

Feedback Vertex Set Reconfiguration in Planar Graphs

Nicolas Bousquet^a, Felix Hommelsheim^b, Yusuke Kobayashi^c, Moritz
Mühlenthaler^d, Akira Suzuki^e

^a*Univ. Lyon, Université Lyon 1, LIRIS, CNRS, UMR 5205, Lyon, France*

^b*Faculty of Mathematics and Computer Science, University of Bremen, Germany*

^c*Research Institute for Mathematical Sciences, Kyoto University, Japan*

^d*Laboratoire G-SCOP, Université Grenoble Alpes, France*

^e*Graduate School of Information Sciences, Tohoku University, Japan*

Abstract

We study the complexity of deciding whether for two given feedback vertex sets of a graph there is a step-by-step transformation between them, such that for each feedback vertex set in the transformation, the next one is obtained by exchanging a single vertex. We give a classification of the complexity of this question for planar graphs in terms of the maximum degree. We show that for planar graphs of maximum degree at most three the problem is tractable because there always exists a transformation, while it is PSPACE-complete when the maximum degree is at most four. The positive side of the classification extends to $K_{3,3}$ -minor-free graphs of maximum degree three. We then consider the Matroid Parity problem, which generalizes feedback vertex sets in graphs of maximum degree three as well as matchings and spanning trees in general graphs. Generalizing known results for the latter two we show that there always exists a transformation between any two non-maximum independent parity sets.

Keywords: feedback vertex set, combinatorial reconfiguration,
polynomial-time algorithm, planar graph, Matroid Parity
2000 MSC: 05C85

Email addresses: nicolas.bousquet@cnrs.fr (Nicolas Bousquet),
fhommels@uni-bremen.de (Felix Hommelsheim), yusuke@kurims.kyoto-u.ac.jp
(Yusuke Kobayashi), moritz.muhlenthaler@grenoble-inp.fr (Moritz Mühlenthaler),
akira@tohoku.ac.jp (Akira Suzuki)

1. Introduction

Many algorithmic questions about reconfiguration problems are of the following form: given the description of a system state and the description of a state that is more desirable, is it possible to transform the system from its current state into the more desired one without “breaking” the system in the process? Such questions, with some generalizations and specializations, have received a substantial amount of attention under the so-called *combinatorial reconfiguration framework*. Historically, the study of reconfiguration questions predates the field of computer science, as many classic one-player games can be formulated as reconfiguration questions [1, 2], e.g., the 15-puzzle and Rubik’s cube. Recently, graph reconfiguration problems have received considerable attention, in particular the reconfiguration of colorings, independent sets, and matchings.

Two main types of questions have been investigated in graph reconfiguration: reachability and connectivity. Given two configurations of a system, the reachability problem asks whether it is possible to transform the first into the second via a sequence of elementary changes, keeping a feasible configuration at each step. The connectivity problem asks whether such a transformation exists for any two configurations of a given system. We refer the reader to the surveys by van den Heuvel [3] and Nishimura [4] for more background on combinatorial reconfiguration. In this paper, we investigate the complexity of both questions for feedback vertex sets of a graph.

A *feedback vertex set* (FVS) of a graph is a subset of the vertices whose deletion results in an acyclic graph. Deciding whether a graph has a feedback vertex set of a given size is an NP-complete problem even for planar graphs of maximum degree four [5]. Interestingly, for graphs of maximum degree at most three, also called *subcubic* graphs, a minimum FVS can be computed in polynomial time by a reduction to a tractable special case of the Matroid Parity problem [6, 7]. Matroid Parity is a generalization of spanning trees and matchings of a graph (see Section 4 for formal definitions). For these two special cases, the corresponding reachability and connectivity problems admit polynomial-time algorithms; even the complexity of finding shortest transformations is well understood [8, 9]. The main motivation of this paper is to investigate whether these results can be generalized to Matroid Parity.

For a graph G and $k \in \mathbb{Z}_{\geq 0}$ let $\mathcal{C}_k(G)$ be the graph whose vertices are feedback vertex sets of G of size k . Two vertices W and W' of $\mathcal{C}_k(G)$ are *adjacent* if their symmetric difference has size exactly two. That is, the

FVS W' can be obtained from W by removing one vertex from $W \setminus W'$ and adding one vertex from $W' \setminus W$. We will refer to the graph $\mathcal{C}_k(G)$ as *reconfiguration graph* and a path from W to W' in $\mathcal{C}_k(G)$ as *reconfiguration sequence* (between W and W'). Given a graph G , an integer $k \geq 0$, and two FVSs W and W' of G , each of size k , the problem FVS-RECONFIGURATION asks whether there is a reconfiguration sequence between W and W' in $\mathcal{C}_k(G)$.

Results. Our main result is a complexity classification of FVS-RECONFIGURATION on planar graphs based on the maximum degree of the input graph G . As for the problem of deciding whether a graph has a FVS of at most a given size [10], the complexity gap occurs between maximum degree three and four.

Theorem 1. *FVS-RECONFIGURATION on planar graphs of maximum degree at most three admits a polynomial-time algorithm and is PSPACE-complete on planar graphs even when the maximum degree is at most four.*

In order to show the positive part of Theorem 1, we prove the following stronger result.

Theorem 2. *Let G be a $K_{3,3}$ -minor-free subcubic graph and $k \in \mathbb{Z}_{\geq 0}$. Then any two FVSs of G of size k admit a reconfiguration sequence and such a sequence can be computed in polynomial time.*

The proof of Theorem 2 yields a transformation of length at most $7k$ (see Section 3.4). Note that excluding $K_{3,3}$ -minors is necessary in Theorem 2 since $\mathcal{C}_2(K_{3,3})$ is not connected. On the other hand, there exist graphs G containing $K_{3,3}$ as a minor such that $\mathcal{C}_k(G)$ is connected, where k is the feedback vertex number of G . For example, when G_1 is the graph constructed in Section 6 with $t = 1$, it is easy to prove that the feedback vertex number is four and $\mathcal{C}_4(G_1)$ is connected.

Using the same ideas as in the proof of Theorem 2 we show that for any subcubic graph G that contains k disjoint cycles the graph $\mathcal{C}_k(G)$ is connected (Theorem 11). This result is tight in the sense that $k - 1$ disjoint cycles are not sufficient; again $\mathcal{C}_2(K_{3,3})$ is an example.

The hardness part of Theorem 1 is based on a reduction from NON-DETERMINISTIC CONSTRAINT LOGIC (NCL), which actually yields the following stronger result.

Theorem 3. FVS-RECONFIGURATION is PSPACE-complete on planar graphs of bounded bandwidth and maximum degree Δ for any $\Delta \geq 4$.

FVS-RECONFIGURATION on subcubic graphs can be seen as a special case of MATROID PARITY RECONFIGURATION (see Section 4 for the definition), and so it is natural to expect our positive result (Theorem 2) is extendable to MATROID PARITY RECONFIGURATION. However, this problem is hard in general. Indeed, since the reduction in [11] gives an exact correspondence between the independent sets of size k of a given graph and the independent parity sets of a Matroid Parity instance, the PSPACE-completeness of Independent Set Reconfiguration [12] implies that, in general, MATROID PARITY RECONFIGURATION is PSPACE-complete. Nevertheless, the following interesting question remains open, where Theorem 2 as well as the polynomial-time algorithms for Matching Reconfiguration and Matroid Basis Reconfiguration [9] only provide partial answers.

Question 1. For special cases of MATROID PARITY RECONFIGURATION, determine the polynomial solvability of the problem.

We provide a positive answer to Question 1 for non-maximum independent parity sets of general matroids (Theorem 15), which implies that the reconfiguration graph of non-minimum FVSs of a subcubic graph is connected.

Theorem 4. Let G be a subcubic graph and let $W_{\text{ini}}, W_{\text{tar}} \subseteq V(G)$ be feedback vertex sets of G with $|W_{\text{ini}}| = |W_{\text{tar}}|$. If G has a feedback vertex set of size at most $|W_{\text{ini}}| - 1$ then W_{ini} and W_{tar} admit a reconfiguration sequence.

Theorem 4 ensures that the reconfiguration graph of non-minimum FVS is connected, but the complexity of the problem for minimum FVS of subcubic graphs remains open.

An edge e of a connected graph is a *bridge* if $G - e$ has at least two connected components. Since a bridge appears in no cycle, for FVS reconfiguration we may restrict our attention to bridgeless graphs. Since a 2-edge-cut of a bridgeless cubic graph generally leaves 3-connected components (which can easily be transformed into cubic graphs), the crucial case is that of 3-connected cubic graphs. Speckenmeyer showed in [5] that any cubic graph with a feedback vertex set of size k has at most $4k - 2$ vertices. Assume now that the number of vertices (as a function of k) is much smaller. That is, the size of a minimum feedback vertex set is “large” in comparison to the total

number of vertices. We show that even in this setting the reconfiguration graph may be disconnected.

Theorem 5. *For any odd integer $k \geq 5$ there is a 3-connected cubic graph on $3k + 3$ vertices such that $\mathcal{C}_k(G)$ is disconnected.*

We do not know what is the maximum α such that, for any 3-connected cubic graph on $\alpha k + \beta$ vertices (for some constant β), $\mathcal{C}_k(G)$ is connected.

