

Reachability of Independent Sets and Vertex Covers Under Extended Reconfiguration Rules

Shuichi Hirahara   

National Institute of Informatics, Tokyo, Japan

Naoto Ohsaka   

CyberAgent, Inc., Tokyo, Japan

Tatsuhiko Suga  

Graduate School of Information Sciences, Tohoku University, Sendai, Japan

Akira Suzuki   

Center for Data-Driven Science and Artificial Intelligence, Tohoku University, Sendai, Japan

Yuma Tamura  

Graduate School of Information Sciences, Tohoku University, Sendai, Japan

Xiao Zhou 

Graduate School of Information Sciences, Tohoku University, Sendai, Japan

Abstract

In reconfiguration problems, we are given two feasible solutions to a graph problem and asked whether one can be transformed into the other via a sequence of feasible intermediate solutions under a given reconfiguration rule. While earlier work focused on modifying a single element at a time, recent studies have started examining how different rules impact computational complexity.

Motivated by recent progress, we study INDEPENDENT SET RECONFIGURATION (ISR) and VERTEX COVER RECONFIGURATION (VCR) under the k -Token Jumping (k -TJ) and k -Token Sliding (k -TS) models. In k -TJ, up to k vertices may be replaced, while k -TS additionally requires a perfect matching between removed and added vertices. It is known that the complexity of ISR crucially depends on k , ranging from PSPACE-complete and NP-complete to polynomial-time solvable.

In this paper, we further explore the gradient of computational complexity of the problems. We first show that ISR under k -TJ with $k = |I| - \mu$ remains NP-hard when μ is any fixed positive integer and the input graph is restricted to graphs of maximum degree 3 or planar graphs of maximum degree 4, where $|I|$ is the size of feasible solutions. In addition, we prove that the problem belongs to NP not only for $\mu = O(1)$ but also for $\mu = O(\log |I|)$. In contrast, we show that VCR under k -TJ is in XP when parameterized by $\mu = |S| - k$, where $|S|$ is the size of feasible solutions. Furthermore, we establish the PSPACE-completeness of ISR and VCR under both k -TJ and k -TS on several graph classes, for fixed k as well as superconstant k relative to the size of feasible solutions.

2012 ACM Subject Classification Theory of computation $\rightarrow$ Graph algorithms analysis; Theory of computation $\rightarrow$ Problems, reductions and completeness

Keywords and phrases combinatorial reconfiguration, extended reconfiguration rule, independent set reconfiguration, vertex cover reconfiguration, PSPACE-completeness, NP-completeness

Digital Object Identifier 10.4230/LIPIcs.ISAAC.2025.39

Related Version *Full Version:* <https://arxiv.org/abs/2510.24226>

Funding This work was partially supported by Institute of Mathematics for Industry, Joint Usage/Research Center in Kyushu University. (FY2024 Workshop (II) “Theory of Combinatorial Reconfiguration and Beyond” (2024a037))

Akira Suzuki: Partially supported by JSPS KAKENHI Grant Numbers JP25K14980.

Yuma Tamura: Partially supported by JSPS KAKENHI Grant Number JP25K21148.

Acknowledgements We are grateful to the anonymous referees for their helpful comments.



© Shuichi Hirahara, Naoto Ohsaka, Tatsuhiko Suga, Akira Suzuki, Yuma Tamura, and Xiao Zhou; licensed under Creative Commons License CC-BY 4.0

36th International Symposium on Algorithms and Computation (ISAAC 2025).

Editors: Ho-Lin Chen, Wing-Kai Hon, and Meng-Tsung Tsai; Article No. 39; pp. 39:1–39:20

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

Reconfiguration problems ask whether it is possible to reach a target state from an initial state by gradually transforming one feasible solution into another, where each intermediate solution must also be feasible. At each step, the current solution can be changed to an “adjacent” one, as defined by a given restriction known as a *reconfiguration rule*. One of the most well-known examples is the *15-puzzle*, where the rule permits sliding a tile into an adjacent empty space. Reconfiguration problems have been extensively studied in the context of classical graph problems involving feasible solutions such as independent sets [3, 4, 6, 7, 11, 18, 20, 23, 27], cliques [22], vertex covers [29], dominating sets [16, 36], and so on (see surveys [31, 38]). Three standard reconfiguration models have been widely studied in the literature: *token jumping* (TJ) [20, 23], *token sliding* (TS) [3, 6, 18, 20], and *token addition/removal* (TAR) [20, 23] models. In the TJ model, one can simultaneously remove any vertex from the current solution and add any vertex outside it. The TS model is a restricted version of TJ, where the removed vertex and the added vertex must be adjacent. In the TAR model, vertices can be added or removed as long as the resulting set remains above (or below) a specified size threshold.

Reconfiguration problems on graphs under those rules have attracted attention in theoretical computer science, and the computational complexity of such problems has been settled [20, 23, 30, 40]. Besides, the field of reconfiguration problems is in the course of trying to apply the theoretical viewpoint to functional implementation for practical use [9, 21, 37, 41].

However, one may find that, in practical scenarios, conventional rules such as changing one element at a time may be too restrictive. For example, reconfiguring a monitoring system or an infrastructure network often requires multiple simultaneous changes due to physical or operational constraints. Even if the system cannot be reconfigured under standard one-element rules, in practice, it is unjustified to conclude that it is “unreachable.” More flexible reconfiguration processes – such as those allowing multiple simultaneous changes – more accurately reflect the feasibility of real-world systems.

Motivated by recent progress and aiming to address the emerging issues, several studies have begun to analyze the computational complexity of reconfiguration problems under extended reconfiguration rules [10, 12, 17, 26, 35].

We study the reconfiguration problem under the extended rules, *k*-Token Jumping (*k*-TJ) and *k*-Token Sliding (*k*-TS) [10, 35], which allow the simultaneous exchange of up to *k* vertices.

1.1 Our Problems

In this paper, all graphs are simple and undirected. For two sets *A* and *B*, the set difference $A \setminus B$ is $\{x \in A : x \notin B\}$, and $A \triangle B$ denotes the symmetric difference between *A* and *B*, that is, $(A \setminus B) \cup (B \setminus A)$. For two vertex subsets *A* and *B* of a graph *G* with $|A| = |B|$, we say that they are *adjacent* under *k*-Token Jumping (*k*-TJ) if $|A \triangle B| \leq 2k$. On the other hand, they are said to be *adjacent* under *k*-Token Sliding (*k*-TS) if $|A \triangle B| \leq 2k$ and there exists a perfect matching between $A \setminus B$ and $B \setminus A$ in the graph *G*. Intuitively, the transformation from *A* to *B* can be seen as moving tokens placed on the vertices in the symmetric difference $A \triangle B$. Under *k*-TJ, up to *k* tokens can be moved simultaneously to any vertices in *G*. In contrast, under *k*-TS, up to *k* tokens can be moved simultaneously along the edges of *G*. Note that 1-TJ and 1-TS coincide with TJ and TS, respectively.

INDEPENDENT SET and VERTEX COVER are NP-complete problems [15] that are thoroughly explored in graph theory. We define their reconfiguration variants, that is, INDEPENDENT SET RECONFIGURATION (ISR) and VERTEX COVER RECONFIGURATION (VCR).

Recall that a set of vertices in a graph G is called an *independent set* if no two vertices in the set are adjacent in G , and a *vertex cover* if every edge in G has at least one endpoint in the set.

In the INDEPENDENT SET RECONFIGURATION problem, we are given a graph G , two independent sets I and J of G (representing the “initial” and “target” configurations of tokens, respectively) such that $|I| = |J|$, and a reconfiguration rule $R \in \{k\text{-TJ}, k\text{-TS}\}$. Then, the problem asks whether there exists a sequence $\sigma = \langle I = I_0, I_1, \dots, I_\ell = J \rangle$ of independent sets of G , where any two consecutive independent sets in σ are adjacent under R . Similarly, in VERTEX COVER RECONFIGURATION, we are given a graph G , two vertex covers S and T of G (representing the “initial” and “target” configurations of tokens, respectively) such that $|S| = |T|$, and a reconfiguration rule $R \in \{k\text{-TJ}, k\text{-TS}\}$. Then, the problem asks whether there exists a sequence $\sigma = \langle S = S_0, S_1, \dots, S_\ell = T \rangle$ of vertex covers of G , where any two consecutive vertex covers in σ are adjacent under R . In each problem, we refer to such a sequence of independent sets or vertex covers as a *reconfiguration sequence*, where ℓ is the length of the reconfiguration sequence. Furthermore, we say that two vertex sets are *reconfigurable* under R if there exists a reconfiguration sequence between them under R . Formally, the two problems are defined as follows.

Problem INDEPENDENT SET RECONFIGURATION (ISR)

Input A simple undirected graph G , two independent sets I and J of G with the same size, and a reconfiguration rule $R \in \{k\text{-TJ}, k\text{-TS}\}$.

Output Are I and J reconfigurable under R ?

Problem VERTEX COVER RECONFIGURATION (VCR)

Input A simple undirected graph G , two vertex covers S and T of G with the same size, and a reconfiguration rule $R \in \{k\text{-TJ}, k\text{-TS}\}$.

Output Are S and T reconfigurable under R ?

Related work. ISR under TJ and TS is PSPACE-complete even for planar graphs of maximum degree 3 and bounded bandwidth [18, 39, 40], and perfect graphs [23]. Under TJ, it is known that any two independent sets of an even-hole-free graph are reconfigurable [23]. Under TS, the problem remains PSPACE-complete for split graphs [3], while it can be solved in polynomial time for interval graphs [4]. For claw-free graphs, ISR under both TJ and TS can be solved in polynomial time [6]. For bipartite graphs, interestingly, ISR is NP-complete under TJ, while PSPACE-complete under TS [27]. Note that ISR and VCR under TJ and TS are essentially equivalent due to their complementary relationship; therefore, their computational complexities coincide.

In [35], several results are presented for ISR under the $k\text{-TJ}$ and $k\text{-TS}$ rules. The paper shows that ISR under both $k\text{-TJ}$ and $k\text{-TS}$ is PSPACE-complete on perfect graphs for any fixed integer $k \geq 2$. Furthermore, $k\text{-TS}$ and TS are essentially equivalent on even-hole-free graphs [35]. As a result, the computational complexity of ISR under $k\text{-TS}$ on several graph classes contained within the class of even-hole-free graphs follows from the results under TS.

