_u is OUT-compatible. ◀

▶ **Lemma 13** (∗).

1. *If G has no vertex of type SYM and Q satisfies the zigzag condition then S_v satisfies the zigzag condition for all $v \in V(G)$.*
2. *Let $\{v_1, \ldots, v_k\} \subseteq V(G)$ be the subset of vertices of G of type SYM. If S_{v_i} satisfies the zigzag condition for $1 \leq i \leq k$ then S_v satisfies the zigzag condition for all $v \in V(G)$.*

▶ **Lemma 14** (∗). *Let (W_v) be a set of walks from q_0 to all vertices $v \in V(G)$. There is some vertex $q \in V(G)$ such that the set of orientation compatible walks for q and the set $(W_q^{-1}W_v)_{v\in V(G)}$ is one of the followings:*

1. *$\emptyset$.*
2. *$\{Q\}$ for some reduced walk Q of even length $|Q| = O(|V(G)|)$.*
3. *The set of all reduced walks of even length from $\alpha(q)$ to $\beta(q)$ that satisfy the zigzag condition.*

Furthermore, we can determine in time $O(|V(G)| \cdot |E(G)|)$ which case holds, and output Q in Case 2.

▶ **Lemma 15.** *Let $V' \subset V(G)$ be any nonempty set of vertices of G. Let $q \in V'$ and for each $v \in V'$, let W_v be a walk from q to v of length $|W_v| \leq |V(G)|$. Then the set of walks from $\alpha(q)$ to $\beta(q)$ that generate symmetric vertex walks with the set $(W_v)_{v\in V'}$ on V' is one of the following:*

1. *$\emptyset$.*
2. *$\{Q\}$ for some symmetric walk Q of length at most $2|V(G)|$.*
3. *All symmetric walks from $\alpha(q)$ to $\beta(q)$.*

Furthermore, we can decide in time $O(|V(G)| \cdot |E(G)|)$ which case holds and output Q in Case 2.

Proof. Assume that Q and $(W_v)_{v\in V(G)}$ generate symmetric vertex walks. Then, in particular, Q is symmetric. Note that $\alpha(W_v)$ and $\beta(W_v)$ may have non symmetric edges, but since Q generates only symmetric walks and S_v is the reduced walk obtained from $\alpha(W_v)^{-1} \cdot Q \cdot \beta(W_v)$, we have that the non-symmetric edges of $\alpha(W_v)$ and of $\beta(W_v)$ must be the same and appear in the same order. Suppose that $e_1, e_2, \ldots, e_p$ are the non-symmetric edges of $\alpha(W_v)$ and $\beta(W_v)$. Then we can write $\alpha(W_v) = A_1e_1A_2 \ldots A_pe_pA_{p+1}$ and $\beta(W_v) = B_1e_1B_2 \ldots B_pe_pB_{p+1}$, where $A_1, \ldots, A_{p+1}$ and $B_1, \ldots, B_{p+1}$ are symmetric. Note that if $\alpha(W_v)$ and $\beta(W_v)$ are symmetric then $\alpha(W_v) = A_1$ and $\beta(W_v) = B_1$. Since all non-symmetric edges cancel in S_v, we obtain

$$A_p^{-1} \cdots A_2^{-1}e_1A_1^{-1}QB_1e_1B_2 \cdots B_p = \varepsilon \ .$$

So in particular, $Q = A_1B_1^{-1}$ (so $|Q| \leq 2|V(G)|$) and $A_p^{-1} \ldots A_2^{-1}B_2 \ldots B_p = \varepsilon$.

It remains to show that there is an algorithm that decides in time $O(|V(G)| \cdot |E(G)|)$ which case holds and outputs Q in Case 2. The algorithm repeats the following for each vertex $v \in V'$:

- Compute $\alpha(W_v)$ and $\beta(W_v)$ and search for non symmetric edges. If all edges in $\alpha(W_v)$ and $\beta(W_v)$ are symmetric, continue with the next vertex. Else do the next steps.

- Check that $\alpha(W_v)$ and $\beta(W_v)$ have the same non-symmetric edges appearing in the same order. If not, there is no symmetric walk Q that generates a symmetric walk S_v and we may conclude that Case 1 holds. Otherwise, let e_1 and e_2 be respectively the first and the last non-symmetric edge of $\alpha(W_v)$ and $\beta(W_v)$. Decompose $\alpha(W_v) = A_1e_1A_2e_2A_3$ and $\beta(W_v) = B_1e_1B_2e_2B_3$.
- If in some previous iteration a reduced walk Q has been fixed, check that $Q = A_1B_1^{-1}$. If not, then report Case 1. If no walk Q has been fixed in an earlier iteration, fix Q to be the reduced walk $A_1B_1^{-1} \in \pi(H)$.
- Finally, check that $A_p^{-1} \dots A_2^{-1}B_2 \dots B_p = \varepsilon$. If this equality doesn't hold, report Case 1.

Then, report case 2 and Q if some walk Q has been fixed. Otherwise, report case 3.

