Reconfiguration of Colorable Sets in Classes of Perfect Graphs

Takehiro Ito
Graduate School of Information Sciences, Tohoku University.
Aoba-yama 6-6-05, Sendai, 980-8579, Japan
takehiro@ecei.tohoku.ac.jp
https://orcid.org/0000-0002-9912-6898

Yota Otachi
Faculty of Advanced Science and Technology, Kumamoto University.
2-39-1 Kurokami, Chuo-ku, Kumamoto, 860-8555 Japan
otachi@cs.kumamoto-u.ac.jp
https://orcid.org/0000-0002-0087-853X

Abstract

A set of vertices in a graph is *c-colorable* if the subgraph induced by the set has a proper c-coloring. In this paper, we study the problem of finding a step-by-step transformation (reconfiguration) between two c-colorable sets in the same graph. This problem generalizes the well-studied Independent Set Reconfiguration problem. As the first step toward a systematic understanding of the complexity of this general problem, we study the problem on classes of perfect graphs. We first focus on interval graphs and give a combinatorial characterization of the distance between two c-colorable sets. This gives a linear-time algorithm for finding an actual shortest reconfiguration sequence for interval graphs. Since interval graphs are exactly the graphs that are simultaneously chordal and co-comparability, we then complement the positive result by showing that even deciding reachability is PSPACE-complete for chordal graphs and for co-comparability graphs. The hardness for chordal graphs holds even for split graphs. We also consider the case where c is a fixed constant and show that in such a case the reachability problem is polynomial-time solvable for split graphs but still PSPACE-complete for co-comparability graphs. The complexity of this case for chordal graphs remains unsettled. As by-products, our positive results give the first polynomial-time solvable cases (split graphs and interval graphs) for Feedback Vertex Set Reconfiguration.

2012 ACM Subject Classification Mathematics of computing → Graph algorithms

Keywords and phrases reconfiguration, colorable set, perfect graph

Digital Object Identifier 10.4230/LIPIcs.SWAT.2018.27

Related Version A full version of the paper is available at `https://arxiv.org/abs/1802.06511`.

Funding T.I. was partially supported by JST CREST Grant Number JPMJCR1402, and JSPS KAKENHI Grant Number JP16K00004, Japan. Y.O. was partially supported by MEXT KAKENHI Grant Number JP24106004, JSPS KAKENHI Grant Number JP25730003, and by FY 2015 Researcher Exchange Program between JSPS and NSERC.

16th Scandinavian Symposium and Workshops on Algorithm Theory (SWAT 2018).
Editor: David Eppstein; Article No. 27; pp. 27:1–27:13
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

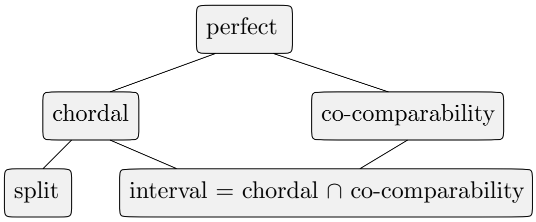

Figure 1 The graph classes studied in this paper.

1 Introduction

Recently, the reconfiguration framework has been applied to several search problems. In a reconfiguration problem, we are given two feasible solutions of a search problem and are asked to determine whether we can modify one to the other by repeatedly applying prescribed reconfiguration rules while keeping the feasibility (see [14, 22, 19]). Studying such a problem is important for understanding the structure of the solution space of the underlying problem. Computational complexity of reconfiguration problems has been studied intensively. For example, the Independent Set Reconfiguration problem under the reconfiguration rules TS [13], TAR [14], and TJ [15] has been studied for several graph classes such as planar graphs [13], perfect graphs [15], claw-free graphs [5], trees [6], interval graphs [4], and bipartite graphs [17].