Related work. FVS-RECONFIGURATION can be seen as reconfiguration of sets whose deletion leaves an acyclic graph (a *forest*), that is, a graph of treewidth one. The reconfiguration of a set whose deletion leaves a graph of treewidth zero, that is a vertex cover, as well as its dual problem independent set reconfiguration, received considerable attention in the last decade, see e.g. [13, 14, 9, 15] (or the survey [4] for exhaustive results).

In the last few years, reconfiguration of independent sets of matroids with additional constraints has been studied. One example is independent sets of graphic matroids, which correspond to edge-sets of acyclic subgraphs of a given graph. In [16, 17], the authors studied the impact of adding some constraints on the complexity of the problem.

Organization. In Section 2 we give basic definitions and notation. Section 3 contains polynomial-time algorithms for FVS-RECONFIGURATION on subcubic $K_{3,3}$ -minor-free graphs (Theorem 2) and graphs that contain k disjoint cycles. In Section 4 we show that MATROID PARITY RECONFIGURATION admits a polynomial-time algorithm for non-maximum independent parity sets and prove Theorem 4. In Section 5 we show that FVS-RECONFIGURATION is PSPACE-complete on graphs of maximum degree at least four (Theorem 3). Finally, in Section 6 we give a construction of 3-connected cubic graphs with $3k + 3$ vertices whose k -reconfiguration graphs are disconnected (Theorem 5).

2. Notation

Throughout this paper we consider undirected graphs without loops and parallel edges. Let G be a graph. The graph G is *subcubic* (resp., *cubic*) if each vertex has degree at most three (resp., exactly three). We denote by $V(G)$ (resp., $E(G)$) the vertex set (resp., edge set) of G . Let $V' \subseteq V(G)$. The graph $G[V']$ is the subgraph of G induced by the vertex set V' , that is, the vertex set of $G[V']$ is V' and the edge set is $\{vw \mid vw \in E(G) \text{ and } v, w \in V'\}$.

For $W \subseteq V(G)$, we denote $G - W = G[V(G) \setminus W]$. For a set $W \subseteq V(G)$ and a vertex $v \in V(G)$, we use the shorthands $W + v$, $W - v$, $G - v$ for $W \cup \{v\}$, $W \setminus \{v\}$, and $G - \{v\}$, respectively.

A vertex set $W \subseteq V(G)$ is called *feedback vertex set* (FVS) of G if $G - W$ is acyclic (a forest). The *feedback vertex number* of G is the smallest integer k such that G admits a feedback vertex set of size k . It will be convenient for us to refer to the vertices of a FVS as *tokens*. Clearly, every cycle in G contains at least one token of a FVS. We say that two FVSs W and W' of a graph are *adjacent* if we obtain W' from W by moving a token on a vertex in $W \setminus W'$ to a vertex in $W' \setminus W$. Such a token movement is called “token jumping” in the context of combinatorial reconfiguration. An instance of FVS-RECONFIGURATION is denoted by a triple $(G, W_{\text{ini}}, W_{\text{tar}})$, where G is a graph and W_{ini} and W_{tar} are the initial and target FVSs with $|W_{\text{ini}}| = |W_{\text{tar}}|$. A reconfiguration sequence consisting of a single FVS is called a *trivial reconfiguration sequence*.

3. $K_{3,3}$ -minor free graphs

A graph H is a *minor* of a graph G if it can be obtained from G by contracting edges and deleting edges or vertices. The goal of this section is to prove that any two FVSs of a $K_{3,3}$ -minor-free subcubic graph admit a reconfiguration sequence (Theorem 2). That is, any instance of FVS-RECONFIGURATION on a $K_{3,3}$ -minor-free graph is trivial in the sense that the answer is always YES. In the following we describe a polynomial-time algorithm that computes a reconfiguration sequence.

3.1. Reductions

Let $I = (G, W_{\text{ini}}, W_{\text{tar}})$ be an instance of FVS-RECONFIGURATION such that the graph G is subcubic. In our algorithm that computes a reconfiguration sequence between W_{ini} and W_{tar} , we exhaustively apply the reductions (R1)–(R7) below in this order of priority if G has at least two vertices. In each reduction, we construct a new instance $I' = (G, W'_{\text{ini}}, W'_{\text{tar}})$ from I . We show that each reduction is *valid*, that is, I' is an instance of FVS-RECONFIGURATION on a subcubic graph, and that if we are given a reconfiguration sequence for I' then we can obtain one for I by a slight modification of the FVSs and by adding a few more steps, where such additional steps are called *extra moves*. We will see later in Proposition 6 that if none of the reductions (R1)–(R7) is applicable then G is cubic and $W_{\text{ini}} \cap W_{\text{tar}} = \emptyset$.

(R1) If there exists a vertex $v \in W_{\text{ini}} \cap W_{\text{tar}}$, then delete v . That is, let $I' = (G - v, W_{\text{ini}} - v, W_{\text{tar}} - v)$.

Since $G - v$ is subcubic and $|W_{\text{ini}} - v| = |W_{\text{tar}} - v|$, I' is an instance of FVS-RECONFIGURATION in a subcubic graph. We also see that a reconfiguration sequence for I' can be converted to one for I by adding v to each FVS. This shows the validity of (R1).

(R2) If there exists a vertex $v \in W_{\text{ini}}$ such that $W_{\text{ini}} - v$ is a FVS of G , then choose an arbitrary vertex w in W_{tar} , let $W'_{\text{ini}} = W_{\text{ini}} - v + w$, and let $I' = (G, W'_{\text{ini}}, W_{\text{tar}})$. Similarly, if there exists a vertex $v \in W_{\text{tar}}$ such that $W_{\text{tar}} - v$ is a FVS of G , then choose an arbitrary vertex w in W_{ini} , let $W'_{\text{tar}} = W_{\text{tar}} - v + w$, and let $I' = (G, W_{\text{ini}}, W'_{\text{tar}})$.

By symmetry, we show the validity of (R2) only for the case when there exists a vertex $v \in W_{\text{ini}}$ such that $W_{\text{ini}} - v$ is a FVS of G . Since $w \notin W_{\text{ini}}$ by (R1), it holds that W'_{ini} is a FVS of G with $|W'_{\text{ini}}| = |W_{\text{tar}}|$. Since W_{ini} is adjacent to W'_{ini} , a reconfiguration sequence for I' can be extended to one for I by adding one extra move, which shows the validity of (R2). Note that, if we apply (R2), then (R1) must be applied in the next step.

(R3) If G contains a vertex v of degree at most one, then let $I' = (G - v, W_{\text{ini}}, W_{\text{tar}})$.

Since (R2) is not applied and v is contained in no cycles, we see that $v \notin W_{\text{ini}}$ and $v \notin W_{\text{tar}}$. Thus, W_{ini} and W_{tar} are FVSs of $G - v$ with $|W_{\text{ini}}| = |W_{\text{tar}}|$. Since any FVS of $G - v$ is also a FVS of G , a reconfiguration sequence for I' can be regarded as a reconfiguration sequence for I . This shows the validity of (R3).

(R4) If G has a connected component that forms a cycle C , then let v (resp. w) be the unique vertex in $W_{\text{ini}} \cap V(C)$ (resp. $W_{\text{tar}} \cap V(C)$), and let $I' = (G, W'_{\text{ini}}, W_{\text{tar}})$, where $W'_{\text{ini}} = W_{\text{ini}} - v + w$.

In this reduction, we note that $W_{\text{ini}} \cap V(C)$ (resp. $W_{\text{tar}} \cap V(C)$) has a unique vertex, because W_{ini} (resp. W_{tar}) is a FVS of G and (R2) is not applied. Therefore, W'_{ini} is well-defined and is a FVS of G with $|W'_{\text{ini}}| = |W_{\text{tar}}|$. Since W_{ini} is adjacent to W'_{ini} , a reconfiguration sequence for I' can be extended to one for I by adding one extra move, which shows the validity of (R4). Note that, if we apply (R4), then (R1) must be applied in the next step.

(R5) If W_{ini} contains a vertex v of degree two, then let $w \in V$ be a vertex of degree three that is closest to v in the following sense: there exists a path P connecting v and w such that each vertex in $V(P) - w$ has degree two. Then, let $I' = (G, W'_{\text{ini}}, W_{\text{tar}})$, where $W'_{\text{ini}} = W_{\text{ini}} - v + w$. Similarly, if W_{tar} contains a vertex v of degree two, then let $w \in V$ be a vertex of degree three that is closest to v and let $I' = (G, W_{\text{ini}}, W'_{\text{tar}})$, where $W'_{\text{tar}} = W_{\text{tar}} - v + w$.

By symmetry, we show the validity of (R5) only for the case when W_{ini} contains a vertex v of degree two. The existence of $w \in V$ in the reduction rule is guaranteed by (R3) and (R4). Observe that any cycle containing v also contains w . This observation shows that W'_{ini} is a FVS of G with $|W'_{\text{ini}}| = |W_{\text{tar}}|$, where we note that $w \notin W_{\text{ini}}$ by (R2). Since W_{ini} is adjacent to W'_{ini} , a reconfiguration sequence for I' can be extended to one for I by adding one extra move, which shows the validity of (R5).

(R6) If G contains a cycle C such that $|V(C)| = 3$ and C has a vertex u of degree two, then let v (resp. w) be the unique vertex in $W_{\text{ini}} \cap V(C)$ (resp. $W_{\text{tar}} \cap V(C)$). Then, let $I' = (G, W'_{\text{ini}}, W_{\text{tar}})$, where $W'_{\text{ini}} = W_{\text{ini}} - v + w$.