Each step runs in time $O(|E(G)|)$ and will repeat at most $|V(G)|$ times, so this algorithm runs in time $O(|V(G)| \cdot |E(G)|)$. ◀

We are now able to obtain a description of all H-realizable walks.

▶ **Theorem 16.** *Let $\alpha, \beta \colon G \to H$. We can find in time $O(|V(G)|)$ a vertex $q \in V(G)$ such that the set Π_q of H-realizable walks from $\alpha(q)$ to $\beta(q)$ is one of the following:*

1. $\Pi_q = \emptyset$.
2. $\Pi_q = \{Q\}$ *for some* $Q \in \pi(H)$.
3. $\Pi_q = \{R^nP \mid n \in \mathbb{Z}\}$ *for some* $R, P \in \pi(H)$.
4. Π_q *contains all reduced walks of even length from $\alpha(q)$ to $\beta(q)$ that satisfy the zigzag condition.*

Moreover, we can determine in time $O(|V(G)| \cdot |E(G)| + |E(H)|)$ which case holds and output Q or R, P in cases 2 or 3. In case 4, we can find such a walk in time $O(|E(G)|)$ (or certify there is none).

Theorem 16 will be proved in the next subsection. Combining Lemma 8 and Lemma 10 with Theorem 16, we immediately obtain Theorem 1.

3.2.3 Proof of Theorem 16

In order to prove Theorem 16 we need the following technical lemma.

▶ **Lemma 17.** *Let R_0 be cyclically reduced walk and let P be a reduced walk starting at the base point of R_0. We can find in time $O(|R_0| + |P|)$ an integer n_0 such that none of R_0 and R_0^{-1} entirely cancels with $R^{n_0}P$.*

Recall that for $\alpha, \beta \colon G \to H$ and $q \in V(G)$, we denote by Π_q the set of all walks from $\alpha(q)$ to $\beta(q)$ that are H-realizable. We are now ready to prove Theorem 16.

Proof of Theorem 16. Fix any $q_0 \in V(G)$ and use breadth first search to compute shortest walks W_v from q_0 to v for all $v \in V(G)$ in time $O(|V(G)| \cdot |E(G)|)$. We invoke Lemma 14 for q_0 and $(W_v)_{v \in V}$ to obtain in time $O(|V(G)| \cdot |E(G)|)$ a vertex $q \in V(G)$ and a description of the set of orientation compatible walks for q and $(W_v)_{v \in V(G)}$. We distinguish the possible outcomes:

Case 1 There is no orientation-compatible walk. Then report that there is no H-realizable walk.

Case 2 There is a unique orientation-compatible reduced walk Q of even length. By Lemma 10, we can decide in time $O(|E(G)| \cdot (|Q| + |V(G)|) = O(|V(G)| \cdot |E(G)|)$ if $|Q|$ is H-realizable as $|Q| = O(|V(G)|)$.

Case 3 All reduced walks of even length from $\alpha(q)$ to $\beta(q)$ that satisfy the zigzag condition and are orientation compatible. Invoke Theorem 5 to get in time $O(|V(G)| \cdot |E(G)| + |E(H)|)$ a description of the set $\bar{\Pi}_q$ of all $\bar{H}$-realizable walks from $\alpha(q)$ to $\beta(q)$. We again distinguish the four possible outcomes.

1. $\bar{\Pi}_q = \emptyset$. There is no $\bar{H}$-realizable walk for α, β, q, so there is no H-realizable walk.
2. $\bar{\Pi}_q = \{Q\}$. There is a unique reduced walk Q that is $\bar{H}$-realizable. Then we can check in time $O(|Q|) = O(|E(G)| \cdot |V(G)| + |E(H)|)$ whether it satisfies the zigzag condition and hence is orientation compatible.
3. $\bar{\Pi}_q = \{R^n P \mid n \in \mathbb{Z}\}$ with $R, P \in \pi(H)$ and R closed and of even length. If R does not satisfy the zigzag condition then the following claim allows us to conclude.

   ▷ Claim 1. Suppose that R does not satisfy the zigzag condition. Then at most one of the $\bar{H}$-realizable walks $R^n P$, $n \in \mathbb{Z}$, satisfies the zigzag condition. Furthermore, we can find such a walk in time $O(|V(G)| \cdot |E(G)| + |E(H)|)$ or conclude there is none.

   Proof. Decompose $R = AR_0A^{-1}$ with all walks minimal and R_0 cyclically reduced. Apply Lemma 17 with the walks R_0 and $A^{-1}P$ and obtain in time $O(|R_0| + |P|) = O(|V(G)| \cdot |E(G)| + |E(H)|)$ an integer $n_0 \in \mathbb{Z}$ such that none of R_0^{-1} and R_0 entirely cancels with $R_0^{n_0} A^{-1} P$.
   If A does not satisfy the zigzag condition, then none of $R^n P$ do for $n \neq n_0$, so only $R^{n_0} P$ can possibly satisfy the zigzag condition, which can be checked in time $O(|R^{n_0} P|) = O(|V(G)| \cdot |E(G)| + |E(H)|)$.
   Otherwise, if A satisfies the zigzag condition, then so do the edges of A^{-1}, so R_0 does not satisfy the zigzag condition (since R does not). For $n > n_0 + 1$, $R^n P \in \pi(H)$ contains an entire R_0 that does not reduce, so it cannot satisfy the zigzag condition. Similarly if $n < n_0 - 1$, then $R^n P$ does not satisfy the zigzag condition since it contains an entire R_0^{-1}. Eventually we only need to test $R^{n_0 - 1} P$, $R^{n_0} P$ and $R^{n_0+1} P$, which can be done in time $O(|P|) = O(|V(G)| \cdot |E(G)| + |E(H)|)$. Observe that each edge of R_0 belongs to precisely two of those three walks, so as it is the case for the edges of R_0 that do not fit the zigzag condition, we deduce that at most one of $R^{n_0-1} P$, $R^{n_0} P$ and $R^{n_0+1} P$ satisfies the zigzag condition. ◁