Křiřan et al. studied several reconfiguration problems, including ISR and VCR, under the (k, d) -Token Jumping model [26]. In this model, k tokens can move simultaneously, and each token can travel within a distance of d . The (k, d) -Token Jumping model may appear to generalize the reconfiguration rules $k\text{-TJ}$ and $k\text{-TS}$; however, the settings are slightly different. The (k, d) -Token Jumping model is defined by a bijection between the current configuration and the next configuration. Consequently, a token can move to a vertex currently occupied by another token, as long as the latter token moves to a different vertex in the same step. In

contrast, the definitions of k -TJ and k -TS are based on the symmetric difference between configurations. In these models, a token is prohibited from moving to a vertex that is currently occupied by another token. Although both the (k, d) -Token Jumping model and k -TJ (resp. k -TS) are natural generalizations of TJ (resp. TS), the difference between them significantly impacts the computational complexity of problems. In fact, for the number t of tokens, VCR under the $(t, 1)$ -Token Jumping model can be solved in polynomial time [26], while VCR under the k -TS is PSPACE-complete when $k = t$ [35].

1.2 Our Contribution

We research the computational complexity of ISR and VCR under k -TJ and k -TS. An overview of our results is provided in Tables 1 and 2. The numbering of theorems follows the system starting from Section 3. Therefore, in this section, note that the numbering does not begin consecutively.

The results when parameterized by the guaranteed value

We first investigate the complexity of ISR under k -TJ when k is defined as $|I| - \mu$, where I is the initial independent set of an ISR instance and μ is a parameter referred to as the *guaranteed value* [28]. This parameter measures how far the instance is from the trivial case: when $\mu = 0$ (that is, $k = |I|$), ISR under k -TJ becomes trivial, as all vertices in the independent set can be replaced simultaneously. Hence, we are interested in the computational complexity of the problem when μ is small. It was shown in [35] that even for any fixed positive integer μ , ISR under k -TJ is NP-complete when $k = |I| - \mu$. However, the reduction presented in [35] introduces a large number of edges. From a practical standpoint, it is particularly important to understand the computational complexity of the problem on sparse graphs, such as planar graphs or graphs with bounded degree.

To answer this question, we present NP-hardness results for these sparse classes.

► **Theorem 2.** *Let μ be any fixed positive integer. ISR under k -TJ is NP-hard for graphs G of maximum degree 3 when $k = |I| - \mu \geq 1$, where I is an initial independent set of G .*

► **Theorem 3.** *Let μ be any fixed positive integer. ISR under k -TJ is NP-hard for planar graphs G of maximum degree 4 when $k = |I| - \mu \geq 1$, where I is an initial independent set of G .*

Here, let us explain the main obstacle in the proofs of these theorems. Consider the case where $\mu = 1$. If the initial and target independent sets I and J satisfy $|I \cap J| \geq 1$, then the reconfiguration is trivial. Hence, to ensure that the instance is non-trivial, it is essential to construct I and J so that $I \cap J = \emptyset$. A common approach used in existing research is to employ a complete bipartite graph; however, this prevents the resulting graph from being sparse. A more delicate reduction is required to preserve the sparsity of the graph. As a key step toward this goal, we introduce a new variant of EXACTLY 3-SAT, which we call INTERNAL EXACTLY 3-SAT, and show that it is NP-complete. In this variant, the input is a 3-CNF formula that is promised to be satisfiable under both the all-true and all-false assignments. The goal is to determine whether there exists a *mixed* satisfying assignment, that is, a satisfying assignment that is neither all-true nor all-false. We convert the all-true and all-false assignments to the initial and target independent sets, respectively. The existence of a mixed satisfying assignment corresponds to the reconfigurability from I to J via an independent set I' such that $|I' \cap I| \geq 1$ and $|I' \cap J| \geq 1$.

■ **Table 1** The complexity of ISR under k -TJ for various graph classes and values of k . Here, I denotes an initial independent set, and Δ , bw , and cw denote the maximum degree, bandwidth, and clique-width, respectively, of a given n -vertex graph. Let $\varepsilon_0 \in (0, 1)$ be a fixed constant.

	k -TJ					
	$k = 1$ (TJ)	any const. $k \geq 2$	any $k \leq \varepsilon_0 I $	$k = I - \mu$		$k = I $
				$\mu = O(\log I)$	any const. $\mu > 0$	
general	PSPACE-c.		PSPACE-c.	open (in NP?)	NP-c. [35]	trivially
bounded Δ	PSPACE-c.	PSPACE-c.	[Theorem 9]	NP-c.		always
$\Delta = 3$	[18, 39, 40]	[Theorem 7]		[Theorem 2]		yes
planar $\cap$ $\Delta = o(\frac{n}{\log n})$			NP-h.	[Theorem 3]		
planar $\cap \Delta = 4$				[Theorem 4]		
planar $\cap \Delta = 3$			open	open (NP-hard?)		
planar $\cap \Delta = 3$ $\cap$ bounded bw				XP		
bounded cw				parameterized by $\mu = I - k$		
perfect	PSPACE-c. [23]	PSPACE-c. [35]		[35]		
bipartite	NP-c. [27]	open				
claw-free	P [6, 23]	PSPACE-c. ($k = 2$)				
line		[Theorem 8]				
even-hole-free	always yes [23]					

We next demonstrate that ISR under k -TJ with $k = |I| - \mu$ belongs to NP even when $\mu = O(\log |I|)$ for some graph classes. Since many reconfiguration problems are PSPACE-complete due to the potentially super-polynomial length of reconfiguration sequences, showing that a reconfiguration problem belongs to NP is non-trivial. It is known that ISR under k -TJ with $k = |I| - \mu$ is in NP when μ is constant [35]. We strengthen this result by proving NP-membership for specific graph classes under the condition $\mu = O(\log |I|)$.

► **Theorem 4.** *Let G be an input graph with n vertices, chromatic number $O(1)$ and maximum degree $o(\frac{n}{\log n})$, and let I be an initial independent set of G . ISR under k -TJ is in NP when $k = |I| - \mu \geq 1$ with any non-negative integer μ at most $O(\log |I|)$.*

In the proof of Theorem 4, we utilize the concept of an *intersecting family* of a set [13], which has been extensively studied in the field of extremal set theory. Based on this concept, we derive an upper bound on the length of the reconfiguration sequence.

Here, the following Theorem 6 constitutes another main result of our work.

► **Theorem 6.** *VCR under k -TJ is in XP for general graphs G when parameterized by $\mu = |S| - k \geq 0$, where S is an initial vertex cover of G .*

Recall that in any graph G , a vertex cover and an independent set are complementary: the complement of a vertex cover of G is an independent set of G . Thus, ISR and VCR are generally considered equivalent problems. However, our result for VCR in Theorem 6 stands in sharp contrast to the known results for ISR (see also Table 2). In the proof of Theorem 6, we design an XP algorithm based on a *clique-compressed reconfiguration graph* [35].

The results when k is constant

We establish the PSPACE-completeness of ISR for fixed k on several graph classes.

► **Theorem 7.** *Let $k \geq 2$ be any fixed positive integer. ISR under $R \in \{k\text{-TS}, k\text{-TJ}\}$ is PSPACE-complete for planar graphs of maximum degree 3 and bounded bandwidth.*

■ **Table 2** Comparison between the complexity of ISR and VCR under k -TJ on general graphs and graphs of maximum degree 3. The vertex subset A of the input graph represents the initial solution for both ISR and VCR. Specifically, A is the input independent set for ISR, and the input vertex cover for VCR. Let $\varepsilon_0 \in (0, 1)$ be some constant.

problems	graph classes	any $k \leq \varepsilon_0 A $	$k = A - \mu$	
			$\mu = O(\log A)$	any fixed $\mu \geq 1$
ISR	general	PSPACE-c.	open (in NP?)	NP-c. [35]
	maximum degree 3	[Theorem 9]	NP-c. [Theorem 2, Theorem 4]	
VCR	general	PSPACE-c.	XP [Theorem 6]	
	maximum degree 3	[Theorem 12]	parameterized by $\mu = A - k$	

Together with known results for the case $k = 1$ [18, 39, 40], our findings provide a complete characterization of the PSPACE-completeness of ISR under k -TJ and k -TS for every fixed positive integer k and planar graphs of maximum degree 3 and bounded bandwidth.

We further demonstrate the following Theorem 8.

► **Theorem 8.** *ISR under $R \in \{2\text{-TJ}, 2\text{-TS}\}$ is PSPACE-complete for line graphs.*

This result stands in sharp contrast to the polynomial-time solvability of ISR under TJ and TS on claw-free graphs [6], which include line graphs as a special case.

The results when k is superconstant

We investigate the complexity of ISR under k -TJ when k is superconstant in the initial independent set size $|I|$. To this end, we construct a polynomial-time reduction from the “optimization variant” called MAXMINISR to ISR. By using the PSPACE-hardness of approximating MAXMINISR [19, 24, 32], we prove that ISR under k -TJ is PSPACE-complete on graphs of maximum degree 3 for a wide range of values of k , including $k = O(1)$, $\Theta(\log |I|)$, $\Theta(|I|^{O(1)})$, and $\Theta(|I|)$.

In particular, we show the following Theorem 9.

► **Theorem 9.** *There exists some constant $\varepsilon_0 \in (0, 1)$ such that ISR under k -TJ on graphs of maximum degree 3 is PSPACE-complete for any k satisfying the following condition: there exists a constant c such that $k \leq \varepsilon_0 |I|$ holds whenever $|I| \geq c$, where I is the initial independent set of the input graph.*

We show that a similar result holds for VCR.

► **Theorem 12.** *There exists some constant $\varepsilon_0 \in (0, 1)$ such that VCR under k -TJ on graphs of maximum degree 3 is PSPACE-complete for any k satisfying the following condition: there exists a constant c such that $k \leq \varepsilon_0 |S|$ holds whenever $|S| \geq c$, where S is the initial vertex cover of the input graph.*

Outline. The remainder of this paper is organized as follows. We begin with preliminaries in Section 2. In Section 3, we study the problems ISR and VCR under k -TJ, focusing on the guaranteed value μ . In Section 4, we demonstrate the PSPACE-completeness of these problems when k is fixed. Finally, in Section 5, we analyze the case where k is superconstant. Due to the space limitation, proofs marked * are omitted and can be found in the full version of this paper.