We note that $W_{\text{ini}} \cap V(C)$ (resp. $W_{\text{tar}} \cap V(C)$) has a unique vertex, because $|W_{\text{ini}} \cap V(C)| \geq 1$ and $|W_{\text{tar}} \cap V(C)| \geq 1$ hold and neither (R1) nor (R5) is applied. Observe that any cycle intersecting $V(C)$ contains both v and w , because G is subcubic and u is of degree two. This observation shows that W'_{ini} is a FVS of G with $|W'_{\text{ini}}| = |W_{\text{tar}}|$. Since W_{ini} is adjacent to W'_{ini} , a reconfiguration sequence for I' can be extended to one for I by adding one extra move, which shows the validity of (R6). Note that, if we apply (R6), then (R1) must be applied in the next step.

(R7) If G contains a vertex v of degree two, then let $w, w' \in V$ be the distinct vertices adjacent to v . Let G' be the graph obtained from G by removing v and adding a new edge ww' . Let $I' = (G', W_{\text{ini}}, W_{\text{tar}})$.

We note that G' contains no parallel edges, because w and w' are not adjacent by (R6). Note also that $v \notin W_{\text{ini}}$ and $v \notin W_{\text{tar}}$ by (R5). Since any FVS of G' is also a FVS of G , a reconfiguration sequence for I' can be regarded as a reconfiguration sequence for I . This shows the validity of (R7).

The following proposition summarizes the properties of the reductions.

Proposition 6. *If $I' = (G', W'_{\text{ini}}, W'_{\text{tar}})$ is obtained from $I = (G, W_{\text{ini}}, W_{\text{tar}})$ by applying one of (R1)–(R7), then we have the following:*

1. *I' is an instance of FVS-RECONFIGURATION in a subcubic graph,*
2. *if we are given a reconfiguration sequence for I' then we can obtain one for I in polynomial time,*
3. *$|V(G')| + |W'_{\text{ini}} \setminus W'_{\text{tar}}| < |V(G)| + |W_{\text{ini}} \setminus W_{\text{tar}}|$, and*
4. *if G is $K_{3,3}$ -minor-free, then so is G' .*

Furthermore, if G has at least two vertices and none of (R1)–(R7) is applicable then G is a cubic graph and $W_{\text{ini}} \cap W_{\text{tar}} = \emptyset$.

Proof. We have already shown the first two conditions, and we can easily check the third condition. Since G' is obtained from G by deleting a vertex or contracting an edge, the fourth condition holds. If neither (R3) nor (R7) is applied, then G is a cubic graph. Furthermore, if (R1) is not applied then $W_{\text{ini}} \cap W_{\text{tar}} = \emptyset$. This completes the proof. $\square$

3.2. Useful lemmas

In order to prove Theorem 2, we begin with the following lemma, which is due to Speckenmeyer [5]. We give a proof for completeness.

Lemma 7. *For a cubic graph G that admits a FVS W of size k we have $|V(G)| \leq 4k - 2$.*

Proof. Note that $|V(G)| \neq 4k - 1$ since each vertex of G has odd degree and the number of odd-degree vertices in a graph is even. Assume for a contradiction that $|V(G)| \geq 4k$. Since G is cubic, the number of edges of G is precisely $\frac{3}{2} \cdot |V(G)|$. Let $G' = G - W$. Then, we obtain

$$\begin{aligned}
 |E(G')| &\geq |E(G)| - 3 \cdot |W| \\
 &= \frac{3}{2} \cdot |V(G)| - 3k \\
 &= |V(G)| - k + \frac{1}{2}(|V(G)| - 4k) \\
 &\geq |V(G)| - k \\
 &= |V(G')|.
 \end{aligned}$$

This shows that G' contains a cycle, a contradiction to the fact that W is a FVS. $\square$

Let W be a FVS of G . For $v \in W$ let $\pi_W(v)$ denote the set of vertices to which the token on v can be moved under the condition that the resulting token positions form a FVS (and no other token is moved). Formally, we have $\pi_W(v) := \{w \in V \setminus W \mid W - v + w \text{ is a FVS of } G\}$. For any $X \subseteq W$ we write $\pi_W(X)$ for $\bigcup_{v \in X} \pi_W(v)$. For an instance $I = (G, W_{\text{ini}}, W_{\text{tar}})$ we use the shorthands $\pi_{\text{ini}} = \pi_{W_{\text{ini}}}$ and $\pi_{\text{tar}} = \pi_{W_{\text{tar}}}$.

Lemma 8. *Let $G = (V, E)$ be a cubic graph and let W be a FVS of G . Then, for any $v \in W$, we have one of the following: $\pi_W(v) = V \setminus W$, $\pi_W(v) = V(C_v) - v$ for some cycle C_v , or $|\pi_W(v)| = 1$.*

Proof. Let $v \in W$. Consider the graph $G_v = G - (W - v)$, i.e., the graph obtained from G by deleting all vertices in W except v . Clearly, since W is a FVS, all the cycles in G_v have to go through v and then must contain two neighbors of v . That is, any cycle in G_v contains v plus a path between two neighbors of v . Let v_1, v_2, v_3 be the neighbors of v in G and let us denote by P_1, P_2, P_3 the paths from v_1 to v_2 , from v_2 to v_3 , and from v_3 to v_1 in $G_v - v$, respectively (if they exist). Note that these paths are uniquely determined since $G_v - v$ is a forest. We consider the following cases separately.

- If none of P_1, P_2 , and P_3 exists, then any vertex in $V \setminus W$ is in $\pi_W(v)$, that is, $\pi_W(v) = V \setminus W$.
- If exactly one of P_1, P_2 , and P_3 exists, then G_v contains a unique cycle, say C_v , and hence $\pi_W(v) = V(C_v) - v$.
- Otherwise, we may assume that P_1 and P_2 exist. Then, P_3 also exists, because the union of P_1 and P_2 contains a path from v_3 to v_1 . Since the union of P_1, P_2 , and P_3 forms a tree as $G_v - v$ is acyclic, there exists a unique vertex w contained in all of these three paths. This shows that $\pi_W(v) = \{w\}$, and hence $|\pi_W(v)| = 1$.

By these three cases, the lemma holds. $\square$

Lemma 9. *Let $G = (V, E)$ be a cubic graph and let W be a FVS of G . If there are two vertices $v_1, v_2 \in W$ such that $\pi_W(v_1) = \pi_W(v_2) = \{u\} \subsetneq V \setminus W$ then G contains a $K_{3,3}$ -minor.*

Proof. Suppose there are two vertices $v_1, v_2 \in W$ such that $\pi_W(v_1) = \pi_W(v_2) = \{u\} \subsetneq V \setminus W$.

Let $u_1, u_2, u_3 \in V$ be the neighbors of u . Consider the graph $G_{v_1} = G - (W - v_1)$. Notice that every cycle of G_{v_1} passes through v_1 and u . For $1 \leq i \leq 3$, let P_i be the path from v_1 to u_i in $G_{v_1} - u$ (if they exist). Note that each P_i is uniquely determined as $G_{v_1} - u$ is a forest. Note also that P_1, P_2 , and P_3 exist since otherwise $\pi_W(v_1) = V \setminus W$ or $\pi_W(v_1) = V(C) - v_1$ for some cycle C , which contradicts $\pi_W(v_1) = \{u\} \subsetneq V \setminus W$. This implies that u_1, u_2 , and u_3 are not contained in $W - v_1$, and hence $v_2 \neq u_i$ for $1 \leq i \leq 3$. Similarly, we consider for $1 \leq i \leq 3$ the paths Q_i from v_2 to u_i in $G_{v_2} - u$, where $G_{v_2} = G - (W - v_2)$. The existence of Q_i shows that $v_1 \neq u_i$ for $1 \leq i \leq 3$.

For distinct $i, j \in \{1, 2, 3\}$, if P_i and P_j have a common vertex w that is different from v_1 , then the concatenation of P_i from u_i to w and P_j from w to u_j plus the vertex u gives a cycle that does not contain any vertex of W , a contradiction. Therefore, P_1, P_2 , and P_3 are mutually vertex disjoint except for v_1 . Similarly, Q_1, Q_2 , and Q_3 are mutually vertex disjoint except for v_2 . We also see that, for distinct $i, j \in \{1, 2, 3\}$, if P_i and Q_j have a common vertex w , then the concatenation of P_i from u_i to w and Q_j from w to u_j plus the vertex u gives a cycle that does not contain any vertex of W , a contradiction. Therefore, P_i and Q_j are vertex disjoint for $i \neq j$.

For each $1 \leq i \leq 3$, let $X_i = (V(P_i) \cup V(Q_i)) \setminus \{v_1, v_2\}$. We see that X_1, X_2 , and X_3 are disjoint by the above argument. Furthermore, each X_i induces a connected subgraph as $u_i \in V(P_i) \cap V(Q_i)$. Then, by contracting all the edges in P_i and Q_i that are not incident to v_1 or v_2 , we obtain a $K_{3,3}$ whose vertices correspond to v_1, v_2, u, X_1, X_2 , and X_3 . This shows that G contains a $K_{3,3}$ -minor. $\square$

3.3. Proof of Theorem 2

To prove Theorem 2, we consider two more reduction steps which decrease the size of the symmetric difference of W_{ini} and W_{tar} . Let $I = (G, W_{\text{ini}}, W_{\text{tar}})$ be an instance of FVS-RECONFIGURATION such that $W_{\text{ini}} \cap W_{\text{tar}} = \emptyset$.