   On the other hand, if R satisfies the zigzag condition then P satisfies the zigzag condition if and only all walks $R^n P$ do. To see this, notice that "badly oriented" edges of P must reduce with "badly oriented" edges of R^n, but there is none in R^n since it is orientation-compatible. So we can again distinguish between Case 1 and Case 3 in time $O(|E(G)| \cdot |V(G)| + |E(H)|)$ and report the result.
4. $\bar{\Pi}_q$ contains all reduced walks of even length from $\alpha(q)$ to $\beta(q)$. We report Case 4. Using breadth first search in the tensor product $G \times K_2$ (we construct the graph $G \times K_2$ by creating for each vertex v_i of G two copies u_i and w_i; two vertices u_i and w_j of $G \times K_2$ are adjacent if and only if v_i and v_j are adjacent in G), we can find such a walk in time $O(|E(G)|)$ or conclude there is none. ◀

4 Reflexive graphs

In this section we sketch the proof of Theorem 2. First, we assume that the graphs G and H are undirected and reflexive. The following property can be thought of as an analogue of the monochromatic neighborhood property for reflexive graphs.

▶ **Definition 18.** *Let $\alpha, \beta : G \to H$ and let S be a H-recoloring sequence from α to β. Then S has the* push-or-pull *property if whenever a vertex $u \in V(G)$ changes its color from a to b then any neighbor $v \in V(G)$ of u has color a or b.*

Intuitively, whenever a vertex u of G changes its color, for any neighbor v of u, we have that u is either "pushed" away from the color of v or "pulled" towards the color v. Notice that if H is triangle-free then any H-recoloring sequence satisfies the push-or-pull property: if a neighbor v of u has a color different from a and b then H contains a triangle.

Let $\alpha, \beta : G \to H$ and let S be an H-recoloring sequence from α to β such that S satisfies the push-or-pull property. We associate to each vertex $u \in V(G)$ the vertex walk $S(u)$ in H corresponding to the successive colors of u according to S. Given a vertex q and a walk Q from $\alpha(q)$ to $\beta(q)$, we say that Q is *H-realizable* for α, β, q if there is an H-recoloring sequence S from α to β such that $Q = S(q)$ and S satisfies the push-or-pull property. By the next lemma, we have that for any vertex $v \in V(G)$, the corresponding walk $S(v)$ generates the walk $S(u)$ of any other vertex by conjugation. Notice that for irreflexive graphs H, the same holds for H-recoloring sequences that satisfy the monochromatic neighborhood property ([17, Lemma 4.1], Lemma 4).

▶ **Lemma 19** (∗). *Let S be a H-recoloring sequence from α to β satisfying the push-or-pull property. Then for any $u, v \in V(G)$ and any u-v walk W, we have $S(v) = \alpha(W)^{-1}S(u)\beta(W)$ in $\pi(H)$.*

Lemma 19 allows us to characterize H-realizable walks based on an algorithm that finds vertices of G whose color cannot change. We end up with the following result which immediately implies that if H is reflexive and triangle-free then H-Recoloring admits a polynomial-time algorithm on reflexive instances, under the condition that if the color of a vertex changes then the old and new colors are neighbors in H (see [13, Theorem 1.1]).

▶ **Theorem 20** (∗). *Let G and H be reflexive undirected graphs and let $\alpha, \beta \colon G \to H$ and $q \in V(G)$. Let $\bar{\Pi}$ be the set of all walks that are H-realizable for α, β, q (in particular, the corresponding H-recoloring sequences satisfy the push-or-pull property). Then one of the following holds:*

1. $\bar{\Pi} = \emptyset$.
2. $\bar{\Pi} = \{Q\}$ *for some* $Q \in \pi(H)$.
3. $\bar{\Pi} = \{R^n P \mid n \in \mathbb{Z}\}$, *for some* $R, P \in \pi(H)$.
4. $\bar{\Pi}$ *contains all reduced walks from* $\alpha(q)$ *to* $\beta(q)$.

Furthermore, we can determine in time $O(|V(G)| \cdot |E(G)| + |E(H)|)$ which case holds and output Q or R, P in cases 2 and 3 such that $|Q|, |R|, |P|$ are bounded by the total running time $O(|V(G)| \cdot |E(G)| + |E(H)|)$. Case 4 happens when $\alpha(C) = \beta(C) = \varepsilon$ in $\pi(H)$ for all closed walks C in G.