2 Preliminaries

For a positive integer i , we write $[i] = \{1, 2, \dots, i\}$. Let $G = (V, E)$ be a finite, simple, and undirected graph with the vertex set V and the edge set E . We use $V(G)$ and $E(G)$ to denote the vertex set and the edge set of G , respectively. For a vertex v of G , $N_G(v)$ and $N_G[v]$ denote the *open neighborhood* and the *closed neighborhood* of v , respectively, that is, $N_G(v) = \{u \in V(G) : uv \in E(G)\}$ and $N_G[v] = N_G(v) \cup \{v\}$. For a vertex set $X \subseteq V(G)$, we define $N_G(X) = \{v \in V(G) \setminus X : uv \in E(G), u \in X\}$ and $N_G[X] = N_G(X) \cup X$. A subgraph of G is a graph G' such that $V(G') \subseteq V(G)$ and $E(G') \subseteq E(G)$. For a subset $S \subseteq V(G)$, $G[S]$ denotes the subgraph induced by S . The *degree* of a vertex v in a graph G is the number of edges incident to v . The *maximum degree* of G is the largest degree among all vertices in G . A sequence $\langle v_0, v_1, \dots, v_\ell \rangle_G$ of vertices of a graph G , where $v_{i-1}v_i \in E$ for each $i \in [\ell]$, is called a *path* if all the vertices are distinct. It is called a *cycle* if $v_0, v_1, \dots, v_{\ell-1}$ are distinct and $v_0 = v_\ell$. The value ℓ is referred to as the *length* of the path or cycle. A graph G is said to be *connected* if there exists a path between u and v for any pair of vertices $u, v \in V(G)$. The *chromatic number* of G is the smallest positive integer c such that G has a c -coloring, where a c -coloring of a graph G is a mapping $f: V(G) \rightarrow [c]$ such that $f(u) \neq f(v)$ if $uv \in E(G)$. The *bandwidth* of a graph $G = (V, E)$ is the minimum integer b such that there exists a bijection $f: V \rightarrow \{1, \dots, |V|\}$ satisfying $|f(u) - f(v)| \leq b$ for every edge $uv \in E(G)$.

We conclude this section with the following simple remark. We can observe that the problems ISR and VCR can be solved in nondeterministic polynomial space. Thus, by applying Savitch's Theorem [34], these problems are in PSPACE. Consequently, to establish PSPACE-completeness of ISR and VCR, it suffices to prove PSPACE-hardness.

3 When Parameterized by a Guaranteed Value

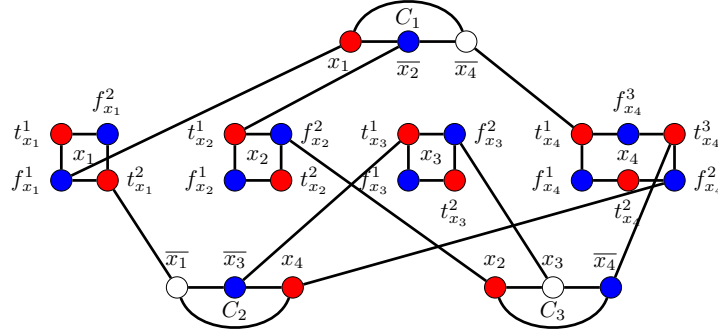
In this section, we discuss ISR and VCR under k -TJ, focusing on the guaranteed value.

3.1 ISR

We investigate the NP-completeness of ISR under k -TJ when $k = |I| - \mu$, where I is the initial independent set of an ISR instance, and μ is a parameter called the *guaranteed value*. Hereafter, we use I to denote the initial independent set of an ISR instance. For any fixed positive integer μ , it is known that ISR under k -TJ is NP-complete when $k = |I| - \mu$ [35]. In Section 3.1.1, we show that ISR under k -TJ remains NP-hard even when the input graph is restricted to graphs of maximum degree 3, or to planar graphs of maximum degree 4, where $k = |I| - \mu$ for any fixed positive integer μ . Then, in Section 3.1.2, we prove that the problem remains in NP even when $\mu = O(\log |I|)$ and an input graph is a graph of bounded chromatic number and maximum degree $o(\frac{n}{\log n})$, where n is the number of vertices in the input graph.

3.1.1 NP-hardness

We show the NP-hardness of ISR under k -TJ for graphs of maximum degree 3 and planar graphs of maximum degree 4 when $k = |I| - \mu$ with any fixed positive integer μ . To this end, we construct a chain of reductions. As a source problem of our reduction, we will use EXACTLY 3-SAT (E3-SAT for short), which is NP-complete [15]. Firstly, E3-SAT is reduced to INTERNAL EXACTLY 3-SAT (INTE3-SAT for short), a new variant of E3-SAT (to the best of our knowledge). Afterward, INTE3-SAT is reduced to our problem.



■ **Figure 1** Illustration of the construction of G from a Boolean formula ϕ' used in the proof of Theorem 2, where $\phi' = (x_1 \vee \bar{x}_2 \vee \bar{x}_4) \wedge (\bar{x}_1 \vee \bar{x}_3 \vee x_4) \wedge (x_2 \vee x_3 \vee \bar{x}_4)$. The blue marked tokens are on I , and the red marked tokens are on J .

Here, let us define E3-SAT and INTE3-SAT. Let $\phi = C_1 \wedge C_2 \wedge \dots \wedge C_m$ be a Boolean formula in conjunctive normal form (CNF), and $X = \{x_1, \dots, x_n\}$ be the variable set of ϕ . Each clause C_i for $i \in [m]$ in ϕ is a disjunction of literals and each literal appears as either a positive form x_j or a negative form $\bar{x}_j$ for a variable $x_j \in X$. A variable assignment is a mapping $b: X \rightarrow \{T, F\}$. We say that a variable assignment *satisfies* ϕ if ϕ evaluates to T . A CNF formula ϕ is called an Ek -CNF formula if any clause in ϕ consists of exactly k literals. Given an E3-CNF formula ϕ , E3-SAT asks whether there exists a variable assignment that satisfies ϕ .

A variable assignment b is called an *all- T assignment* if $b(x) = T$ for all $x \in X$, and an *all- F assignment* if $b(x) = F$ for all $x \in X$. Otherwise, it is called *mixed*. An Ek -CNF formula ϕ is called *sandwiched* if all- T and all- F assignments satisfy ϕ . In other words, any clause of a sandwiched Ek -CNF formula ϕ contains both positive and negative literals. Given a sandwiched Ek -CNF formula ϕ , INTE k -SAT asks whether there exists a mixed variable assignment that satisfies ϕ . We first show that INTE3-SAT is NP-complete by reducing E3-SAT.

► **Lemma 1.** INTE3-SAT is NP-complete.

For the proof, refer to Section 3.1.1.1. By the polynomial-time reduction from INTE3-SAT, we prove the following Theorem 2.

► **Theorem 2.** Let μ be any fixed positive integer. ISR under k -TJ is NP-hard for graphs G of maximum degree 3 when $k = |I| - \mu \geq 1$, where I is an initial independent set of G .

Proof. We use a polynomial-time reduction from INTE3-SAT. Let ϕ' be an instance of INTE3-SAT and X' be the set of variables of ϕ' . We will construct an instance $(G, I, J, k\text{-TJ})$ of ISR under k -TJ where $k = |I| - 1$ (see Figure 1 for an illustration), and then modify for any fixed $\mu \geq 1$.

For each variable $x \in X'$, let a_x denote the number of clauses of ϕ' in which x appears as a literal. For each variable $x \in X'$, we set up a *variable gadget* Y_x , which is defined as a cycle with $2a_x$ vertices (note that if x appears only once, then Y_x is a path with two vertices). The vertices in Y_x are labeled with $t_x^1, f_x^1, t_x^2, f_x^2, \dots, t_x^{a_x}, f_x^{a_x}$ in a counterclockwise order. Intuitively, t_x^i and f_x^i for each $i \in [a_x]$ correspond to T and F assignments for x , respectively. We refer to a vertex of Y_x corresponding to T (resp. F) assignment to x as a *true vertex* (resp. *false vertex*). For each clause C , we set up a *clause gadget* K_C , which is defined as

a complete graph with three vertices. The vertices in K_C correspond to the three literals in C . We refer to a vertex of K_C corresponding to a positive literal (resp. negative literal) of C as a *positive vertex* (resp. *negative vertex*). Finally, we connect variable gadgets and clause gadgets with edges as follows: For a vertex v in a clause gadget corresponding to a literal l of a variable x with occurrence $i \in [a_x]$, connect v to t_x^i if the literal is negative; otherwise, connect v to f_x^i . Let G be the obtained graph. Observe that G is a graph of maximum degree 3. Let I be a set of all false vertices in variable gadgets and a negative vertex arbitrarily chosen from each clause gadget. Let J be a set of all true vertices in variable gadgets and a positive vertex arbitrarily chosen from each clause gadget. Note that such I and J always exist as each clause has both positive and negative literals due to the definition of INTE3-SAT. Finally, we set $k = |I| - 1$.

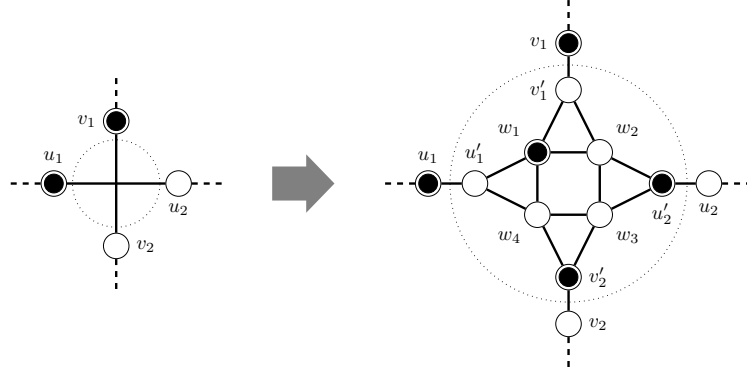
If $\mu \geq 2$, then we also add $\mu - 1$ isolated vertices to G . Let V^* be the set of all added vertices. Then let G^* be an obtained graph, $I^* = I \cup V^*$, and $J^* = J \cup V^*$. We also set $k = |I^*| - \mu$. Note that I is a maximum independent set of G and thus any independent set with size $|I^*| = |I| + |V^*|$ of G^* contains V^* . This allows us to move at most $k = |I^*| - \mu = |I| + |V^*| - \mu = |I| - 1$ tokens simultaneously only on G . Therefore, $(G, I, J, (|I| - 1)\text{-TJ})$ is a yes-instance if and only if $(G^*, I^*, J^*, (|I^*| - \mu)\text{-TJ})$ is a yes-instance.