- (G) If there exist vertices $v \in W_{\text{ini}}$ and $w \in W_{\text{tar}}$ such that $w \in \pi_{\text{ini}}(v)$, then consider a new instance $I' = (G, W'_{\text{ini}}, W_{\text{tar}})$, where $W'_{\text{ini}} = W_{\text{ini}} - v + w$. Similarly, if there exist vertices $v \in W_{\text{ini}}$ and $w \in W_{\text{tar}}$ such that $v \in \pi_{\text{tar}}(w)$, then let $I' = (G, W_{\text{ini}}, W'_{\text{tar}})$, where $W'_{\text{tar}} = W_{\text{tar}} - w + v$.

This operation is called a *greedy-move*. We see that, from a reconfiguration sequence for I' , we can easily construct a reconfiguration sequence for I by adding one extra move.

Algorithm 1: FVS-RECONFIGURATION algorithm

input : Subcubic $K_{3,3}$ -minor-free graph $G = (V, E)$, FVSs W_{ini} and W_{tar} of the same size.
output: Reconfiguration sequence from W_{ini} to W_{tar} .
1 **if** $|V| = 1$ **then**
2 **return** trivial reconfiguration sequence
3 **else**
4 Apply one of (R1)–(R7), (G), and (D) in this order of priority to obtain a new instance $I' = (G', W'_{\text{ini}}, W'_{\text{tar}})$
5 Compute a reconfiguration sequence σ' for I' recursively
6 Convert σ' into a reconfiguration sequence σ for $(G, W_{\text{ini}}, W_{\text{tar}})$
7 **return** σ

(D) If there exist vertices $v \in W_{\text{ini}}$, $w \in W_{\text{tar}}$, and $u \in V \setminus (W_{\text{ini}} \cup W_{\text{tar}})$ such that $u \in \pi_{\text{ini}}(v) \cap \pi_{\text{ini}}(w)$, then consider a new instance $I' = (G, W'_{\text{ini}}, W'_{\text{tar}})$, where $W'_{\text{ini}} = W_{\text{ini}} - v + u$ and $W'_{\text{tar}} = W_{\text{tar}} - w + u$.

This operation is called a *detour-greedy-move*. We see that the size of the symmetric difference of W'_{ini} and W'_{tar} is smaller than that of W_{ini} and W_{tar} . We also see that, from a reconfiguration sequence for I' , we can easily construct a reconfiguration sequence for I by adding two extra moves.

Our algorithm applies one of (R1)–(R7), (G), and (D), and then apply the algorithm recursively, which is summarized in Algorithm 1. To show the validity of the algorithm, we prove the following lemma.

Lemma 10. *Suppose that $I = (G, W_{\text{ini}}, W_{\text{tar}})$ is an instance of FVS-RECONFIGURATION such that G is a cubic $K_{3,3}$ -minor-free graph and $W_{\text{ini}} \cap W_{\text{tar}} = \emptyset$. If I has no greedy-move, then it has a detour-greedy-move.*

Proof. Suppose that G is a cubic $K_{3,3}$ -minor-free graph and $W_{\text{ini}} \cap W_{\text{tar}} = \emptyset$. Suppose also that there is no greedy-move. By definition, we have $\pi_{\text{ini}}(W_{\text{ini}}) \cap W_{\text{tar}} = \emptyset$ and $\pi_{\text{tar}}(W_{\text{tar}}) \cap W_{\text{ini}} = \emptyset$. Now assume for a contradiction that there is no detour-greedy-move. Then we have $\pi_{\text{ini}}(W_{\text{ini}}) \cap \pi_{\text{tar}}(W_{\text{tar}}) = \emptyset$.

We now claim that $|\pi_{\text{ini}}(v)| = 1$ for each $v \in W_{\text{ini}}$. Assume this is false, which implies that $\pi_{\text{ini}}(v) = V \setminus W_{\text{ini}}$ or $\pi_{\text{ini}}(v) = V(C_v) - v$ for some cycle C_v by Lemma 8. If $\pi_{\text{ini}}(v) = V \setminus W_{\text{ini}}$, then $\pi_{\text{ini}}(v) \cap W_{\text{tar}} \neq \emptyset$, a contradiction. If $\pi_{\text{ini}}(v) = V(C_v) - v$ for some cycle C_v , then since $W_{\text{tar}} \cap V(C_v) \neq \emptyset$ as W_{tar} is a FVS, we obtain $\pi_{\text{ini}}(v) \cap W_{\text{tar}} \neq \emptyset$, a contradiction.

By Lemma 9, for each pair of distinct vertices $v, w \in W_{\text{ini}}$, we have $\pi_{\text{ini}}(v) \cap \pi_{\text{ini}}(w) = \emptyset$. Hence, we have $|\pi_{\text{ini}}(W_{\text{ini}})| = |W_{\text{ini}}|$. By the same argument, we obtain $|\pi_{\text{tar}}(W_{\text{tar}})| = |W_{\text{tar}}|$. In summary we have

1. $W_{\text{ini}} \cap W_{\text{tar}} = \emptyset$,
2. $\pi_{\text{ini}}(W_{\text{ini}}) \cap W_{\text{tar}} = \emptyset$,
3. $\pi_{\text{tar}}(W_{\text{tar}}) \cap W_{\text{ini}} = \emptyset$,
4. $\pi_{\text{ini}}(W_{\text{ini}}) \cap \pi_{\text{tar}}(W_{\text{tar}}) = \emptyset$,
5. $|\pi_{\text{ini}}(W_{\text{ini}})| = |W_{\text{ini}}|$, and
6. $|\pi_{\text{tar}}(W_{\text{tar}})| = |W_{\text{tar}}|$.

Recall that by definition we have $W_{\text{ini}} \cap \pi(W_{\text{ini}}) = \emptyset$ and $W_{\text{tar}} \cap \pi(W_{\text{tar}}) = \emptyset$. Combining this with the previous observations we obtain

$$|W_{\text{ini}} \cup \pi_{\text{ini}}(W_{\text{ini}}) \cup W_{\text{tar}} \cup \pi_{\text{tar}}(W_{\text{tar}})| = 2|W_{\text{ini}}| + 2|W_{\text{tar}}|.$$

According to Lemma 7 we have that $|V(G)| \leq 4k - 2$, where $k := |W_{\text{ini}}| = |W_{\text{tar}}|$. Then we have

$$\begin{aligned} 4k &= 2|W_{\text{ini}}| + 2|W_{\text{tar}}| \\ &= |W_{\text{ini}} \cup \pi_{\text{ini}}(W_{\text{ini}}) \cup W_{\text{tar}} \cup \pi_{\text{tar}}(W_{\text{tar}})| \\ &\leq |V(G)| \\ &\leq 4k - 2, \end{aligned}$$

a contradiction. Therefore, if there is no greedy-move then there is a detour-greedy-move. $\square$

Proof of Theorem 2. If $|V(G)| = 1$ and $|W_{\text{ini}}| = |W_{\text{tar}}|$, then $W_{\text{ini}} = W_{\text{tar}}$ holds, and hence $(G, W_{\text{ini}}, W_{\text{tar}})$ has a trivial reconfiguration sequence. By Proposition 6 and Lemma 10, in each iteration of Algorithm 1, we can apply one of (R1)–(R7), (G), and (D) unless $|V(G)| = 1$. Since each reduction decreases the value of $|V(G)| + |W_{\text{ini}} \setminus W_{\text{tar}}|$ by at least one (see Proposition 6), the algorithm terminates in polynomial steps. The validity of each reduction rule shows that Algorithm 1 correctly returns a reconfiguration sequence for $(G, W_{\text{ini}}, W_{\text{tar}})$, which completes the proof of Theorem 2. $\square$

3.4. Length of sequence

We now show that the proof of Theorems 2 yields a reconfiguration sequence of length at most $7k$, where the length is defined as the number of token movements. For an instance $I = (G, W_{\text{ini}}, W_{\text{tar}})$ of FVS-RECONFIGURATION, consider the value

$$f(I) = 3|W_{\text{ini}}| + 2|W_{\text{ini}} \setminus W_{\text{tar}}| + |V^2(G) \cap (W_{\text{ini}} \cup W_{\text{tar}})| + |V^1(G)|,$$

where $V^1(G)$ (resp. $V^2(G)$) is the set of vertices of degree one (resp. two) in G .

We show that the length of a reconfiguration sequence is bounded by $f(I)$. To this end, we observe that $f(I)$ is non-increasing in the procedure and if a reduction requires t extra moves, then $f(I)$ decreases by at least t as follows.