In order to obtain Theorem 2, we run the algorithm of Theorem 20 on $\bar{H}$ and check for each of the four cases whether there is a corresponding $\bar{H}$-recoloring sequence that is compatible with the orientation of the arcs of H. In Case 1 there is no H-recoloring sequence since there is no $\bar{H}$-recoloring sequence. To deal with Case 2, we use a greedy-type algorithm ("move-forward algorithm") that first constructs from the walk Q the vertex walks of all other vertices of G using Lemma 19 and then either finds a H-recoloring sequence moving vertices to their next color step-by-step or detects an obstruction in the form of a cyclic dependency of color changes. We essentially reduce Case 3 to Case 2, by showing that it suffices to check the H-realizability of the walks $\{R^n P \mid n \in \mathbb{Z}\}$ only for a polynomial number of values of n. For each value n of interest, we run the move-forward algorithm for the walk $R^n P$. In Case 4, we first compute the set V' of vertices of G that belong to some directed closed walk. We show that the H-realizable walks are exactly those which generate symmetric walks on V'. We use a characterization of such walks in order to conclude whether there is a H-recoloring sequence satisfying the push-or-pull property or not.

Notice that Theorem 20 requires both graphs G and H to be reflexive. The final step in the proof of Theorem 2 is to observe that if H contains no 4-cycle of algebraic girth 0 then we can remove this requirement for the graph G.

5 Conclusion

We showed that H-Recoloring admits a polynomial-time algorithm whenever i) H is a loopless digraph without a 4-cycle of algebraic girth 0 and ii) H is a reflexive digraph containing neither a triangle of algebraic girth 1 nor a 4-cycle of algebraic girth 0. For this purpose we make use of the topological approach developed by Wrochna [17]. Additionally, we showed that all known polynomial-time algorithms for H-Recoloring can be obtained using this approach. That is, in all cases, whenever there is a H-recoloring sequence then in particular all H-colorings of in such a sequence are homotopy-equivalent. This leads to the interesting question, whether homotopy-equivalence is *exactly* the condition that makes H-Recoloring tractable.

References

1. Paul Bonsma and Luis Cereceda. Finding paths between graph colourings: PSPACE-completeness and superpolynomial distances. *Theoretical Computer Science*, 410(50):5215–5226, 2009. doi:10.1016/j.tcs.2009.08.023.
2. Richard C Brewster, Jae-Baek Lee, and Mark H Siggers. Recoloring reflexive digraphs. *Discrete Mathematics*, 341(6):1708–1721, 2018.
3. Richard C Brewster, Jae-Baek Lee, and Mark H Siggers. Reconfiguration of homomorphisms to reflexive digraph cycles. *Discrete Mathematics*, 344(8):112441, 2021. doi:10.1016/j.disc.2021.112441.
4. Richard C Brewster, Sean McGuinness, Benjamin Moore, and Jonathan A Noel. A dichotomy theorem for circular colouring reconfiguration. *Theor. Comput. Sci.*, 639:1–13, 2016. doi:10.1016/j.tcs.2016.05.015.
5. Andrei A Bulatov. A dichotomy theorem for nonuniform CSPs. In *IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 319–330. IEEE, 2017.
6. Luis Cereceda, Jan van den Heuvel, and Matthew Johnson. Finding paths between 3-colorings. *Journal of Graph Theory*, 67(1):69–82, 2011. doi:10.1002/jgt.20514.
7. Anton Dochtermann and Anurag Singh. Homomorphism complexes, reconfiguration, and homotopy for directed graphs, 2021. doi:10.48550/arXiv.2108.10948.
8. Tomás Feder and Moshe Y Vardi. The computational structure of monotone monadic SNP and constraint satisfaction: A study through datalog and group theory. *SIAM J. Comput.*, 28(1):57–104, 1998. doi:10.1137/S0097539794266766.
9. Parikshit Gopalan, Phokion G Kolaitis, Elitza N Maneva, and Christos H Papadimitriou. The connectivity of boolean satisfiability: Computational and structural dichotomies. *SIAM J. Comput.*, 38(6):2330–2355, 2009. doi:10.1137/07070440X.
10. Tesshu Hanaka, Takehiro Ito, Haruka Mizuta, Benjamin Moore, Naomi Nishimura, Vijay Subramanya, Akira Suzuki, and Krishna Vaidyanathan. Reconfiguring spanning and induced subgraphs. *Theor. Comput. Sci.*, 806:553–566, 2020. doi:10.1016/j.tcs.2019.09.018.
11. Takehiro Ito, Erik D Demaine, Nicholas JA Harvey, Christos H Papadimitriou, Martha Sideri, Ryuhei Uehara, and Yushi Uno. On the complexity of reconfiguration problems. *Theoretical Computer Science*, 412(12–14):1054–1065, 2011. doi:10.1016/j.tcs.2010.12.005.

12 Jae-Baek Lee, Jonathan A Noel, and Mark H Siggers. Reconfiguring graph homomorphisms on the sphere. *Eur. J. Comb.*, 86:103086, 2020. doi:10.1016/j.ejc.2020.103086.