For this reason, we discuss only the case when $\mu = 1$ here. The instance $(G, I, J, k\text{-TJ})$ of ISR under $k\text{-TJ}$ is clearly obtained in polynomial time. To complete our reduction, we will show that ϕ' is a yes-instance of INTE3-SAT if and only if $(G, I, J, k\text{-TJ})$ is a yes-instance of ISR under $k\text{-TJ}$ where $k = |I| - 1$.

Assume that there is a variable assignment b' that is mixed and satisfies ϕ' . We construct an independent set I' of G as follows. For a variable x_i with $i \in [n]$, if x_i is assigned T , then all true vertices of Y_{x_i} are contained in I' . Similarly, if x_i is assigned F , then all false vertices of Y_{x_i} are contained in I' . Since those vertices are chosen according to b' , one vertex in each clause gadget that corresponds to a literal evaluating T can also be contained in I' . Furthermore, since b' is mixed, we have $|I \cap I'| \geq 1$ and $|I' \cap J| \geq 1$. Therefore, a reconfiguration sequence $\sigma = \langle I, I', J \rangle$ exists.

Conversely, assume that there is a reconfiguration sequence $\sigma = \langle I = I_0, I_1, \dots, I_\ell = J \rangle$. Consider an integer $i \in [\ell]$ such that tokens on I_i are placed on true vertices for the first time. In other words, all tokens of I_{i-1} in variable gadgets are on false vertices. Since any positive vertex of any clause gadget is adjacent to a false vertex of a variable gadget, all positive vertices are not in I_{i-1} . Then, we claim that not all true vertices are in I_i . For the sake of contradiction, assume that all true vertices are in I_i . Then, since each negative vertex is adjacent to a true vertex, each token of I_i in clause gadgets is on a positive vertex. Thus, since I_{i-1} contains only false vertices and negative vertices, $I_{i-1} \cap I_i = \emptyset$. However, $|I_{i-1} \cap I_i| \geq 1$ must hold because they are adjacent under $k\text{-TJ}$, which is a contradiction. Therefore, we conclude that I_i contains both true and false vertices. From our construction of the variable gadgets, either true or false vertices are in I_i for each variable gadget. For each $j \in [n]$, let b' be a variable assignment such that x_j is assigned T if and only if true vertices of Y_{x_j} are in I_i . Then, literals of clauses in ϕ' corresponding to vertices in I_i evaluate T from our construction, that is, b' satisfies ϕ' . Therefore, ϕ' is a yes-instance of INTE3-SAT. ◀

By Theorem 2, we have proven the NP-hardness of ISR under $k\text{-TJ}$ on graphs of maximum degree 3 unless we can move all tokens. We now proceed to the NP-hardness of ISR under $k\text{-TJ}$ on planar graphs of maximum degree 4. The constructed graph G in the proof of Theorem 2 may have some edges crossing on a plane. We will eliminate these crossings by replacing them with a *crossover gadget* (as shown in Figure 2). A crossover gadget consists of eight vertices $u'_1, u'_2, v'_1, v'_2, w_1, \dots, w_4$ and twelve edges: $u'_1 w_1, u'_1 w_4, u'_2 w_2, u'_2 w_3, v'_1 w_1,$



■ **Figure 2** An illustration of replacing the crossing edges u_1u_2 and v_1v_2 with a crossover gadget. The gadget contains eight vertices and three tokens, and its maximum degree is 4.

$v'_1w_2, v'_2w_3, v'_2w_4, w_1w_2, w_2w_3, w_3w_4$, and w_4w_1 . For two crossing edges u_1u_2 and v_1v_2 of G , replacing this crossing with a crossover gadget means removing the edges u_1u_2 and v_1v_2 , inserting a crossover gadget, and adding the edges $u_1u'_1, u_2u'_2, v_1v'_1$, and $v_2v'_2$. The crossover gadget always contains exactly three tokens corresponding to a maximum independent set of the gadget.

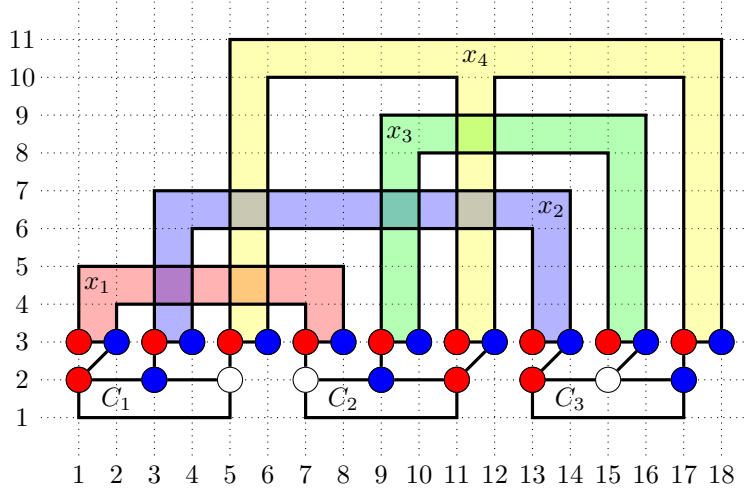
► **Theorem 3.** Let μ be any fixed positive integer. ISR under k -TJ is NP-hard for planar graphs G of maximum degree 4 when $k = |I| - \mu \geq 1$, where I is an initial independent set of G .

Proof. As the same reason in the proof of Theorem 2, we provide the proof when $\mu = 1$ (that is, $k = |I| - 1$). Let $(G, I, J, k\text{-TJ})$ be the instance of ISR constructed from ϕ' in the proof of Theorem 2, where $k = |I| - 1$. Consider any drawing of G on the plane, which may have crossings.

To make G into a planar graph of maximum degree 4, we replace each crossing in the drawing with a crossover gadget, which is shown in Figure 2. Suppose that we replaced a crossing of edges u_1u_2 and v_1v_2 with a crossover gadget. We claim that the crossover gadget correctly “simulates” the crossing if the crossing is *good*, that is, every independent set I' of G with size $|I|$ satisfies $|I' \cap \{u_1, u_2\}| = |I' \cap \{v_1, v_2\}| = 1$. Although not every drawing of G meets this condition¹, we claim that G always admits a drawing where all crossings are good. To this end, we construct a drawing of G in which all edge crossings occur only between edges belonging to distinct variable gadgets.

Let m be the number of clauses and n be the number of variables in ϕ' , respectively. We will represent all variable gadgets and clause gadgets on a grid with $(2n + 3)$ rows and $6m$ columns such that edges between variable gadgets and clause gadgets have no crossing (see also Figure 3). Let (i, j) be the coordinates of a point on the Euclidean plane, where $i \in [6m]$ and $j \in [2n + 3]$. For each clause gadget corresponding to a clause C_i of ϕ' , the three literal vertices of the gadget are positioned at $(6(i - 1) + 1, 2)$, $(6(i - 1) + 3, 2)$, and $(6(i - 1) + 5, 2)$. Furthermore, the edges are embedded to minimize the sum of their length, with one of the three edges utilizing the bottom border. If a literal vertex, corresponding to

¹ For example, the drawing in Figure 1 contains a non-good crossing of two edges $f_{x_3}^2v_{x_3}$ and $f_{x_4}^2u_{x_4}$, where the two vertices v_{x_3} and u_{x_4} correspond to the literal x_3 in clause C_3 and the literal x_4 in clause C_2 , respectively. In this crossing, $f_{x_3}^2$ and v_{x_3} are both not in the independent set J .



■ **Figure 3** Another drawing of G (shown in Figure 1), corresponding to the Boolean formula $\phi' = (x_1 \vee \bar{x}_2 \vee \bar{x}_4) \wedge (\bar{x}_1 \vee \bar{x}_3 \vee x_4) \wedge (x_2 \vee x_3 \vee \bar{x}_4)$, on a grid with 11 rows and 18 columns. The intersection of a dotted horizontal line labeled $i \in \{1, \dots, 11\}$ and a dotted vertical line labeled $j \in \{1, \dots, 18\}$ represents the coordinate (i, j) . Thick lines represent the edges of the graph G , and all edge crossings occur only between edges belonging to distinct variable gadgets.

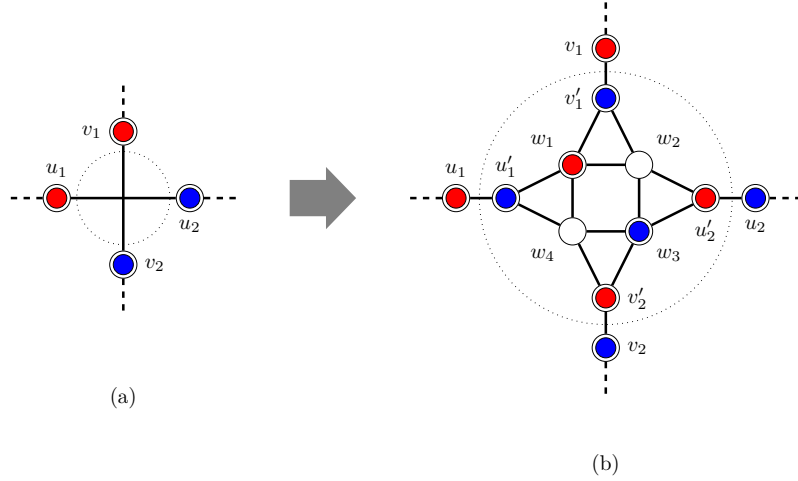
the j -th occurrence of a literal of variable x , is placed at $(i', 2)$, then the true vertex t_x^j and the false vertex f_x^j are positioned at $(i', 3)$ and $(i' + 1, 3)$, respectively, and are connected by a shortest edge.