- In (R1), we need no extra move and $f(I)$ does not increase, because $|W_{\text{ini}}|$ decreases by one, $|W_{\text{ini}} \setminus W_{\text{tar}}|$ does not change, and $|V^2(G) \cap (W_{\text{ini}} \cup W_{\text{tar}})| + |V^1(G)|$ increases by at most three as G is subcubic.
- In (R2), (R4), and (R6), we need one extra move and $f(I)$ decreases by at least two, because $|W_{\text{ini}} \setminus W_{\text{tar}}|$ decreases by one and the other terms do not increase.
- In (R3), we need no extra move and $f(I)$ does not increase; if the degree of v is zero, then every term in $f(I)$ does not change; and if the degree of v is one, then $|V^1(G)|$ decreases by one, $|V^2(G) \cap (W_{\text{ini}} \cup W_{\text{tar}})|$ increases by at most one, and the other terms do not change.
- In (R5), we need one extra move and $f(I)$ decreases by at least one, because $|V^2(G) \cap (W_{\text{ini}} \cup W_{\text{tar}})|$ decreases by one and the other terms do not increase.
- In (R7), we need no extra move and $f(I)$ does not change.
- In (G) and (D), we move tokens at most twice and $f(I)$ decreases by two, because $|W_{\text{ini}} \setminus W_{\text{tar}}|$ decreases by one and the other terms do not change.

Since $f(I) = 0$ implies $W_{\text{ini}} = W_{\text{tar}} = \emptyset$, the above argument shows that the length of a reconfiguration sequence is bounded by $f(I)$.

In the first stage of our recursive algorithm, we exhaustively apply (R1), (R2), and (R3). Let $I^\circ = (G^\circ, W_{\text{ini}}^\circ, W_{\text{tar}}^\circ)$ be the instance after this stage. We observe that, if a token is moved in (R2), then it is removed in (R1) in the next iteration. This shows that we move tokens at most $|W_{\text{ini}}| - |W_{\text{ini}}^\circ|$ times in this stage. Since G° has no vertex of degree one, there is a reconfiguration sequence for I° of length at most $f(I^\circ) \leq 3|W_{\text{ini}}^\circ| + 2|W_{\text{ini}}^\circ \setminus W_{\text{tar}}^\circ| + |W_{\text{ini}}^\circ \cup W_{\text{tar}}^\circ| \leq 7|W_{\text{ini}}^\circ|$. Therefore, there is a reconfiguration sequence for I of length at most $(|W_{\text{ini}}| - |W_{\text{ini}}^\circ|) + 7|W_{\text{ini}}^\circ| \leq 7|W_{\text{ini}}| = 7k$.

3.5. Graphs with k disjoint cycles

We consider a special case of FVS-RECONFIGURATION with k tokens on subcubic graphs that contain k disjoint cycles. By using the same ideas as in the proof of Theorem 2, we show the following theorem.

Theorem 11. *Let G be a subcubic graph that contains k disjoint cycles. Then any two FVS of G of size k admit a reconfiguration sequence. Furthermore, such a sequence can be found in polynomial time.*

Proof. Consider FVSS of G of size k . Then, each of the k cycles contains a unique token. We exhaustively apply the reductions (R1)–(R7), and observe that the resulting graph has as many tokens as disjoint cycles. We may therefore assume that G is cubic and that $W_{\text{ini}} \cap W_{\text{tar}} = \emptyset$ by Proposition 6. Let $C_1, C_2, \dots, C_k$ be k disjoint cycles of G , where $k = |W_{\text{ini}}| = |W_{\text{tar}}|$. We first observe that each token can only be moved to a vertex on the same cycle. Indeed, if the token on cycle C_i is moved to a vertex outside of C_i then the cycle C_i remains after removing the resulting set of tokens, a contradiction to the definition of feedback vertex set.

Using this observation we can show that there always exists a greedy-move or a detour-greedy-move as in the proof of Theorem 2. Suppose to the contrary that there is neither a greedy-move nor a detour-greedy-move. By (R1), we have $W_{\text{ini}} \cap W_{\text{tar}} = \emptyset$. Furthermore, since there is no greedy-move we have $\pi_{\text{ini}}(W_{\text{ini}}) \cap W_{\text{tar}} = \emptyset$ and $\pi_{\text{tar}}(W_{\text{tar}}) \cap W_{\text{ini}} = \emptyset$. Since no token can be moved to another cycle among $C_1, C_2, \dots, C_k$, we have $|\pi_{\text{ini}}(W_{\text{ini}})| \geq |W_{\text{ini}}|$ and $|\pi_{\text{tar}}(W_{\text{tar}})| \geq |W_{\text{tar}}|$ by Lemma 8. Since there is no detour-greedy-move, we have $\pi_{\text{ini}}(W_{\text{ini}}) \cap \pi_{\text{tar}}(W_{\text{tar}}) = \emptyset$. Using the counting argument of the proof of Lemma 10, this leads to a contradiction to Lemma 7. This completes the proof of Theorem 11. $\square$

Note that the length of the reconfiguration sequence is at most $7k$ also for this case as in Section 3.4.

4. Reconfiguration of non-minimum FVS

The objective of this section is to show Theorem 4, that is, to show that the reconfiguration of FVS in subcubic graphs is always possible if we have one extra token. For this purpose, we first reduce FVS-RECONFIGURATION to the reconfiguration problem on the Matroid Parity (called MATROID PARITY RECONFIGURATION) in Section 4.1, and then show a result on MATROID PARITY RECONFIGURATION (Theorem 15) in Section 4.2, which implies Theorem 4.

4.1. Reduction to Matroid Parity

A *matroid* M is a pair $(S, \mathcal{I})$ where S is a finite set and $\mathcal{I} \subseteq 2^S$ is a family of *independent sets* satisfying the *independence axioms*: (I1) $\emptyset \in \mathcal{I}$, (I2) $I' \subseteq I \in \mathcal{I} \Rightarrow I' \in \mathcal{I}$, and (I3) $I, I' \in \mathcal{I}$, $|I| < |I'| \Rightarrow \exists v \in I' \setminus I$ s.t. $I + v \in \mathcal{I}$. Suppose that the ground set S is partitioned into pairs, called *lines*. Each $v \in S$ has its *mate* $\bar{v}$ such that $\{v, \bar{v}\}$ is a line. Let L denote the set of lines. A set $T \subseteq L$ of lines is called an *independent parity set* if $\bigcup_{\ell \in T} \ell \in \mathcal{I}$, where we denote $I(T) := \bigcup_{\ell \in T} \ell$. The problem of finding an independent parity set of maximum cardinality is called the *Matroid Parity problem* [18]. This problem can be solved in polynomial time for linear matroids [6], while it requires an exponential number of independence oracle calls in the general case.

As for FVS, we consider the reconfiguration of independent parity sets, which we call MATROID PARITY RECONFIGURATION. We say that two independent parity sets T and T' are *adjacent* if and only if $|T \setminus T'| = |T' \setminus T| = 1$. A *reconfiguration sequence* between two independent parity sets T_{ini} and T_{tar} is a sequence $T_0, T_1, \dots, T_h$ of independent parity sets such that $T_{\text{ini}} = T_0$, $T_{\text{tar}} = T_h$, and T_{i-1} and T_i are adjacent for $i = 1, \dots, h$.

It is known that the problem of finding a minimum FVS in subcubic graphs can be reduced to the Matroid Parity problem with a graphic matroid [6, 7]. By using the same construction, we reduce FVS-RECONFIGURATION in subcubic graphs to the reconfiguration of independent parity sets as follows.

Suppose that $G = (V, E)$ is a subcubic graph. We first subdivide each edge $e = uv \in E$, that is, remove the edge e and add a new vertex z_e and two new edges $z_e u$ and $z_e v$. Let H be the obtained graph and let $Z := \{z_e \mid e \in E\}$. We say that a vertex set $T \subseteq V$ *induces a 3-forest* in H if $H[T \cup Z]$

is a forest.¹ By the definition, we observe the following.

Observation 12. *A set $T \subseteq V$ induces a 3-forest in H if and only if $V \setminus T$ is a feedback vertex set in G .*

Lovász [6] shows that 3-forests in H can be represented by independent parity sets, which can be used to obtain a polynomial-time algorithm for the minimum feedback vertex set problem in (sub)cubic graphs (see [7] for another reduction).

Proposition 13 (Lovász [6]). *There exists a graphic matroid $(S, \mathcal{I})$ with the line set V such that a subset $T \subseteq V$ is an independent parity set if and only if it induces a 3-forest in H .*

By Observation 12 and Proposition 13, there exists a reconfiguration sequence between two independent parity sets $T_{\text{ini}}, T_{\text{tar}} \subseteq V$ of $(S, \mathcal{I})$ if and only if there exists a reconfiguration sequence between two feedback vertex sets $V \setminus T_{\text{ini}}$ and $V \setminus T_{\text{tar}}$ in G .

4.2. Reconfiguration of independent parity sets

We now consider the reconfiguration of independent parity sets. We first show the following lemma.

Lemma 14. *Let $(S, \mathcal{I})$ be a matroid with a line set L , and let T^* be an independent parity set. For any independent parity set $X \subseteq L$ with $|X| \leq |T^*| - 1$, there exists a sequence $T_0, T_1, \dots, T_h$ of independent parity sets such that $T_0 = X$, $T_h \subsetneq T^*$, and T_{i-1} and T_i are adjacent for $i = 1, \dots, h$.*

Proof. We prove Lemma 14 by induction on $|X \setminus T^*|$. If $|X \setminus T^*| = 0$, then $X \subsetneq T^*$, which shows that the sequence with $h = 0$ satisfies the conditions.

Suppose that $|X \setminus T^*| \geq 1$. Since $|I(T^*)| > |I(X)|$, (I3) shows that there exists an element $v \in I(T^*) \setminus I(X)$ such that $I(X) + v \in \mathcal{I}$. Let $\ell = \{v, \bar{v}\}$ be the line containing v , where $\bar{v}$ is the mate of v . We define a new independent parity set $X' \subseteq L$ by considering the following two cases.