13 Jae-Baek Lee, Jonathan A Noel, and Mark H Siggers. Recolouring homomorphisms to triangle-free reflexive graphs. *Journal of Algebraic Combinatorics*, pages 1–21, 2022.

14 Benjamin Lévêque, Moritz Mühlenthaler, and Thomas Suzan. Reconfiguration of digraph homomorphisms. *CoRR*, abs/2205.09210, 2022. doi:10.48550/arXiv.2205.09210.

15 Naomi Nishimura. Introduction to reconfiguration. *Algorithms*, 11(4):52, 2018.

16 Jan van den Heuvel. The complexity of change. In Simon R Blackburn, Stefanie Gerke, and Mark Wildon, editors, *Surveys in Combinatorics 2013*, volume 409. London Mathematical Society Lectures Note Series, 2013.

17 Marcin Wrochna. Homomorphism reconfiguration via homotopy. *SIAM J. Discret. Math.*, 34(1):328–350, 2020. doi:10.1137/17M1122578.

18 Dmitriy Zhuk. A proof of the CSP dichotomy conjecture. *Journal of the ACM*, 67(5):1–78, 2020.

A Proofs omitted from Section 3

Proof of Lemma 4. We use induction on the length ℓ of S. Let $\ell = 1$ and suppose $\sigma_0 \neq \sigma_1$, so a vertex $w \in V(G)$ is recolored from $\sigma_0(w) = a$ to $\sigma_1(w) = b$ and all neighbors of w have color h. By definition, we have $S(w) = (a\,h)(h\,b)$ and $S(v) = \varepsilon$ for $v \in V(G) \setminus \{w\}$. If $W = \varepsilon$ then $\sigma_0(W) = \sigma_1(W) = \varepsilon$ and $S(u) = S(v)$ since $u = v$, so we are done. If W has length one then, without loss of generality, $W = u \to v$. We consider three cases:

- $u \neq w$ and $v \neq w$. Then $S(u) = S(v) = \varepsilon$ and $\sigma_0(W) = \sigma_1(W)$.
- $u \neq w$ and $v = w$. Then $S(u) = \varepsilon$, $S(v) = (ah)(hb)$. Since S satisfies the monochromatic neighborhood property, all neighbors of v, including u, have color h, so $\sigma_0(W) = (h, a)$, $\sigma_1(W) = (h, b)$.
- $u = w$ and $v \neq w$. Then $S(u) = (ah)(hb)$ and $S(v) = \varepsilon$. Again using the fact that S satisfies the monochromatic neighborhood property, we have $\sigma_0(W) = (ah)$ and $\sigma_1(W) = (bh)$.

In each case we have $S(v) = \sigma_0(W)^{-1} S(u) \sigma_1(W)$. If W has length at least two then we may split W inductively into $W = W_1W_2$ such that W_2 is of length one, so W_1 is a walk from u to z and W_2 is a walk from z to v. Then we have

$$\begin{aligned}\sigma_1(W) &= \sigma_1(W_1)\sigma_i(W_2)\\ &= \sigma_1(W_1)S(z)^{-1}\sigma_0(W_2)S(v)\\ &= S(u)^{-1}\sigma_0(W)S(v)\ .\end{aligned}$$

If the sequence S has length more than one we use the same idea and split S inductively into $S = S_1S_2$ such that S_2 has length one. Then for each $v \in V(G)$ we have $S(v) = S_1(v)S_2(v)$ and

$$\begin{aligned}S(v) &= S_1(v)S_2(v)\\ &= S_1(v)\sigma_{\ell-1}(W)^{-1}S_2(u)\sigma_\ell(W)\\ &= \sigma_0(W)^{-1}S(u)\sigma_\ell(W)\ ,\end{aligned}$$

which concludes the proof. ◀

Proof of Lemma 10. Use Lemma 9 to decide in time $O(|E(G)| \cdot (|Q| + |V(G)|))$ whether Q is $\bar{H}$-realizable for α, β, q. If not then Q is not H-realizable for α, β, q. Otherwise, for each vertex $v \in V(G)$, use breadth first search to find a shortest walk W_v from q to v in G in time

$O(|E(G)|)$, then compute $S(v) := \alpha(W_v)^{-1} \cdot Q \cdot \beta(W_v)$, reduce the resulting walk and check the zigzag condition in time $O(|Q| + |V(G)|)$. By Lemma 7, the zigzag condition is satisfied for each vertex $v \in V(G)$ if and only if Q is H-realizable for α, β, q. In total, for each vertex $v \in V(G)$ the computations can be performed in time $O(|Q| + |E(G)|)$. ◀

Proof of Lemma 13. We prove the first statement. Assume there is no vertex of type SYM and that Q satisfies the zigzag condition. Without loss of generality, we may assume that q is of type IN, so Q is IN-compatible. Let v be any other vertex of G and P a path from q to v in G. Since there is no vertex of type SYM we deduce that P is alternating between vertices of IN and vertices of type OUT. By Lemma 12 S_w satisfies the zigzag condition for each vertex w in P. In particular, S_w does.