Additionally, each literal vertex is joined with either of them, which follows the construction of G in the proof of Theorem 2: for a vertex v in a clause gadget corresponding to a literal l of a variable x with occurrence $j \in [a_x]$, connect v to t_x^j if l is negative, and to f_x^j otherwise. Then, for a variable gadget corresponding to a variable x_p with $p \in [n]$, we embed the remaining edges in the gadget using only the vertical lines where its vertices are located and the $(2p + 2)$ -th and $(2p + 3)$ -th horizontal lines, minimizing the total length. (If the variable gadget is a path with 2 vertices, there is nothing to do.)

Now, a new embedding of G is obtained and denoted by $D(G)$. Although $D(G)$ and G are isomorphic, $D(G)$ only has crossing edges in variable gadgets. Since every variable gadget forms a cycle of even length and has tokens placed alternately on its vertices, exactly one endpoint of each edge in the variable gadgets belongs to any independent set of G of size exactly $|I|$. Thus, all crossings of $D(G)$ are good.

We pick a crossing of two edges, say u_1u_2 and v_1v_2 . Since both u_1u_2 and v_1v_2 are edges of variable gadgets, we may assume without loss of generality that $u_1, v_1 \in I$ and $u_2, v_2 \in J$. We then replace this crossing with our crossover gadget. Let G^* denote the resulting graph. For the initial (resp. target) independent set I (resp. J) of G , adding three tokens on w_1, u'_2 , and v'_2 (resp. w_3, u'_1 , and v'_1) in the crossover gadget (as shown in Figure 4) results in the independent set I^* (resp. J^*). Let $(G^*, I^*, J^*, k\text{-TJ})$ be a new instance of ISR under $k\text{-TJ}$, where $k = |I^*| - 1$. Note that, since the sets of vertices added to I and J are disjoint, we have $I^* \cap J^* = \emptyset$.

For each configuration of tokens on u_1, u_2, v_1 , and v_2 in G , there exists a unique configuration of three tokens on the crossover gadgets in G^* as shown in Figure 5. Thus, the configuration of tokens on the vertices of the crossover gadget can be changed in G^* if and only if the configuration of tokens on u_1, u_2, v_1 , and v_2 is changed in G . Therefore,



■ **Figure 4** An illustration of adding tokens to I and J , yielding I^* and J^* . (a) Vertices u_1 and v_1 belong to I (red tokens), and u_2 and v_2 belong to J (blue tokens). (b) In I^* , the new vertices w_1, u'_1, u'_2 , and v'_2 are added, while in J^* , the new vertices w_3, u'_1 , and v'_1 are added. The two independent sets I^* and J^* remain disjoint.

by faithfully simulating token moves along the edges u_1u_2 and v_1v_2 of G using the token arrangements shown in Figure 5, one can show that $(G, I, J, k\text{-TJ})$ is a yes-instance if and only if $(G^*, I^*, J^*, k\text{-TJ})$ is a yes-instance.

The graph G^* has one less edge crossing than G . Consequently, we can obtain a plane graph G' by repeating the above replacement for each crossing of $D(G)$. As $D(G)$ has $O(mn)$ crossings, the construction of G' can be done in polynomial time. Since a crossover gadget is a graph of maximum degree 4, G' is a plane graph of maximum degree 4. This completes our proof. ◀

3.1.1.1 Proof of Lemma 1

Proof. It is obvious that INTE3-SAT is in NP. To prove the NP-completeness, we use a polynomial-time reduction from E3-SAT.

Let $\phi = C_1 \wedge C_2 \wedge \dots \wedge C_m$ be an instance of E3-SAT, and $X = \{x_1, \dots, x_n\}$ be the variable set of ϕ . We will construct a sandwiched E3-CNF formula ϕ' with $7mn^2$ clauses and $n + 2mn^2$ variables at the end of our construction. Before starting our construction, we restrict ϕ so that neither the all- T assignment nor the all- F assignment satisfies ϕ . E3-SAT remains NP-hard with this restriction; otherwise, we could solve E3-SAT without the restriction by checking whether the all- T or all- F assignment satisfies ϕ at first. We obtain a new CNF formula ϕ^* as follows:

$$\phi^* = \phi \vee \bigwedge_{i=1}^n x_i \vee \bigwedge_{i=1}^n \overline{x_i} = \bigwedge_{h=1}^m C_h \vee \bigwedge_{i=1}^n x_i \vee \bigwedge_{i=1}^n \overline{x_i} = \bigwedge_{h=1}^m \bigwedge_{i=1}^n \bigwedge_{j=1}^n (C_h \vee x_i \vee \overline{x_j}).$$

Since C_h consists of exactly three literals, $C_h \vee x_i \vee \overline{x_j}$ is a disjunction of five literals. Thus, ϕ^* is an E5-CNF formula. Furthermore, ϕ^* is a sandwiched E5-CNF formula, that is, each clause of ϕ^* has both a positive literal x_i and a negative literal $\overline{x_j}$. From our conversion, ϕ^* has mn^2 clauses and n variables. It is observed that for any mixed variable assignment b , ϕ evaluates T if and only if ϕ^* evaluates T . Thus, we can immediately say that ϕ is a yes-instance of E3-SAT without all- T and all- F assignments if and only if ϕ' is a yes-instance of INTE5-SAT.

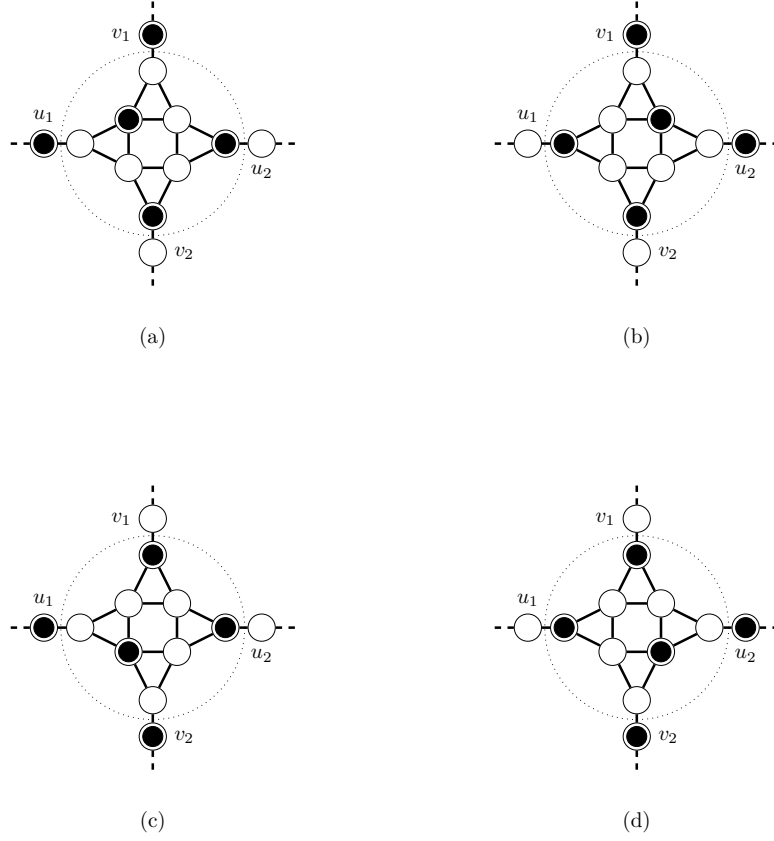


Figure 5 An illustration of how the crossover gadget works. The crossing edges u_1u_2 and v_1v_2 , where one endpoint of each edge is occupied by a token, are replaced by a crossover gadget: (a) u_1 and v_1 are occupied; (b) u_2 and v_1 are occupied; (c) u_1 and v_2 are occupied; (d) u_2 and v_2 are occupied.

Then, we explain how to convert a sandwiched E5-CNF formula ϕ^* to a sandwiched E3-CNF formula ϕ' with $7mn^2$ clauses and $n + 2mn^2$ variables. We repeat the following operation on ϕ^* until all clauses have size exactly 3. Let $\phi_0^* = \phi^*$ and ϕ_i^* for a positive integer i be a formula obtained from a sandwiched CNF formula ϕ_{i-1}^* .

Let C be a clause in ϕ_{i-1}^* with size at least 4. Then C contains $x \vee y$ or $\bar{x} \vee \bar{y}$ for variables x, y . If C contains $x \vee y$, then we replace $x \vee y$ with a new variable z and combine the modified formula with the formula $x \vee y \leftrightarrow z$ using the $\wedge$ operator. Furthermore, $x \vee y \leftrightarrow z$ is transformed as follows:

$$\begin{aligned}
 x \vee y \leftrightarrow z &= (x \vee y \vee \bar{z}) \wedge (\overline{(x \vee y)} \vee z) \\
 &= (x \vee y \vee \bar{z}) \wedge ((\bar{x} \wedge \bar{y}) \vee z) \\
 &= (x \vee y \vee \bar{z}) \wedge (\bar{x} \vee z) \wedge (\bar{y} \vee z) \\
 &= (x \vee y \vee \bar{z}) \wedge (\bar{x} \vee \bar{x} \vee z) \wedge (\bar{y} \vee \bar{y} \vee z).
 \end{aligned} \tag{1}$$

We claim that ϕ_i^* obtained from ϕ_{i-1}^* by the above operation is a sandwiched CNF formula. Clearly, each clause in ϕ_i^* that remains unchanged from ϕ_{i-1}^* contains both positive and negative literals. The clause in ϕ_i^* obtained from C by replacing $x \vee y$ with a positive literal z contains a negative literal because C also has a negative literal. Combined with Equation (1), ϕ_i^* is a sandwiched CNF formula. Consequently, if there is a clause in ϕ_{i-1}^* that contains $x \vee y$, then a sandwiched CNF formula ϕ_i^* is obtained.

Similarly, if C contains $\bar{x} \vee \bar{y}$, then we replace $\bar{x} \vee \bar{y}$ with the negative literal of a new variable z and combine the modified formula with the formula $\bar{x} \vee \bar{y} \leftrightarrow \bar{z} = (\bar{x} \vee \bar{y} \vee z) \wedge (x \vee x \vee \bar{z}) \wedge (y \vee y \vee \bar{z})$ using the $\wedge$ operator. As with the previous argument, we can say that ϕ_i^* is a sandwiched CNF formula.