In this paper, we initiate the study on the problem of reconfiguring *colorable sets*, which generalizes Independent Set Reconfiguration. For a graph $G = (V, E)$ and an integer $c \geq 1$, a vertex set $S \subseteq V$ is *c-colorable* if the subgraph $G[S]$ induced by S admits a proper c-coloring. For example, the 1-colorable sets in a graph are exactly the independent sets of the graph. Recently, c-colorable sets have been studied from the viewpoint of wireless network optimization (see [2, 3] and the references therein). The Colorable Set Reconfiguration problem asks given two c-colorable sets S and S' in a graph G, whether we can reach from S to S' by repeatedly applying allowed local changes. We consider the following three local change operations (see Section 2 for formal definitions):

- TAR(k): either adding or removing one vertex while keeping the size of the set at least a given threshold k.
- TJ: swap one member for one nonmember.
- TS: swap one member for one nonmember adjacent to the member.

In perfect graphs, being c-colorable is equivalent to having no clique of size more than c. This property often makes problems related to coloring tractable. Thus, to understand this very general problem, we start the study of Colorable Set Reconfiguration on classes of perfect graphs. Figure 1 shows the graph classes studied in this paper and the inclusion relationships (see Section 2.2 for definitions).

Our contribution

Before we start our investigation on the reconfiguration problem, we first fill a gap in the complexity landscape of the search problem Colorable Set that asks for finding a large c-colorable set. When $c = 1$, Colorable Set is equivalent to the classical problem of finding a large independent set that can be solved in polynomial time for perfect graphs. For larger c, it was only known that the case $c = 2$ is NP-complete for perfect graphs [1]. To

Table 1 Summary of the results. PSPACE-completeness results here apply to TS also, while polynomial-time algorithms do not. The case of $c = 1$ is equivalent to Independent Set Reconfiguration.

	Colorable Set Reconfiguration under TAR/TJ		
	$c = 1$	fixed $c \geq 2$	arbitrary c
perfect	PSPACE-c		
co-comparability	PSPACE-c [15][2]	PSPACE-c (omitted)	
chordal	P [15]	?	PSPACE-c
split	P	P (Thm 4.4)	PSPACE-c (Thm 4.5)
interval	P	P (Thm 3.11)	
bipartite	NP-c [17]	Trivial if $c \geq 2$	

make the complexity status of Colorable Set for perfect graphs complete, we show that it is NP-complete for any fixed $c \geq 2$.[1]

We then show complexity divergences among the classes of perfect graphs in Figure 1, in particular under TAR and TJ. See Table 1 for a summary of our results. Our results basically say that the problem under TAR and TJ is tractable on interval graphs but further generalization is not quite possible.

More specifically, we first study the problem on interval graphs and show that a shortest reconfiguration sequence under TAR can be found in linear time (Theorem 3.11). This implies the same result under TJ. Next we study the problem on split graphs. We show that the complexity depends on c. When c is a fixed constant, the problem is polynomial-time solvable under TAR and TJ (Theorem 4.4). If c is a part of input, then we can show that the problem is PSPACE-complete under all rules, including TS (Theorem 4.5). While the hardness result applies also to chordal graphs, it is unclear whether a similar positive result for chordal graphs can be obtained when c is a fixed constant. We only know that the case of $c = 1$ under TAR and TJ is polynomial-time solvable as chordal graphs are even-hole-free [15]. We also show that for every fixed $c \geq 1$ the problem is PSPACE-complete for co-comparability graphs under all rules.[3] Thus, our results are in some sense tight since the interval graphs are exactly the chordal co-comparability graphs and split graphs are chordal graphs (see Figure 1).

As a byproduct of Theorems 3.11 and 4.4, the Feedback Vertex Set Reconfiguration problem [18] turns out to be polynomial-time solvable for split graphs and interval graphs under TAR and TJ. These are the first polynomial-time solvable cases for Feedback Vertex Set Reconfiguration. To see the polynomial-time solvability, observe that the complements $V(G) \setminus S$ of 2-colorable sets S in a chordal graph G are exactly the feedback vertex sets in the graph[4] and reconfigurations of the complements are equivalent to reconfigurations of the original vertex sets under TAR and TJ.