It remains to prove the second statement. Let $X = \{v_1, \dots, v_k\} \subseteq V(G)$ be the vertices of G of type SYM and suppose that S_{v_i} satisfies the zigzag condition for $1 \leq i \leq k$. Let $v \in V(G)$ be any vertex that is not of type SYM and let P be a shortest path from X to v. Again, P is alternating between vertices of type IN and vertices of type OUT and hence for each vertex w of P, we obtain that S_w satisfies the zigzag condition by Lemma 12. ◀

Proof of Lemma 14. Let $V' \subseteq V(G)$ be the set of vertices of type SYM. Notice that V' can be computed in time $O(|E(G)|)$. Suppose first, suppose that $V' = \emptyset$. Then, by statement A the orientation-compatible walks are precisely those of even length that satisfy the zigzag-condition. We can therefore indicate Case 3 with $q := q_0$. Now suppose that $V' \neq \emptyset$. Let $q \in V'$ and apply Lemma 15 to determine in time $O(|V(G)| \cdot |E(G)|)$ all walks from $\alpha(q)$ to $\beta(q)$ that generate symmetric walks with the set $(W_v)_{v \in V'}$ on all vertices of V'. Invoke the second statement of Lemma 13 to deduce that:

1. In Case 1 of Lemma 15 we report Case 1, i.e., there is no orientation-compatible walk.
2. In Case 2 of Lemma 15 we report Case 2 and output Q if Q has even length and Case 1 otherwise.
3. In Case 3 of Lemma 15 we report Case 3.

The total runtime is dominated by the algorithm of Lemma 15, hence $O(|V(G)| \cdot |E(G)|)$ as claimed. ◀

Proof of Lemma 17. Start with $n = 0$. Check if R_0 entirely reduces with $R_0^n P$ and if so then replace n by $n+1$. Similarly, if R_0^{-1} reduces with $R_0^n P$, then replace n by $n-1$. Repeat until none of R_0^{-1} or R_0 entirely reduces with $R_0^n P$. Each step is done in $|R_0|$ and reduces $|R_0^n P|$ by $|R_0|$, so we deduce that this process terminate in time $O(|R_0| + |P|)$. Also observe that $|n_0|$ is polynomial in $|V(G)|$ and $|V(H)|$. ◀

B Additional results

B.1 Shortest H-recoloring

Let H be a fixed digraph. Given a digraph G and two homomorphism $\alpha, \beta : G \to H$, the problem Shortest H-Recoloring asks for the length of a shortest H-recoloring sequence from α to β (if there is none, the shortest length is ∞). The topological approach, in particular the description of H-realizable walks, is also useful for finding shortest H-recoloring sequences. Again, we may follow Wrochna's arguments from [17] to obtain a polynomial-time algorithm for Shortest H-Recoloring under the conditions of of Theorem 1.

▶ **Theorem 21.** *Let H be a loopless digraph that contains no 4-cycle of algebraic girth 0 as a subgraph. Then Shortest H-Recoloring admits a polynomial-time algorithm.*

We observe that for any H-recoloring sequence, the corresponding walk $S(v)$ of a vertex v (see Section 3.1) is reduced.

▶ **Lemma 22** ([17, see Corollary 6.2]). *Let H be a loopless digraph that contains no 4-cycle of algebraic girth 0 as a subgraph. Let $\alpha, \beta : G \to H$ be graph homomorphisms and $S := \sigma_0, \dots, \sigma_\ell$ a shortest H-recoloring sequence from $\alpha = \sigma_0$ to $\beta = \sigma_\ell$. Then for any vertex $v \in V(G)$, the walk $S(v)$ is reduced.*

Proof. Observe that the length $|S|$ of the H-recoloring sequence S is $|S| = \frac{1}{2}\sum_{v\in V(G)} |S(v)|$. Fix any base vertex $q \in V(G)$ to notice that $Q = S(q)$ is H-realizable. Reducing for $v \in V(G)$ all walks $S(v)$ to $S_r(v)$, Lemma 8 constructs an associated H-recoloring sequence of length $\sum_{v\in V(G)} |S_r(v)|$. Since $S_r(v)$ is the reduction of $S(v)$, we have $|S_r(v)| \leq |S(v)|$. Thus if $|S|$ is minimal then $|S(v)| = |S_r(v)|$ for each $v \in V(G)$ and hence the walks $S(v)$ are reduced. ◀

Following the proof of [17, see Theorem 8.1] nearly word by word and additionally taking care the orientation compatibility yields Theorem 21.

Proof of Theorem 21. By Lemma 22, it suffices to choose a walk $Q \in \Pi'$ from the description in Theorem 16 that minimizes

$$\sum_{v\in V(G)} |S(v)| = \sum_{v\in V(G)} |\alpha(W_v)^{-1} \cdot Q \cdot \beta(W_v)| \ , \tag{1}$$

where W_v is an arbitrary chosen walk from q to v. In cases 1 and 2 of Theorem 16 this is trivial. Recall that in Case 3 we have $Q = R^n \cdot P$ for any $n \in \mathbb{N}$. It is easy to see that $|n| \leq 2|V(G)| + |P|$ in a shortest sequence, since repeating R will eventually increase all summands of (1). It thus suffices to compute (1) for all these choices of n.