1. Suppose that $I(X + \ell) \in \mathcal{I}$. In this case, define $X' = X + \ell - \ell'$, where ℓ' is an arbitrary line in $X \setminus T^*$. Note that such ℓ' exists, because $|X \setminus T^*| \geq 1$. By (I2), we obtain $I(X') \in \mathcal{I}$.

¹A 3-forest is defined for a 3-uniform hypergraph in [6], but we adopt an equivalent definition using H .

2. Suppose that $I(X + \ell) \notin \mathcal{I}$. Let $J_0 := I((X + \ell) \cap T^*)$, which is in $\mathcal{I}$ by (I2). Note that $v, \bar{v} \in J_0$. Since $I(X) + v \in \mathcal{I}$ and $J_0 \in \mathcal{I}$, by applying (I3) repeatedly, we obtain $J \subseteq I(X + \ell)$ such that $J \in \mathcal{I}$, $J_0 \subseteq J$, and $|J| = |I(X) + v|$. This shows that $J = I(X + \ell) - w$ for some element $w \in I(X + \ell) \setminus J_0$. Let ℓ' be the line containing w and define $X' = X + \ell - \ell'$. Then, we obtain $I(X') \subseteq J$, and hence $I(X') \in \mathcal{I}$ by (I2). Note that $\ell' \in X \setminus T^*$, because $w \in I(X + \ell) \setminus J_0 = I(X \setminus T^*)$.

In both cases, X' is an independent parity set such that $|X'| = |X| \leq |T^*| - 1$ and $|X' \setminus T^*| = |X \setminus T^*| - 1$. By the induction hypothesis, there exists a sequence from X' to T^* satisfying the conditions in the statement. Since X and X' are adjacent, this shows the existence of a desired sequence from X to T^* . $\square$

We are now ready to prove the following theorem.

Theorem 15. *Let $(S, \mathcal{I})$ be a matroid with a line set L , and let $T_{\text{ini}}, T_{\text{tar}} \subseteq L$ be its independent parity sets with $|T_{\text{ini}}| = |T_{\text{tar}}|$. If there exists an independent parity set $T^* \subseteq L$ with $|T^*| \geq |T_{\text{ini}}| + 1$, then there exists a reconfiguration sequence between T_{ini} and T_{tar} .*

Proof. By Lemma 14, there exists a reconfiguration sequence between T_{ini} and U_{ini} , where U_{ini} is some independent parity set with $U_{\text{ini}} \subsetneq T^*$. By Lemma 14 again, there exists a reconfiguration sequence between T_{tar} and U_{tar} , where U_{tar} is some independent parity set with $U_{\text{tar}} \subsetneq T^*$. Since $|U_{\text{ini}}| = |U_{\text{tar}}|$, there exists a sequence $U_0, U_1, \dots, U_h$ such that $U_0 = U_{\text{ini}}$, $U_h = U_{\text{tar}}$, $U_i \subseteq U_{\text{ini}} \cup U_{\text{tar}}$ for $i = 0, 1, \dots, h$, and U_{i-1} and U_i are adjacent for $i = 1, \dots, h$. Since $U_i \subseteq U_{\text{ini}} \cup U_{\text{tar}} \subseteq T^*$ and T^* is an independent parity set, we obtain $I(U_i) \in \mathcal{I}$ by (I2). Therefore, $U_0, U_1, \dots, U_h$ is a reconfiguration sequence between U_{ini} and U_{tar} . By concatenating these three reconfiguration sequences (i.e., the one between T_{ini} and U_{ini} , the one between U_{ini} and U_{tar} , and the one between U_{tar} and T_{tar}), we obtain a reconfiguration sequence between T_{ini} and T_{tar} , which completes the proof. $\square$

By this theorem and the reduction in Section 4.1, we obtain Theorem 4.

As described in Section 1, the above result gives a positive answer to a special case of Question 1. We remark here that the following interesting special cases of Question 1 are still open.

Question 2. *Does MATROID PARITY RECONFIGURATION for linear matroids admit a polynomial-time algorithm?*

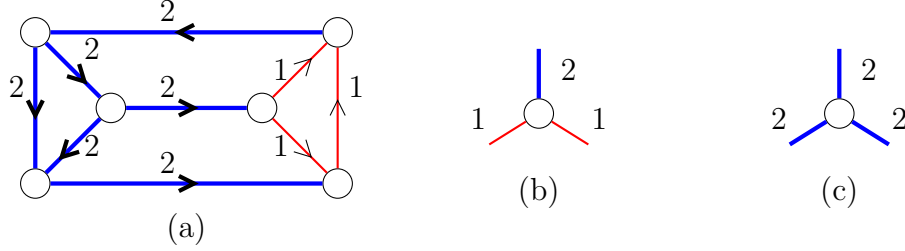


Figure 1: (a) A configuration of an NCL machine, (b) AND node, and (c) OR node.

Question 3. *Does FVS-RECONFIGURATION in subcubic graphs admit a polynomial-time algorithm?*

Question 4. *Is there a polynomial-time algorithm that, given two common independent sets of two matroids, decides whether there is a reconfiguration sequence between the two?*

5. Hardness proof for planar graphs of maximum degree four

In this section, we prove Theorem 3. It is obvious that FVS-RECONFIGURATION is in PSPACE. We show the PSPACE-hardness by giving a polynomial-time reduction from NONDETERMINISTIC CONSTRAINT LOGIC (NCL for short) [14, 19]. We first define formally NCL in Section 5.1. Then we give the construction of our reduction in Section 5.2. Finally, we show its correctness in Section 5.3.

5.1. Definition of Nondeterministic Constraint Logic

An *NCL machine* M is defined as an undirected graph together with a weight assignment from $\{1, 2\}$ on the edges of the graph. A *configuration* of M is an orientation (direction) of the edges such that, for every vertex v , the sum of weights of incoming arcs on v is at least 2. We will sometimes call a configuration of M a *valid* orientation of M to emphasize the constraint of incoming weights. Fig. 1(a) illustrates a configuration of an NCL machine, where each weight-2 edge is depicted by a thick (blue) line and each weight-1 edge by a thin (red) line. Two configurations of an NCL machine are said to be adjacent if they differ on exactly one edge, i.e., one is obtained from the other by flipping exactly one edge. Given an NCL machine M and two configurations C_{ini} and C_{tar} , the NCL problem asks for the existence of a sequence of adjacent configurations of M transforming C_{ini} into C_{tar} where

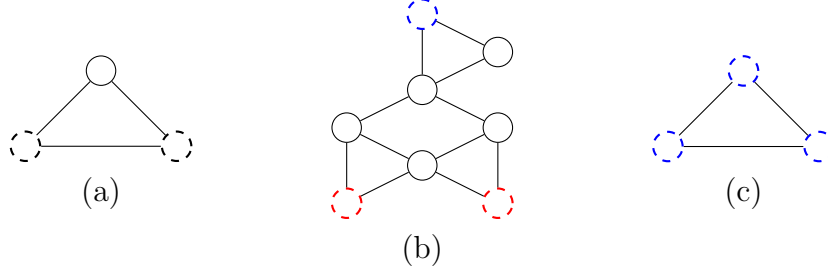


Figure 2: (a) edge-gadget, (b) AND-gadget, and (c) OR-gadget. The dashed vertices correspond to connectors. In the AND-gadget, the top connector will be identified with that of the weight-2 edge.

every intermediate configuration is valid. The NCL problem is known to be PSPACE-complete, see e.g. [14, 19].

It is known that NCL remains PSPACE-complete even for planar graphs of bounded bandwidth containing only two types of vertices, called *AND nodes* and *OR nodes* [19]. A vertex of degree three is called an AND node if its three incident edges have weights 1, 1, and 2 (see Fig. 1(b)). A vertex of degree three is called an OR node if every incident edge has weight 2 (see Fig. 1(c)). The NCL machine depicted in Fig. 1(a) only contains AND/OR nodes. From now on, we assume that such a graph is given as an input. Edges of NCL machines are called *NCL-edges*.

5.2. Construction of our reduction

Let I be an instance of NCL consisting of an NCL machine M and two configurations C_{ini} and C_{tar} of M .

Let us construct a graph G from M as follows. We will replace each NCL-edge and each AND/OR node, by a gadget. Each edge-gadget (resp. AND/OR-gadget) will contain two (resp. three) special vertices called *connector*. These connectors are denoted by (e, u) and (e, v) for the edge $e = uv$. If a vertex u has incident edges e_1, e_2 , and e_3 , then the corresponding connectors are (e_1, u) , (e_2, u) , and (e_3, u) , respectively.

The edge-gadget corresponding to an NCL-edge is a triangle with two connectors (see Fig. 2(a) for an illustration). For every AND-node, we construct an AND-gadget shown in Fig. 2(b). Note that every AND-gadget contains a cycle with no connectors, which we call the *central cycle*. Finally, each OR-gadget is a triangle where all the vertices are connectors (see Fig. 2(c) for an illustration). The graph G is obtained from the disjoint collection of gadgets

by simply identifying the connector (e, u) of the edge-gadget of e with the connector (e, u) of the vertex-gadget of u .