1 This result is omitted in this version.
2 The reduction in [15] outputs co-comparability graphs.
3 This result is omitted in this version.
4 Each induced cycle in a chordal graph is a triangle, and thus 2-colorable (or equivalently, odd cycle free) chordal graphs are forests.

2 Preliminaries

We say, as usual, that an algorithm for a graph $G = (V, E)$ runs in *linear time* if the running time of the algorithm is $O(|V| + |E|)$.

A *proper c-coloring* of a graph assigns a color from $\{1, \dots, c\}$ to each vertex in such a way that adjacent vertices have different colors. Given a graph G and an integer c, Graph Coloring asks whether G admits a proper c-coloring. This problem is NP-complete even if c is fixed to 3 [9]. The minimum c such that a graph admits a proper c-coloring is its *chromatic number*.

The Colorable Set problem is a generalization of Graph Coloring where we find a large induced subgraph of the input graph that admits a proper c-coloring. Let $G = (V, E)$ be a graph. For a set of vertices $S \subseteq V$, we denote by $G[S]$ the subgraph induced by S. A vertex set $S \subseteq V$ is *c-colorable* in G if $G[S]$ has a proper c-coloring. Now the problem is defined as follows:

Problem: Colorable Set
Input: A graph G and integers c and k.
Question: Does G have a c-colorable set of size at least k?

The problem of finding a large c-colorable set is studied for a few important classes of perfect graphs (see Figure 1 and Table 1). For the class of perfect graphs, it is known that a maximum 1-colorable set (that is, a maximum independent set) can be found in polynomial time [12]. Parameterized complexity [16] and approximation [7] of Colorable Set on perfect graphs are also studied.

2.1 Reconfiguration of colorable sets

Let S and S' be c-colorable sets in a graph G. Then, $S \leftrightarrow S'$ *under* $\mathsf{TAR}(k)$ for a nonnegative integer k if $|S|, |S'| \geq k$ and $|S \vartriangle S'| = 1$, where $S \vartriangle S'$ denotes the symmetric difference $(S \setminus S') \cup (S' \setminus S)$. Here $S \leftrightarrow S'$ means that S and S' can be reconfigured to each other in one step and TAR stands for "token addition & removal." A sequence $\langle S_0, S_1, \dots, S_\ell \rangle$ of c-colorable sets in G is a *reconfiguration sequence* of length ℓ between S_0 and S_ℓ under $\mathsf{TAR}(k)$ if $S_{i-1} \leftrightarrow S_i$ holds under $\mathsf{TAR}(k)$ for all $i \in \{1, 2, \dots, \ell\}$. A reconfiguration sequence under $\mathsf{TAR}(k)$ is simply called a $\mathsf{TAR}(k)$*-sequence*. We write $S_0 \leftrightsquigarrow S_\ell$ *under* $\mathsf{TAR}(k)$ if there exists a $\mathsf{TAR}(k)$-sequence between S_0 and S_ℓ. Note that every reconfiguration sequence is *reversible*, that is, $S_0 \leftrightsquigarrow S_\ell$ if and only if $S_\ell \leftrightsquigarrow S_0$. Now the problem we are going to consider is formalized as follows:

Problem: Colorable Set Reconfiguration under TAR ($\mathrm{CSR}_{\mathsf{TAR}}$ for short)
Input: A graph G, integers c and k, and c-colorable sets S and S' of G.
Question: Does $S \leftrightsquigarrow S'$ under $\mathsf{TAR}(k)$ hold?

We denote by (G, c, S, S', k) an instance of $\mathrm{CSR}_{\mathsf{TAR}}$. We assume that both $|S| \geq k$ and $|S'| \geq k$ hold; otherwise it is trivially a no-instance. Note that the lower bound k guarantees that none of the sets in the reconfiguration sequence is too small. Without the lower bound, the reachability problem becomes trivial as S can always reach S' via $\emptyset$.