In Case 4, consider an H-realizable walk Q, i.e., any reduced walk of even length from $\alpha(q)$ to $\beta(q)$ that satisfies the zigzag condition. Let P_1 be the longest common prefix of Q and $\alpha(W_v) \in \pi(H)$, choosing $v \in V(G)$ to maximize its length. That is, P_1 is longest such that all of P_1 will reduce with $\alpha(W_v)^{-1}$ in some summand of (1). Similarly, let P_2 be the longest common suffix of Q and some $\beta(W_v)^{-1} \in \pi(H)$. Either P_1 and P_2 overlap, or $Q = P_1 Q' P_2$, for some $Q' \in \pi(H)$. In the latter case, since P_1 and P_2 are longest, no element of Q' will reduce in any summand of (1), the sum can be written as

$$\sum_{v\in V(G)} |\alpha(W_v)^{-1} \cdot Q \cdot \beta(W_v)| = \sum_{v\in V(G)} \left(|\alpha(W_v)^{-1} \cdot P_1| + |Q'| + |P_2 \cdot \beta(W_v)|\right).$$

Where we take $\alpha(W_v)^{-1}$ and $\beta(W_v)$ after reduction in $\pi(H)$. Also observe that both P_1 and P_2 must zigzag properly according to the type of q (see Definition 11), i.e., if q is of type IN and $P_1 = (a_1a_2)\dots(a_{n-1}a_n)$, then $a_1 \leftarrow a_2 \rightarrow a_3 \dots$ up to a_n (so the orientation of the last arc depends on parity). Similarly if $P_2 = (b_nb_{n-1})\dots(b_2b_1)$, then $\dots b_3 \leftarrow b_2 \rightarrow b_1$. Thus we can guess P_1 by enumerating all prefixes of all $\alpha(W_v)$ that zigzag properly according to the type of q, similarly guess P_2 and guess how much they overlap. In case they do not overlap, the sum is minimized by taking Q' to be an arbitrary shortest path of appropriate parity, and that zigzags correctly according to the type of q, from the tail of P_1 to the head of P_2 in H. Enumerating all possibilities for (the length of) P_1, P_2 and the overlap can be done in polynomial time, and a shortest path of given parity that zigzags correctly in H can be found by duplicating every vertex, i.e., finding a shortest path that zigzags correctly in the tensor product $H \times K_2$. ◀

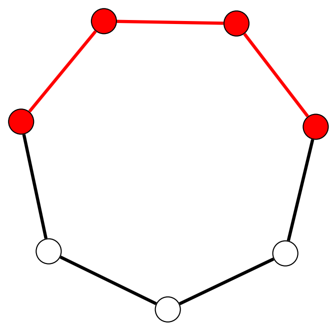


Figure 5 C_p for the circular clique $G_{p,p'}$ with $p = 7$ and $p' = 2$. If $(uv) \in E(G)$ and $\sigma(u) = i$ is the vertex in the bottom, then $\sigma(v)$ must belong to the path in red.

B.2 H-recoloring for circular cliques

Given two integers p, p' such that $p/p' \geq 2$, the circular clique $G_{p,p'}$ has as vertex set the set $\mathbb{Z}_p$ of integers modulo p and has edge set $\{(ij) \mid i - j \bmod p \leq p'\}$. The p-cycle C_p is a graph on the vertex set $\mathbb{Z}_p$ and edge set $\{(i(i+1)) \mid i \in \mathbb{Z}_p\}$. We sketch a proof of the following result by Brewster at al. from [4] using the topological approach of Wrochna.

▶ **Theorem 23** (See [4]). *Let p, p' be fixed positive integers with $2 \leq p/p' < 4$. Let $H := G_{p,p'}$ be the circular clique of parameters p, p'. Then H-Recoloring admits a polynomial-time algorithm.*

By the definition of homomorphisms $G \to G_{p,p'}$ we have for each neighbor u of a vertex $v \in V(G)$, the color of v must belong to a path $P_u(v)$ on C_p which depends on the color of u (see Figure 5). The constraint $p/p' < 4$ is necessary to ensure that the intersection $P(v) = \bigcap_{u \in N(v)} P_u(v)$ is a path on C_p, which is the set of all colors to which the color of v can change. Thus, we can associate vertex walks $S(v)$ in C_p to an H-recoloring sequence S by associating to each color change of v the (unique) walk in $P(v)$ between the two consecutive colors. Also, we convert walks W in $G_{p,p'}$ to walks $\widetilde{W}$ in C_p as follows: replace each edge (ab) in W by the walk $(a\,(a+1))\ldots((b-1)\,b)$ in C_p and concatenate to obtain $\widetilde{W}$. We can prove the following lemma.