Notice that gadgets have maximum degree at most four and connectors have degree two, and hence each vertex of G has degree at most four since we identify at most two vertices of different gadgets together. Moreover, since M and the gadgets are planar, G is also planar. Since M has bounded bandwidth and the gadgets have bounded size, G also has bounded bandwidth.

Let k be the number of NCL-edges plus the number of AND-vertices. We consider FVSSs of size k of G . Since the graph G contains at least k disjoint cycles by construction, we have the following.

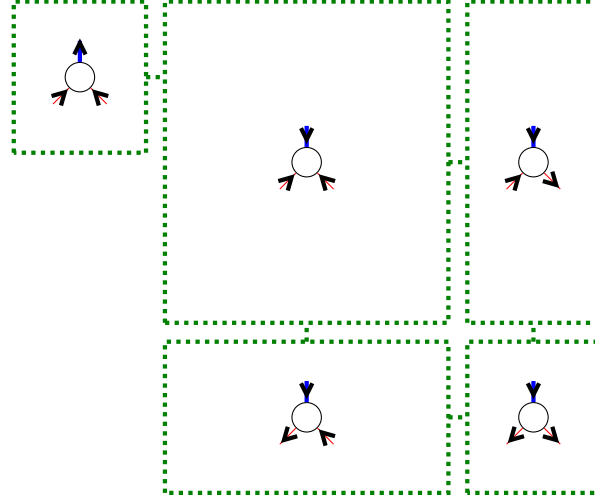
Observation 16. *Every FVS of size k of G satisfies the following: (i) there is exactly one token on the central cycle of every AND-gadget, and (ii) there is exactly one token on every edge-gadget.*

Let us now define our initial and target FVSSs W_{ini} and W_{tar} of G that correspond to configurations C_{ini} and C_{tar} of the NCL machine M . In W_{ini} (resp. W_{tar}), each edge-gadget has one token: if the direction of an NCL-edge $e = uv$ is inward for v , then we put a token to the connector (e, v) , otherwise, we put a token on the connector (e, u) . For each AND-gadget, we add one additional token on the central cycle so that every cycle in the gadget contains a token, which is possible as shown in Fig. 3 (b). If we have multiple choices of the additional tokens, then we choose an arbitrary one. Since C_{ini} (resp. C_{tar}) is valid, the cycle in each OR-gadget contains a token, and hence W_{ini} (resp. W_{tar}) is a FVS of size k of G . This completes the construction of our corresponding instance. The construction can be performed in polynomial time.

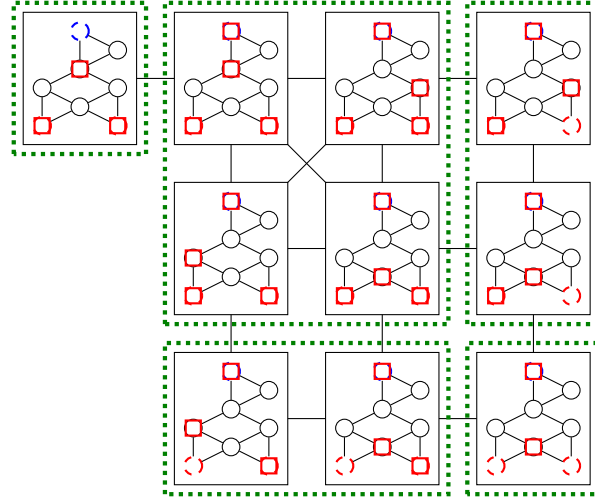
5.3. Correctness of our reduction

Let $(G, W_{\text{ini}}, W_{\text{tar}})$ be an instance of FVS-RECONFIGURATION that corresponds to an instance $(M, C_{\text{ini}}, C_{\text{tar}})$ of NCL. Let us prove that there exists a sequence of NCL configurations of M between C_{ini} and C_{tar} if and only if there exists a reconfiguration sequence between W_{ini} and W_{tar} on G .

We first prove the only-if direction. Suppose that there exists a sequence of NCL configurations of M between C_{ini} and C_{tar} . We show that any flip of an NCL-edge can be simulated by a reconfiguration sequence of FVSSs of G . Let C_i and C_{i+1} be two consecutive NCL configurations in the sequence. Let W_i and W_{i+1} be FVSSs of G corresponding to C_i and C_{i+1} , respectively,



(a)



(b)

Figure 3: (a) Valid orientations of the three incident edges of AND node. Each box represents a valid orientation of the three edges, and two boxes are joined by an edge if their orientations are adjacent, that is, one configuration is obtained from the other one by changing the direction of a single edge.

(b) All the FVSS of the AND-gadget which have exactly one token on the central cycle and tokens on connectors.

which can be constructed in the same way as W_{ini} and W_{tar} . By the construction, W_i and W_{i+1} have the same token on every edge-gadget except for the one corresponding to the flipped NCL-edge. An issue is that W_i and W_{i+1} may have different tokens on the central cycle of each AND-gadget, and so we deal with each AND-gadget carefully. Fig. 3(a) represents all the valid orientations of the three incident edges for an AND-node as well as their adjacency in the NCL reconfiguration graph. Fig. 3(b) shows all the FVSs of an AND-gadget that have exactly one token on the central cycle. In the figure, FVSs in the same large dotted box contain the same set of connectors and then correspond to the same orientation. Since every dotted box induces a connected reconfiguration graph, two FVSs corresponding to the same valid orientation of the NCL machine can be reconfigured into each other (without modifying tokens on the connectors). Furthermore, the AND-gadget preserves the “external adjacency” in the following sense: if we contract the FVSs in the same dotted box into a single vertex, then the resulting graph is exactly the reconfiguration graph depicted in Fig 3(a). Therefore W_i can be reconfigured to W_{i+1} . Hence there exists a reconfiguration sequence between W_{ini} and W_{tar} .

We now prove the if direction. Suppose that there exists a reconfiguration sequence $W_0, W_1, \dots, W_h$ from W_{ini} to W_{tar} . We may assume that every FVS in the sequence contains no vertex of degree two, since otherwise we can replace a vertex of degree two with its neighboring vertex. Then, by Observation 16, each edge-gadget contains exactly one token on a connector. We define an orientation C_i of M that corresponds to W_i as follows: the NCL-edge $e = uv$ is directed inward for v if and only if W_i contains the connector (e, v) . Since W_i is a FVS and each OR-gadget forms a cycle, each OR-node has at least one incoming arc in C_i , that is, the total weight of inward arcs is at least 2. One can also see that, for each AND-node, the total weight of inward arcs of C_i is at least 2 by considering all the cases (see Fig. 3(b)). Therefore, each C_i is valid. Note that W_{ini} and W_{tar} correspond to C_{ini} and C_{tar} , respectively, by construction. For $i = 1, 2, \dots, h$, since W_{i-1} and W_i are adjacent, either C_{i-1} and C_i are adjacent or $C_{i-1} = C_i$; see Fig. 3. By deleting identical consecutive orientations if necessary, we can obtain a sequence of valid adjacent orientations between C_{ini} and C_{tar} .

This completes the proof of Theorem 3 for $\Delta = 4$.

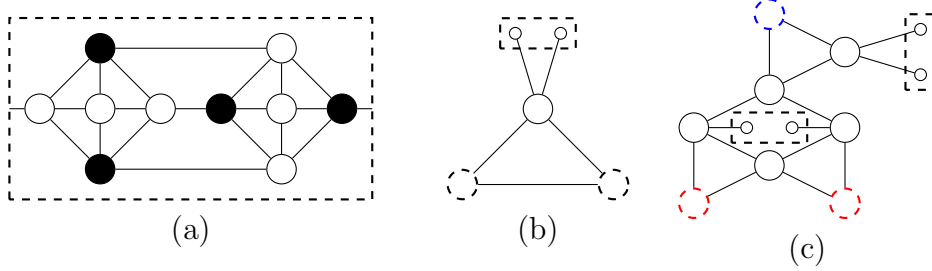


Figure 4: (a) The gadget H for 4-regular graphs, where the filled vertices are added to FVSS, (b)(c) edge-gadget and AND-gadget for 4-regular graphs.

5.4. Modifying degrees

We show that the problem remains hard even if the maximum degree is larger. Let G be the graph constructed in Section 5.2. Let G' be the graph obtained from G by adding a new vertex v' as a leaf, that is, add v' together with an edge between v' and a vertex v of G . One can easily check that the degree of vertex v increases, and there is a reconfiguration sequence in G' if and only if there is one in G . By adding more leaves, we can increase the maximum degree as much as we want. Therefore, the problem remains PSPACE-complete for every $\Delta \geq 4$, which completes the proof of Theorem 3.

We can also see that the hardness result holds also for 4-regular graphs.

Corollary 17. *FVS-RECONFIGURATION is PSPACE-complete on planar 4-regular graphs of bounded bandwidth.*

Proof. Let G be a graph of maximum degree four constructed in Section 5.2. By attaching gadget H in Figure 4(a), we can obtain a 4-regular graph G' as follows: for each edge-gadget, we attach a copy of H as in Figure 4(b), and for each AND-gadget, we attach two copies of H , as in Figure 4(c). Note that G' is also a planar graph of bounded bandwidth. Then, for each attached gadget, we add four tokens to both W_{ini} and W_{tar} as shown in Figure 4(a). Since any FVS of G' contains at least four tokens in each attached gadget, one can easily check that there is a reconfiguration sequence in G' if and only if there is one in G . $\square$

6. No-instances on 3-connected graphs

To prove Theorem 5 we use the following construction of a graph G_t for an odd integer $t \geq 3$ on $3t + 9$ vertices, whose feedback vertex number is $k := t + 2$.