For a $\mathrm{CSR}_{\mathsf{TAR}}$-instance (G, c, S, S', k), we denote by $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S')$ the length of a shortest $\mathsf{TAR}(k)$-sequence in G between S and S'; if there is no such a sequence, then we set $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S') = \infty$.

We note that $\mathrm{CSR}_{\mathsf{TAR}}$ is a decision problem and hence does not require the specification of an actual $\mathsf{TAR}(k)$-sequence. Similarly, the shortest variant of $\mathrm{CSR}_{\mathsf{TAR}}$ simply requires to output the value of $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S')$.

Other reconfiguration rules

Although the TAR rule is our main target, we also study two other well-known rules TJ (token jumping) and TS (token sliding). Let S and S' be c-colorable sets in a graph G. For TJ and TS, we additionally assume that $|S| = |S'|$ because these rules do not change the size of a set. Now the rules are defined as follows:

- $S \leftrightarrow S'$ under TJ if $|S \setminus S'| = |S' \setminus S| = 1$;
- $S \leftrightarrow S'$ under TS if $|S \setminus S'| = |S' \setminus S| = 1$ and the two vertices in $S \triangle S'$ are adjacent in G.

Reconfiguration sequences under TJ and TS as well as the reconfiguration problems $\mathrm{CSR}_{\mathsf{TJ}}$ and $\mathrm{CSR}_{\mathsf{TS}}$ are defined analogously. An instance of $\mathrm{CSR}_{\mathsf{TJ}}$ or $\mathrm{CSR}_{\mathsf{TS}}$ is represented as (G, c, S, S'), and $\mathsf{dist}_{\mathsf{TJ}}(S, S')$ and $\mathsf{dist}_{\mathsf{TS}}(S, S')$ are defined in the same way.

The following relation can be shown in almost the same way as Theorem 1 in [15] and means that $\mathrm{CSR}_{\mathsf{TJ}}$ is not harder than $\mathrm{CSR}_{\mathsf{TAR}}$ in the sense of Karp reductions.

▶ **Lemma 2.1.** *Let S and S' be c-colorable sets of size $k + 1$ in a graph G. Then, $S \leftrightsquigarrow S'$ under* TAR(k) *if and only if $S \leftrightsquigarrow S'$ under* TJ*. Furthermore, it holds that $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S') = 2 \cdot \mathsf{dist}_{\mathsf{TJ}}(S, S')$.*

To make the presentation easier, we often use the shorthands $S + v$ for $S \cup \{v\}$ and $S - v$ for $S \setminus \{v\}$. For a vertex v of a graph G, we denote the neighborhood of v in G by $N_G(v)$.

2.2 Graph classes

A *clique* in a graph is a set of pairwise adjacent vertices. A graph is *perfect* if the chromatic number equals the maximum clique size for every induced subgraph [11]. The following fact follows directly from the definition of perfect graphs and will be used throughout this paper.

▶ **Observation 2.2.** *A vertex set $S \subseteq V(G)$ of a perfect graph G is c-colorable if and only if $G[S]$ has no clique of size more than c.*

There are many subclasses of perfect graphs. Chordal graphs form one of the most well-known subclasses of perfect graphs, where a graph is *chordal* if it contains no induced cycle of length greater than 3.

Co-comparability graphs form another large class of perfect graphs. A graph $G = (V, E)$ is a *co-comparability graph* if there is a linear ordering $\prec$ on V such that $u \prec v \prec w$ and $\{u, w\} \in E$ imply $\{u, v\} \in E$ or $\{v, w\} \in E$. Although they are less known than chordal graphs, co-comparability graphs generalize several important graph classes such as interval graphs, permutation graphs, trapezoid graphs, and co-bipartite graphs (see [11, 20]).