Let ϕ' be the sandwiched CNF formula obtained from $\phi^* = \phi_0^*$ by repeating the above operation until all clauses have size exactly 3. Since exactly two replacements occur per clause in ϕ^* , we have $\phi' = \phi_{2mn^2}^*$. Moreover, six new clauses and two new variables are added for each clause in ϕ^* . Therefore, ϕ' has $7mn^2$ clauses and $n + 2mn^2$ variables.

To complete our reduction, for each $i \in [2mn^2]$, we will show that there exists a mixed variable assignment that satisfies ϕ_{i-1}^* if and only if there exists a mixed variable assignment that satisfies ϕ_i^* . This immediately implies that $\phi^* = \phi_0^*$ is a yes-instance of INTE5-SAT if and only if $\phi' = \phi_{2mn^2}^*$ is a yes-instance of INTE3-SAT. Let X_i be the set of variables in ϕ_i^* .

Suppose that there is a mixed variable assignment b_{i-1} for X_{i-1} that satisfies ϕ_{i-1}^* . Consider a variable assignment b_i for $X_i = X_{i-1} \cup \{z\}$ such that $b_i(x) = b_{i-1}(x)$ for each $x \in X_{i-1}$. Moreover, set $b_i(z) = x \vee y$ if z replaces $x \vee y$, and set $b_i(z) = \bar{x} \vee \bar{y}$ if z replaces $\bar{x} \vee \bar{y}$. Since b_{i-1} is a mixed variable assignment, b_i is also a mixed variable assignment. Furthermore, it is easy to see that b_i satisfies ϕ_i^* .

Conversely, suppose that there is a mixed variable assignment b_i for X_i that satisfies ϕ_i^* . Due to $x \vee y \leftrightarrow z$ or $\bar{x} \vee \bar{y} \leftrightarrow \bar{z}$, the variable assignment b_{i-1} such that $b_{i-1}(x) = b_i(x)$ for each $x \in X_{i-1}$ satisfies ϕ_{i-1}^* . We claim that the variable assignment b_{i-1} is mixed. For the sake of contradiction, assume that b_{i-1} is not mixed, that is, b_{i-1} is either the all- T assignment or all- F assignment. If $x \vee y \leftrightarrow z$ is added into ϕ_{i-1}^* , then $b_i(x) = b_i(y) = b_i(z)$ holds. Thus, b_i is also not mixed, a contradiction. Similarly, if $\bar{x} \vee \bar{y} \leftrightarrow \bar{z}$ is added into ϕ_{i-1}^* , then $b_i(x) = b_i(y) = b_i(z)$ holds, a contradiction. Therefore, we conclude that b_{i-1} is a mixed variable assignment for X_{i-1} that satisfies ϕ_{i-1}^* . This completes the proof. $\blacktriangleleft$

3.1.2 Membership in NP

Next, we show that ISR under k -TJ with $k = |I| - \mu$ is in NP not only when μ is constant but also μ is at most $O(\log |I|)$ for graphs of bounded maximum degree and planar graphs of maximum degree $o(\frac{n}{\log n})$, where n is the number of vertices in the input graph.

► **Theorem 4.** *Let G be an input graph with n vertices, chromatic number $O(1)$ and maximum degree $o(\frac{n}{\log n})$, and let I be an initial independent set of G . ISR under k -TJ is in NP when $k = |I| - \mu \geq 1$ with any non-negative integer μ at most $O(\log |I|)$.*

To prove Theorem 4, we will evaluate the length of a shortest reconfiguration sequence between any two independent sets of an input graph. Firstly, we introduce a lemma on the maximum size of intersecting families of sets. Let N, r be positive integers with $N \geq r$, and let $L \subseteq \{0, 1, \dots, r-1\}$. We say that a family $\mathcal{F}$ of r -element subsets of $[N]$ is an (N, r, L) -system if $|F \cap F'| \in L$ holds for all distinct $F, F' \in \mathcal{F}$. Let $m(N, r, L)$ denote the maximum size of (N, r, L) -systems.

For any positive integers N, r, p with $N \geq r > p \geq 1$, the following upper bound is known (see, for example, [14, 25, 33]):

$$m(N, r, \{0, \dots, p-1\}) \leq \binom{N}{p} / \binom{r}{p}. \quad (2)$$

Using Equation (2), we prove Lemma 5, which provides an upper bound on the length of any shortest reconfiguration sequence in the general case.

► **Lemma 5.** *Let G be an input graph with $n = |V(G)|$, I and J be initial and target independent sets of G , and $\mu < |I|$ be any non-negative integer. If I and J are reconfigurable under k -TJ, then the length of a shortest reconfiguration sequence between I and J under k -TJ, where $k = |I| - \mu \geq 1$, is at most $O((\frac{n}{k})^\mu)$.*

Proof. When $\mu = 0$, there exists a reconfiguration sequence $\langle I, J \rangle$ of length 1. Therefore, we assume that $\mu \geq 1$. Consider any shortest reconfiguration sequence $\sigma = \langle I = I_0, I_1, \dots, I_\ell = J \rangle$ between I and J . Let $\mathcal{I} = \{I_i : i \in \{0, 1, \dots, \ell\}, i \text{ is even}\}$. Note that $|\mathcal{I}| = \lfloor \frac{\ell}{2} \rfloor + 1$. Since σ is the shortest reconfiguration sequence, for any two independent sets I_i and I_j in $\mathcal{I}$ with $i < j$, we have $|I_i \cap I_j| < \mu$; otherwise, I_i and I_j are adjacent under k -TJ and we can obtain a shorter reconfiguration sequence σ' from σ by removing all independent sets $I_{i+1}, \dots, I_{j-1}$, which is a contradiction. Then, we observe that $\mathcal{I}$ is an $(n, |I|, \{0, \dots, \mu - 1\})$ -system. By using Equation (2), we have

$$|\mathcal{I}| = \left\lfloor \frac{\ell}{2} \right\rfloor + 1 \leq m(n, |I|, \{0, \dots, \mu - 1\}) \leq \binom{n}{\mu} / \binom{|I|}{\mu} \leq \left(\frac{n}{|I| - \mu} \right)^\mu.$$

Therefore, the length ℓ of σ is at most $O((\frac{n}{|I| - \mu})^\mu) = O((\frac{n}{k})^\mu)$. ◀

Now, we can prove Theorem 4.

Proof of Theorem 4. It is trivial when $\mu = 0$, thus we assume that $\mu \geq 1$. Suppose first that $2\mu \geq |I|$. Then, there is some constant c such that $|I| < c$ since $\mu = O(\log |I|)$. We can solve ISR under k -TJ by enumerating all independent sets of G with constant size in polynomial time.

Suppose next that $2\mu < |I|$ for sufficiently large $|I|$. Let A and B be two independent sets such that $A \subseteq I$ and $B \subseteq J$ with size exactly μ . Let G^* be the subgraph of G obtained by removing all vertices in $N[A \cup B]$. If G^* has an independent set I^* with size k , then there is a reconfiguration sequence $\langle I, A \cup I^*, B \cup I^*, J \rangle$ between I and J . We say that this reconfiguration sequence is a *simple reconfiguration sequence*. Since the vertex set of G^* is $V(G) \setminus N[A \cup B]$, the size of $V(G^*)$ is at least $n - 2\mu(\Delta + 1)$, where Δ is the maximum degree of G . Let χ and χ^* be the chromatic numbers of G and G^* , respectively. Then, we observe that $\chi^* \leq \chi$. By the relationship between the chromatic number and the independence number, G^* has an independent set with size at least $|V(G^*)|/\chi^* \geq (n - 2\mu(\Delta + 1))/\chi^*$. Thus, if $(n - 2\mu(\Delta + 1))/\chi^* \geq k$, then I and J are always reconfigurable, as a simple reconfiguration sequence exists. Note that the length of a simple reconfiguration sequence is 3.

It remains to consider the case where $(n - 2\mu(\Delta + 1))/\chi^* < k$, in which a simple reconfiguration sequence between I and J may not exist. Combined with Lemma 5, the length ℓ of a shortest reconfiguration sequence between I and J satisfies

$$\ell = O\left(\left(\frac{n}{k}\right)^\mu\right) = O\left(\left(\frac{n\chi^*}{n - 2\mu(\Delta + 1)}\right)^\mu\right) = O\left((\chi^*)^\mu \left(\frac{1}{1 - \frac{2\mu(\Delta + 1)}{n}}\right)^\mu\right). \quad (3)$$

Since $\mu = O(\log |I|) = O(\log n)$ and $\Delta = o(\frac{n}{\log n})$, we have $\mu\Delta = o(n)$. Hence, we have $(2\mu(\Delta + 1))/n = o(1)$. Furthermore, $\chi^* \leq \chi = O(1)$. Thus, from Equation (3), we have

$$\ell = O((\chi^*)^\mu \left(\frac{1}{1 - o(1)}\right)^\mu) = O(1)^{O(\log n)} = O(n^{O(1)}).$$

Therefore, ℓ is polynomially bounded in n , and hence the problem belongs to NP. This completes the proof. ◀

It is known that the chromatic number of G is at most $\Delta + 1$, where Δ is the maximum degree of G [8]. In addition, the chromatic number of any planar graph is at most 4 [1, 2]. Therefore, Theorem 4 gives the results including graphs of bounded maximum degree and planar graphs of maximum degree $o(\frac{n}{\log n})$.

3.2 VCR

In Section 3.1, we showed that ISR under k -TJ with $k = |I| - \mu$, where μ is any fixed positive integer, is NP-complete even for graphs of maximum degree 3 and for planar graphs of maximum degree 4. In contrast to this intractability, VCR under k -TJ is in XP for general graphs when parameterized by $\mu = |S| - k > 0$, where S is an initial vertex cover of an input graph.

► **Theorem 6 (*)**. *VCR under k -TJ is in XP for general graphs G when parameterized by $\mu = |S| - k \geq 0$, where S is an initial vertex cover of G .*

In the proof of Theorem 6, we present an XP algorithm for the problem.