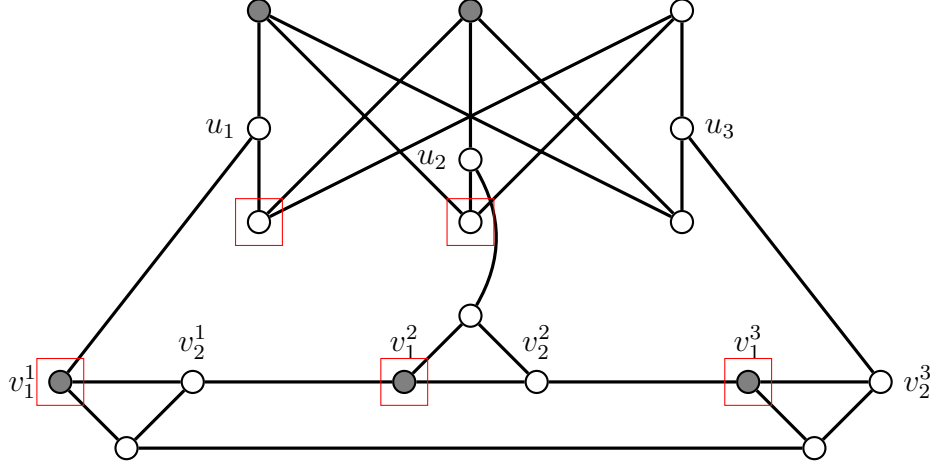


Figure 5: The graph G_3 . Two feedback vertex sets in different components of $\mathcal{C}_5(G_3)$ are indicated by the grey vertex set and the one in the red squares, respectively.

Proof of Theorem 5. Let H be the graph on nine vertices that is obtained from a $K_{3,3}$ by subdividing the edges of a perfect matching. Let u_1, u_2, u_3 be the respective midpoints added during the subdivision and let $\{w_1^1, w_2^1, w_3^1\}$ and $\{w_1^2, w_2^2, w_3^2\}$ be the bipartition of the vertices before the subdivision. For an odd integer $t \geq 3$, let G_t be the graph composed of H and t triangles on the vertex sets $T_i = \{v_1^i, v_2^i, v_3^i\}$ for $1 \leq i \leq t$ and add edges $v_2^i v_1^{i+1}$ for $1 \leq i \leq t-1$ as well as $u_1 v_1^1$, $u_3 v_3^t$, $u_2 v_3^{\lceil t/2 \rceil}$, and $v_3^i v_3^{t-i+1}$ for $1 \leq i \leq \lceil t/2 \rceil - 1$ (see Figure 5). It is readily verified that G_t is a 3-connected cubic graph on $3t + 9 = 3k + 3$ vertices.

We now consider the feedback vertex number of G_t for an odd integer $t \geq 3$. Since at least one vertex is needed on each triangle and two on H , the feedback vertex number of G_t is at least $t + 2$. Furthermore, since $W = \{v_1^i \mid 1 \leq i \leq t\} \cup \{w_1^1, w_2^1\}$ is a FVS of G_t of size $t + 2$, the feedback vertex number of G_t is $t + 2 = k$.

It remains to show that $\mathcal{C}_{t+2}(G_t)$ is not connected for an odd integer $t \geq 3$. Consider $W_1 = \{v_1^i \mid 1 \leq i \leq t\} \cup \{w_1^1, w_2^1\}$ and $W_2 = \{v_1^i \mid 1 \leq i \leq t\} \cup \{w_1^2, w_2^2\}$, which are FVSs of G_t , and show that W_1 is not reconfigurable to W_2 . Assume to the contrary, there is a reconfiguration sequence from W_1 to W_2 . Since every FVS of size k contains exactly two vertices in H , the restriction of this sequence to H gives a reconfiguration sequence from $\{w_1^1, w_2^1\}$ to $\{w_1^2, w_2^2\}$. However, we see that $\mathcal{C}_2(H)$ has no path from $\{w_1^1, w_2^1\}$ to $\{w_1^2, w_2^2\}$, which is a contradiction. Therefore, $\mathcal{C}_{t+2}(G_t)$ is not connected.

Figure 5 shows the graph G_3 together with two feedback vertex sets that are in different components of $\mathcal{C}_5(G_3)$. $\square$

Acknowledgement

Nicolas Bousquet was supported by ANR project GrR (ANR-18-CE40-0032). Yusuke Kobayashi was partially supported by JSPS KAKENHI Grant Numbers JP20K11692, JP20H05795, JP22H05001, Japan. Akira Suzuki was partially supported by JSPS KAKENHI Grant Numbers JP18H04091, JP20K11666, JP20H05794, Japan.

References

- [1] W. W. Johnson, W. E. Story, Notes on the “15” puzzle, *American Journal of Mathematics* 2 (4) (1879) 397–404.
- [2] G. Kendall, A. J. Parkes, K. Spoerer, A survey of NP-complete puzzles, *ICGA Journal* (2008) 13–34.
- [3] J. van den Heuvel, The complexity of change, *Surveys in Combinatorics* 2013 409 (2013) 127–160.
- [4] N. Nishimura, Introduction to reconfiguration, *Algorithms* 11 (4) (2018) 52. doi:10.3390/a11040052.
- [5] E. Speckenmeyer, On feedback vertex sets and nonseparating independent sets in cubic graphs, *Journal of Graph Theory* 12 (3) (1988) 405–412.
- [6] L. Lovász, Matroid matching and some applications, *Journal of Combinatorial Theory, Series B* 28 (2) (1980) 208–236. doi:https://doi.org/10.1016/0095-8956(80)90066-0.
- [7] S. Ueno, Y. Kajitani, S. Gotoh, On the nonseparating independent set problem and feedback set problem for graphs with no vertex degree exceeding three, *Discrete Mathematics* 72 (1-3) (1988) 355–360.
- [8] N. Bousquet, T. Hatanaka, T. Ito, M. Mühlenthaler, Shortest reconfiguration of matchings, in: *International Workshop on Graph-Theoretic Concepts in Computer Science*, Springer, 2019, pp. 162–174.

- [9] T. Ito, E. D. Demaine, N. J. Harvey, C. H. Papadimitriou, M. Sideri, R. Uehara, Y. Uno, On the complexity of reconfiguration problems, *Theoretical Computer Science* 412 (12) (2011) 1054–1065. doi:<https://doi.org/10.1016/j.tcs.2010.12.005>.
- [10] E. Speckenmeyer, Untersuchungen zum feedback vertex set problem in ungerichteten graphen, Ph.D. thesis, Universität Paderborn (1983).
- [11] J. A. Soto, A simple PTAS for weighted matroid matching on strongly base orderable matroids, *Discret. Appl. Math.* 164 (2014) 406–412. doi:[10.1016/j.dam.2012.10.019](https://doi.org/10.1016/j.dam.2012.10.019).
- [12] M. Kamiński, P. Medvedev, M. Milanič, Complexity of independent set reconfigurability problems, *Theoretical computer science* 439 (2012) 9–15.
- [13] M. Bonamy, N. Bousquet, Token sliding on chordal graphs, in: *Graph-Theoretic Concepts in Computer Science - 43rd International Workshop, WG 2017*, 2017, pp. 127–139. doi:[10.1007/978-3-319-68705-6_10](https://doi.org/10.1007/978-3-319-68705-6_10).
- [14] R. A. Hearn, E. D. Demaine, PSPACE-completeness of sliding-block puzzles and other problems through the nondeterministic constraint logic model of computation, *Theor. Comput. Sci.* 343 (1-2) (2005) 72–96. doi:[10.1016/j.tcs.2005.05.008](https://doi.org/10.1016/j.tcs.2005.05.008).
- [15] D. Lokshтанov, A. E. Mouawad, The complexity of independent set reconfiguration on bipartite graphs, *ACM Trans. Algorithms* 15 (1) (2019) 7:1–7:19. doi:[10.1145/3280825](https://doi.org/10.1145/3280825).
- [16] N. Bousquet, T. Ito, Y. Kobayashi, H. Mizuta, P. Ouyard, A. Suzuki, K. Wasa, Reconfiguration of spanning trees with many or few leaves, in: *28th Annual European Symposium on Algorithms, ESA 2020, September 7-9, 2020, Pisa, Italy (Virtual Conference)*, 2020, pp. 24:1–24:15. doi:[10.4230/LIPIcs.ESA.2020.24](https://doi.org/10.4230/LIPIcs.ESA.2020.24).
- [17] N. Bousquet, T. Ito, Y. Kobayashi, H. Mizuta, P. Ouyard, A. Suzuki, K. Wasa, Reconfiguration of spanning trees with degree constraint or diameter constraint, in: *39th International Symposium on Theoretical Aspects of Computer Science, STACS 2022, Marseille, France (Virtual Conference)*, 2022.

- [18] E. Lawler, Combinatorial Optimization - Networks and Matroids, Holt, Rinehart and Winston, New York, 1976.
- [19] T. C. van der Zanden, Parameterized complexity of graph constraint logic, in: Proceedings of the 10th International Symposium on Parameterized and Exact Computation (IPEC 2015), Vol. 43 of Leibniz International Proceedings in Informatics (LIPIcs), 2015, pp. 282–293.