▶ **Lemma 24.** *Let S be a $G_{p,p'}$-recoloring sequence from α to β, then for any $u, v \in V(G)$ and any uv walk W, we have $S(v) = \widetilde{\alpha(W)}^{-1} S(u) \widetilde{\beta(W)}$ in $\pi(C_p)$.*

This lemma implies in particular that vertex walks must be topologically valid: for any vertex $q \in V(G)$ and any closed walk C from q to q, $\widetilde{\beta(C)} = S(q)^{-1} \widetilde{\alpha(C)} S(q)$ in $\pi(C_p)$. In particular, the walk of any vertex $q \in V(G)$ generates all the others. Again, we can define that a closed walk $C = (v_0\, v_1) \ldots (v_{n-1}\, v_n)$ from $q = v_0$ to $q = v_n$ is α-tight if $\alpha(v_{i+1}) = \alpha(v_i) + p' \bmod p$ for all $i \in \{0, \ldots, n-1\}$. We say that a walk Q from $\alpha(q)$ to $\beta(q)$ in C_p is realizable if there is a $G_{p,p'}$-recoloring sequence from α to β such that $Q = S(q)$ in $\pi(C_p)$. We obtain again a similar construction theorem the as in the previous sections.

▶ **Theorem 25.** *Let $\alpha, \beta : G \to G_{p,p'}$ be two $G_{p,p'}$-colorings of a graph G. Furthermore, let $q \in V(G)$ be any vertex and let Q be a reduced walk from $\alpha(q)$ to $\beta(q)$. Then Q is realizable for α, β, q if and only if*
1. *Q is topologically valid for α, β, q.*
2. *For each α-tight closed walk in G and any vertex v on this walk, for any walk W from v to q, we have $Q = \widetilde{\alpha(W)}^{-1}\widetilde{\beta(W)}$ in $\pi(H)$.*

Furthermore, there is an algorithm that, given a reduced walk Q, constructs a H-recoloring sequence S from α to β or certifies that Q cannot satisfy one of the previous conditions. This algorithm runs in time $O(|V(G)|^2 + |V(G)| \cdot |Q|)$. The H-recoloring sequence S is such that $S(q) = Q$.

The proof of Theorem 25 is essentially the same that the proof of Theorem 29 in [14] or of Theorem 6.1 in [17]. The only difference is that we will provide recoloring sequences that are longer since the color changes will follow edges in C_p. Using Theorem 5 from [14] (and that an α-tight closed walk can be found in polynomial time if there is one), we directly obtain a description of realizable walks:

▶ **Theorem 26.** *Let $\alpha, \beta \colon G \to H = G_{p,p'}$ and $q \in V(G)$. Let Π be the set of all reduced walks that are realizable for α, β, q. One of the following holds:*
1. *$\Pi = \emptyset$.*
2. *$\Pi = \{Q\}$ for some $Q \in \pi(C_p)$.*
3. *$\Pi = \{R^n P \mid n \in \mathbb{Z}\}$, for some $R, P \in \pi(C_p)$.*

Furthermore, there is an algorithm that determines in time $O(|V(G)| \cdot |E(G)|)$ which case holds and outputs Q or R, P in cases 2, 3 such that $|Q|, |R|, |P|$ are bounded by the time $O(|V(G)| \cdot |E(G)|)$.

Since here $\pi(C_p) \simeq \mathbb{Z}$, it turns out that the case where all walks from $\alpha(q)$ to $\beta(q)$ in C_p are realizable happens in Case 3 when R turns a single time around C_p. So Theorem 23 directly follows. Notice that this result is already proved in [4] together with a dichotomy Theorem for H-recoloring where H-is a circular clique. Our goal here was to show how the topological approach can be exploited even without the monochromatic neighborhood property or the push-or-pull property.

B.3 H-recoloring for transitive tournaments

The transitive tournament T_n is an acyclic orientation of the complete graph on n vertices. Dochtermann [7] showed that when H is a transitive tournament then H-Recoloring is always Yes and a corresponding H-recoloring sequence can be found in polynomial-time. However, we can easily recover this result using the following construction: Say $\{1, \ldots, n\}$ are the vertices of T_n, with $i \to j$ if and only if $i < j$. Let P_n be the undirected path on the same vertices with only the edges $\{i, i+1\}$, $1 \leq i < n$. So for any instance (G, α, β) of H-Recoloring, for each $v \in V(G)$, there is a unique reduce walk S_v in P_n from $\alpha(v)$ to $\beta(v)$ which can be realized (as a vertex walk) by the following special case of the move-forward algorithm (see Lemma 30 in [14]): for each vertex v of G, try to move v onto its next color in S_v and repeat until we reach β. Suppose by contradiction that at any step σ, there is a cycle of obstruction $C = (u_0 u_1) \ldots (u_n u_0)$ where for all i, u_i prevents its neighbor u_i from moving. Suppose without loss of generality that S_{u_0} is increasing in P_n. So $\sigma(u_1) > \sigma(u_0)$, hence $u_0 \to u_1$ and eventually S_{u_1} since it must remain above $\sigma(u_0)$. So by induction, all walks S_{u_0} are increasing and we have a directed cycle $u_0 \to u_1 \to \ldots \to u_n \to u_0$ in G, which is impossible.

The homotopy group $\pi(P_n)$ being trivial, it is not necessary to define topological validity in this case.