Let $(G, S, T, k\text{-TJ})$ be an instance of VCR. We can consider the *reconfiguration graph* $\mathcal{C} = (\mathcal{V}, \mathcal{E})$ for the instance, such that each node $ws' \in \mathcal{V}$ corresponds to a vertex cover S' of G with size exactly $|S|$, and edges represent adjacency under k -TJ. Since the number of such vertex covers can be superpolynomial, explicitly constructing $\mathcal{C}$ is infeasible in general.

Our approach builds on the clique-compressed reconfiguration graph technique introduced in [35], which compactly represents $\mathcal{C}$ by grouping cliques into single nodes. This compressed graph has at most $O(n^\mu)$ nodes and preserves essential connectivity, making it sufficient for solving the problem if constructed efficiently.

Here, we briefly describe the characteristics of instances that enable the construction of our XP algorithm. If $|S \cap T| \geq \mu$, then S and T are reconfigurable, since we can simultaneously move $k = |S| - \mu$ tokens. Therefore, we focus on the case where $|S \cap T| < \mu$. Since both S and T are vertex covers, the induced subgraph $G[V(G) \setminus (S \cap T)]$ is a bipartite graph. This restricted structural property serves as a key ingredient in the design of our XP algorithm.

4 PSPACE-completeness When k is Constant

In this section, we investigate the PSPACE-completeness of ISR under k -TJ and k -TS when k is fixed. Note that the computational complexity of ISR and VCR under k -TJ and k -TS is the same when k is fixed, due to their complementary relationship.

4.1 Planar Graphs

► **Theorem 7 (*)**. *Let $k \geq 2$ be any fixed positive integer. ISR under $R \in \{k\text{-TS}, k\text{-TJ}\}$ is PSPACE-complete for planar graphs of maximum degree 3 and bounded bandwidth.*

To prove Theorem 7, we construct a reduction from NONDETERMINISTIC CONSTRAINT LOGIC, which was invented by Hearn and Demaine [18] and has been used to prove the PSPACE-hardness of reconfiguration problems, including ISR under TS [18].

4.2 Line Graphs and Claw-free Graphs

We state that ISR under $R \in \{2\text{-TJ}, 2\text{-TS}\}$ is PSPACE-complete even for line graphs, which contrasts that ISR under $R \in \{TJ, TS\}$ can be solved in polynomial time for line graphs (more generally, claw-free graphs) [6]. For a graph G , its *line graph* $L(G)$ is defined as follows: each vertex of $L(G)$ corresponds to an edge of G , and two vertices in $L(G)$ are adjacent if and only if their corresponding edges in G share a common endpoint.

► **Theorem 8 (*)**. *ISR under $R \in \{2\text{-TJ}, 2\text{-TS}\}$ is PSPACE-complete for line graphs.*

In the proof of Theorem 8, we reduce PERFECT MATCHING RECONFIGURATION, which is known to be PSPACE-complete on bipartite graphs of maximum degree 5 and bounded bandwidth [5], to our problem.

5 PSPACE-completeness When k is Superconstant

This section is devoted to establishing the PSPACE-completeness of ISR and VCR under $k\text{-TJ}$ when k is superconstant in the initial independent set size $|I|$ and the initial vertex cover size $|S|$, respectively.

5.1 ISR

The main result in this section is the following.

► **Theorem 9**. *There exists some constant $\varepsilon_0 \in (0, 1)$ such that ISR under $k\text{-TJ}$ on graphs of maximum degree 3 is PSPACE-complete for any k satisfying the following condition: there exists a constant c such that $k \leq \varepsilon_0 |I|$ holds whenever $|I| \geq c$, where I is the initial independent set of the input graph.*

To prove Theorem 9, we will construct a polynomial-time reduction from the *optimization variant* of ISR called MAXMIN INDEPENDENT SET RECONFIGURATION (MAXMINISR for short) [20]. In the problem, we adopt the *token addition and removal* (TAR for short) [20], under which two independent sets of a graph G are adjacent if one is obtained from the other by adding or removing a single vertex of G . In MAXMINISR, given a graph G and two independent sets I, J of G , we are asked to find a reconfiguration sequence $\sigma = \langle I = I_0, I_1, \dots, I_\ell = J \rangle$ under TAR that maximizes $\text{val}(\sigma)$, where $\text{val}(\sigma) = \min\{|I'| : I' \in \sigma\}$. For a graph G , we use $\alpha(G)$ to denote the number of vertices in a maximum independent set of G . Let (G, I, J) be an instance of MAXMINISR, and $\text{val}_{\max}(I, J)$ be the maximum value of $\text{val}(\sigma)$ over all possible reconfiguration sequences σ from I to J under TAR. Recently, the following Theorem 10 was proven [19, 24, 32].

► **Theorem 10** ([19, 24, 32]). *Let I and J be initial and target independent sets of an input graph G in MAXMINISR. Then, there exists some constant $\varepsilon_0 \in (0, 1)$ such that it is PSPACE-hard to distinguish between the following two cases:*

- (i) $\text{val}_{\max}(I, J) \geq \alpha(G) - 1$, and
- (ii) $\text{val}_{\max}(I, J) < (1 - \varepsilon_0)(\alpha(G) - 1)$.

The same hardness result holds even when the maximum degree of G is 3, $|I| = |J| = \alpha(G)$, and $\frac{\alpha(G)}{|V(G)|} \in [\frac{1}{3}, \frac{1}{2}]$.

To lead to Theorem 9 from Theorem 10, we provide the following lemma.

► **Lemma 11**. *Let I and J be initial and target independent sets of an input graph G in MAXMINISR and ISR. Let f be a given function and g be any function defined on integers such that $x - g(x) \geq 1$ for all positive integers x and there exists a fixed positive integer n_0 satisfying $g(n) \geq f(n)$ for all integers $n \geq n_0$. Suppose that it is PSPACE-hard to distinguish between the following two cases:*

- (i) $\text{val}_{\max}(I, J) \geq |I| - 1$, and
- (ii) $\text{val}_{\max}(I, J) < f(|I|)$.

Then, ISR under $k\text{-TJ}$ is PSPACE-hard, where $k = |I| - g(|I|) \geq 1$.

Proof. Let (G, I, J) be an instance of MAXMINISR, and $(G, I, J, k\text{-TJ})$ be an instance of ISR where $k = |I| - g(|I|)$. We assume that $|I| \geq n_0$ and hence $k \geq 1$ since we can solve all instances with $|I| < n_0$ by enumerating all independent sets of constant size. We now show that if (G, I, J) satisfies condition (i), then $(G, I, J, k\text{-TJ})$ is a yes-instance, and if (G, I, J) satisfies condition (ii), then $(G, I, J, k\text{-TJ})$ is a no-instance. We will show the latter one by proving the contrapositive: if $(G, I, J, k\text{-TJ})$ is a yes-instance, then (G, I, J) does not satisfy condition (ii).

Firstly, suppose that there is a reconfiguration sequence $\sigma = \langle I = I_0, I_1, \dots, I_\ell = J \rangle$ between I and J under TAR such that $\text{val}(\sigma) \geq |I| - 1$. It is known that this assumption holds if and only if there is a reconfiguration sequence under TJ between I and J [23]. Thus, there is a reconfiguration sequence under TJ between I and J , and that is also a reconfiguration sequence under $k\text{-TJ}$. Therefore, $(G, I, J, k\text{-TJ})$ is a yes-instance.

Conversely, suppose that there is a reconfiguration sequence $\sigma' = \langle I = I_0, I_1, \dots, I_\ell = J \rangle$ between I and J under $k\text{-TJ}$ where $k = |I| - g(|I|) \geq 1$. Then, for any two consecutive independent sets I_{i-1} and I_i with $i \in [\ell]$, we have $|I_{i-1} \cap I_i| \geq |I| - k = g(|I|)$. Additionally, we can transform from I_{i-1} to I_i under TAR as follows: Firstly, we remove tokens on vertices in $I_{i-1} \setminus I_i$ one by one; then, we add tokens on vertices in $I_i \setminus I_{i-1}$ one by one. Through these steps, we have no independent set with size smaller than $|I_{i-1} \cap I_i| \geq g(|I|) \geq f(|I|)$. Therefore, there is a sequence σ under TAR such that $\text{val}(\sigma) \geq g(|I|) \geq f(|I|)$. That is, (G, I, J) does not satisfy condition (ii). This completes the proof. $\blacktriangleleft$

We set $f(x) = (1 - \varepsilon_0)(x - 1)$ and let $g(x)$ be an arbitrary function such that $x - 1 \geq g(x) \geq f(x)$ for all $x \geq x_0$, for some constant x_0 . For example, $g(x)$ may be chosen as $x - c$ for some constant c , $x - \lceil \log x \rceil$, $x - \lceil x^{1/2} \rceil$, or $x - \lceil \varepsilon x \rceil$ for some constant $\varepsilon \leq \varepsilon_0$. Combining Theorem 10 and Lemma 11, ISR under $k\text{-TJ}$ on graphs of maximum degree 3 is PSPACE-complete for $k = |I| - g(|I|)$, as claimed in Theorem 9. This result includes the PSPACE-completeness of ISR under $k\text{-TJ}$ for various values of k , such as $\Theta(1)$, $\Theta(\log |I|)$, $\Theta(|I|^{O(1)})$, and $\Theta(|I|)$.

5.2 VCR

Similarly to Theorem 9, we can prove the following Theorem 12.

► **Theorem 12 (*)**. *There exists some constant $\varepsilon_0 \in (0, 1)$ such that VCR under $k\text{-TJ}$ on graphs of maximum degree 3 is PSPACE-complete for any k satisfying the following condition: there exists a constant c such that $k \leq \varepsilon_0 |S|$ holds whenever $|S| \geq c$, where S is the initial vertex cover of the input graph.*

6 Conclusion and Future Work

In this paper, we investigated the computational complexity of the fundamental reconfiguration problems ISR and VCR on various graph classes under the extended reconfiguration rules $k\text{-TJ}$ and $k\text{-TS}$.

The following open problems are suggested for future research: (1) Is ISR under $k\text{-TJ}$ with $k = |I| - \mu$ NP-hard on planar graphs of maximum degree 3 for any fixed positive integer μ ? (2) Is ISR under $k\text{-TJ}$ with $k = |I| - \mu$ in NP on general graphs not only when μ is fixed but also when $\mu = O(\log |I|)$? (See also Table 1.)

References

- 1 Kenneth Appel and Wolfgang Haken. Every planar map is four colorable. part I: Discharging. *Illinois J. Math.*, 21(3):429–490, 1977. doi:10.1215/ijm/1256049011.
- 2 Kenneth Appel, Wolfgang Haken, and John Koch. Every planar map is four colorable. part II: Reducibility. *Illinois J. Math.*, 21(3):491–567, 1977. doi:10.1215/ijm/1256049012.
- 3 Rémy Belmonte, Eun Jung Kim, Michael Lampis, Valia Mitsou, Yota Otachi, and Florian Sikora. Token sliding on split graphs. *Theory Comput. Syst.*, 65(4):662–686, 2021. doi:10.1007/S00224-020-09967-8.
- 4 Marthe Bonamy and Nicolas Bousquet. Token sliding on chordal graphs. In *WG*, pages 127–139, 2017. doi:10.1007/978-3-319-68705-6_10.
- 5 Marthe Bonamy, Nicolas Bousquet, Marc Heinrich, Takehiro Ito, Yusuke Kobayashi, Arnaud Mary, Moritz Mühlenthaler, and Kunihiro Wasa. The perfect matching reconfiguration problem. In *MFCS*, pages 80:1–80:14, 2019. doi:10.4230/LIPICS.MFCS.2019.80.
- 6 Paul S. Bonsma, Marcin Kamiński, and Marcin Wrochna. Reconfiguring independent sets in claw-free graphs. In *SWAT*, pages 86–97, 2014. doi:10.1007/978-3-319-08404-6_8.
- 7 Marcin Briański, Stefan Felsner, Jędrzej Hodor, and Piotr Micek. Reconfiguring independent sets on interval graphs. In *MFCS*, pages 23:1–23:14, 2021. doi:10.4230/LIPICS.MFCS.2021.23.
- 8 R. L. Brooks. On colouring the nodes of a network. *Mathematical Proceedings of the Cambridge Philosophical Society*, 37(2):194–197, 1941. doi:10.1017/S030500410002168X.
- 9 Remo Christen, Salomé Eriksson, Michael Katz, Christian Muise, Alice Petrov, Florian Pommerening, Jendrik Seipp, Silvan Sievers, and David Speck. PARIS: planning algorithms for reconfiguring independent sets. In *ECAI*, pages 453–460, 2023. doi:10.3233/FAIA230303.
- 10 Mark de Berg, Bart M. P. Jansen, and Debajyoti Mukherjee. Independent-set reconfiguration thresholds of hereditary graph classes. *Discret. Appl. Math.*, 250:165–182, 2018. doi:10.1016/J.DAM.2018.05.029.
- 11 Erik D. Demaine, Martin L. Demaine, Eli Fox-Epstein, Duc A. Hoang, Takehiro Ito, Hirotaka Ono, Yota Otachi, Ryuhei Uehara, and Takeshi Yamada. Linear-time algorithm for sliding tokens on trees. *Theor. Comput. Sci.*, 600:132–142, 2015. doi:10.1016/J.TCS.2015.07.037.
- 12 Naoki Domon, Akira Suzuki, Yuma Tamura, and Xiao Zhou. The shortest path reconfiguration problem based on relaxation of reconfiguration rules. In *WALCOM*, pages 227–241, 2024. doi:10.1007/978-981-97-0566-5_17.
- 13 Paul Erdős, Chao Ko, and Richard Rado. Intersection theorems for systems of finite sets. *Q. J. Math.*, 12(1):313–320, 1961. doi:10.1093/qmath/12.1.313.
- 14 Peter Frankl and Norihide Tokushige. Invitation to intersection problems for finite sets. *J. Comb. Theory A*, 144:157–211, 2016. doi:10.1016/J.JCTA.2016.06.017.
- 15 M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.
- 16 Arash Haddadan, Takehiro Ito, Amer E. Mouawad, Naomi Nishimura, Hirotaka Ono, Akira Suzuki, and Youcef Tebbal. The complexity of dominating set reconfiguration. *Theor. Comput. Sci.*, 651:37–49, 2016. doi:10.1016/J.TCS.2016.08.016.
- 17 Hiroki Hatano, Naoki Kitamura, Taisuke Izumi, Takehiro Ito, and Toshimitsu Masuzawa. Independent set reconfiguration under bounded-hop token jumping. In *WALCOM*, pages 215–228, 2025. doi:10.1007/978-981-96-2845-2_14.
- 18 Robert A. Hearn and Erik 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):72–96, 2005. doi:10.1016/J.TCS.2005.05.008.
- 19 Shuichi Hirahara and Naoto Ohsaka. Probabilistically checkable reconfiguration proofs and inapproximability of reconfiguration problems. In *STOC*, pages 1435–1445, 2024. doi:10.1145/3618260.3649667.
- 20 Takehiro Ito, Erik D. Demaine, Nicholas J. A. Harvey, Christos H. Papadimitriou, Martha Sideri, Ryuhei Uehara, and Yushi Uno. On the complexity of reconfiguration problems. *Theor. Comput. Sci.*, 412(12-14):1054–1065, 2011. doi:10.1016/J.TCS.2010.12.005.

- 21 Takehiro Ito, Jun Kawahara, Yu Nakahata, Takehide Soh, Akira Suzuki, Junichi Teruyama, and Takahisa Toda. ZDD-based algorithmic framework for solving shortest reconfiguration problems. In *CPAIOR*, pages 167–183, 2023. doi:10.1007/978-3-031-33271-5_12.
- 22 Takehiro Ito, Hirotaka Ono, and Yota Otachi. Reconfiguration of cliques in a graph. *Discret. Appl. Math.*, 333:43–58, 2023. doi:10.1016/J.DAM.2023.01.026.
- 23 Marcin Kamiński, Paul Medvedev, and Martin Milanič. Complexity of independent set reconfigurability problems. *Theor. Comput. Sci.*, 439:9–15, 2012. doi:10.1016/J.TCS.2012.03.004.
- 24 Karthik C. S. and Pasin Manurangsi. On inapproximability of reconfiguration problems: PSPACE-hardness and some tight NP-hardness results. *CoRR*, abs/2312.17140, 2023. doi:10.48550/arXiv.2312.17140.
- 25 Gyula Katona, Tibor Nemetz, and Miklos Simonovits. On a graph-problem of Turán. *Matematikai Lapok*, 15, 1964.
- 26 Jan Matyáš Kříšťan and Jakub Svoboda. Reconfiguration using generalized token jumping. In *WALCOM*, pages 244–265, 2025. doi:10.1007/978-981-96-2845-2_16.
- 27 Daniel Lokshtanov and Amer E. Mouawad. The complexity of independent set reconfiguration on bipartite graphs. *ACM Trans. Algorithms*, 15(1):7:1–7:19, 2019. doi:10.1145/3280825.
- 28 Meena Mahajan and Venkatesh Raman. Parameterizing above guaranteed values: MaxSat and MaxCut. *J. Algorithms*, 31(2):335–354, 1999. doi:10.1006/JAGM.1998.0996.
- 29 Amer E. Mouawad, Naomi Nishimura, Venkatesh Raman, and Sebastian Siebertz. Vertex cover reconfiguration and beyond. *Algorithms*, 11(2):20, 2018. doi:10.3390/A11020020.
- 30 Amer E. Mouawad, Naomi Nishimura, Venkatesh Raman, Narges Simjour, and Akira Suzuki. On the parameterized complexity of reconfiguration problems. *Algorithmica*, 78(1):274–297, 2017. doi:10.1007/S00453-016-0159-2.
- 31 Naomi Nishimura. Introduction to reconfiguration. *Algorithms*, 11(4):52, 2018. doi:10.3390/A11040052.
- 32 Naoto Ohsaka. Gap preserving reductions between reconfiguration problems. In *STACS*, pages 49:1–49:18, 2023. doi:10.4230/LIPICS.STACS.2023.49.
- 33 Vojtech Rödl. On a packing and covering problem. *Eur. J. Comb.*, 6(1):69–78, 1985. doi:10.1016/S0195-6698(85)80023-8.
- 34 Walter J. Savitch. Relationships between nondeterministic and deterministic tape complexities. *J. Comput. Syst. Sci.*, 4(2):177–192, 1970. doi:10.1016/S0022-0000(70)80006-X.
- 35 Tatsuhiro Suga, Akira Suzuki, Yuma Tamura, and Xiao Zhou. Changing induced subgraph isomorphisms under extended reconfiguration rules. In *WALCOM*, pages 346–360, 2025. doi:10.1007/978-981-96-2845-2_22.
- 36 Akira Suzuki, Amer E. Mouawad, and Naomi Nishimura. Reconfiguration of dominating sets. *J. Comb. Optim.*, 32(4):1182–1195, 2016. doi:10.1007/S10878-015-9947-X.
- 37 Takahisa Toda, Takehiro Ito, Jun Kawahara, Takehide Soh, Akira Suzuki, and Junichi Teruyama. Solving reconfiguration problems of first-order expressible properties of graph vertices with boolean satisfiability. In *ICTAI*, pages 294–302, 2023. doi:10.1109/ICTAI59109.2023.00050.
- 38 Jan van den Heuvel. The complexity of change. In *Surveys in Combinatorics 2013*, volume 409 of *London Mathematical Society Lecture Note Series*, pages 127–160. Cambridge University Press, 2013. doi:10.1017/CB09781139506748.005.
- 39 Tom C. van der Zanden. Parameterized complexity of graph constraint logic. In *IPEC*, pages 282–293, 2015. doi:10.4230/LIPICS.IPEC.2015.282.
- 40 Marcin Wrochna. Reconfiguration in bounded bandwidth and tree-depth. *J. Comput. Syst. Sci.*, 93:1–10, 2018. doi:10.1016/J.JCSS.2017.11.003.
- 41 Yuya Yamada, Mutsunori Banbara, Katsumi Inoue, Torsten Schaub, and Ryuhei Uehara. Combinatorial reconfiguration with answer set programming: Algorithms, encodings, and empirical analysis. In *WALCOM*, pages 242–256, 2024. doi:10.1007/978-981-97-0566-5_18.