The classes of chordal graphs and co-comparability graphs are incomparable.[5] It is known that the class of interval graphs characterizes their intersection; namely, a graph is an interval graph if and only if it is a co-comparability graph and chordal [10]. Recall that a graph is an *interval graph* if it is the intersection graph of closed intervals on the real line.

Another well-studied subclass of chordal graphs (and hence of perfect graphs) is the class of split graphs. A graph $G = (V, E)$ is a *split graph* if V can be partitioned into a clique K and an independent set I. To emphasize that G is a split graph, we write $G = (K, I; E)$. The classes of interval graphs and split graphs are incomparable.[6]

[5] A cycle of four vertices is a co-comparability graph but not chordal. The *net* graph obtained by attaching a pendant vertex to each vertex of a triangle is chordal but not a co-comparability graph.

[6] A path with five or more vertices is an interval graph but not a split graph. The net graph is a split graph but not an interval graph.

3 Shortest reconfiguration in interval graphs

In this section, we show that $\mathrm{CSR}_{\mathsf{TAR}}$ for interval graphs can be solved in linear time. Our result is actually stronger and says that an actual shortest $\mathsf{TAR}(k)$-sequence can be found in linear time, if one exists. By Lemma 2.1, the same result is obtained for TJ. We first give a characterization of the distance between two c-colorable sets in an interval graph (Section 3.1). This characterization says that a shortest $\mathsf{TAR}(k)$-sequence has length linear in the number of vertices of the graph. We then show that the distance can be computed in linear time (Section 3.2). We finally present a linear-time algorithm for finding a shortest $\mathsf{TAR}(k)$-sequence (Section 3.3).

It is known that a graph is an interval graph if and only if its maximal cliques can be ordered so that each vertex appears consecutively in that ordering [10, 8]. We call a list of the maximal cliques ordered in such a way a *clique path*. Let $G = (V, E)$ be an interval graph and $(M_1, \dots, M_t)$ be a clique path of G; that is, for each vertex $v \in V$, there are indices l_v and r_v such that $v \in M_i$ if and only if $l_v \le i \le r_v$. Given an interval graph, a clique path and the indices l_v and r_v for all vertices can be computed in linear time [21]. Hence we can assume that we are additionally given such information. Note that $\mathcal{I} = \{[l_v, r_v] : v \in V\}$ is an interval representation of G. Namely, $\{u, v\} \in E$ if and only $[l_u, r_u] \cap [l_v, r_v] \neq \emptyset$.

Let K be a clique in an interval graph G. By the Helly property of intervals, the intersection of all intervals in K is nonempty; that is, $\bigcap_{v \in K} [l_v, r_v] \neq \emptyset$ (see [20]). A point in the intersection $\bigcap_{v \in K} [l_v, r_v]$ is a *clique point* of K.

3.1 The distance between c-colorable sets

Let (G, c, S, S', k) be an instance of $\mathrm{CSR}_{\mathsf{TAR}}$. The set S is *locked in G* if S is a maximal c-colorable set in G and $|S| = k$. The following lemma follows immediately from the definition.

▶ **Lemma 3.1.** *Let G be a graph, and let S and S' be distinct c-colorable sets of size at least k in G. If S or S' is locked in G, then $S \not\leftrightsquigarrow S'$.*

Proof. Assume without loss of generality that S is locked in G. If there is a c-colorable set S_1 in G such that $S \leftrightarrow S_1$, then $S \subsetneq S_1$ as $|S| = k$. This contradicts the maximality of S. Since $S \neq S'$, we can conclude that $S \not\leftrightsquigarrow S'$. ◀

The rest of this subsection is dedicated to a proof of the following theorem, which implies that the converse of the lemma above also holds for interval graphs.

▶ **Theorem 3.2.** *Let G be an interval graph, and let S and S' be distinct c-colorable sets of size at least k in G. If S and S' are not locked in G, then the distance $d := \mathsf{dist}_{\mathsf{TAR}(k)}(S, S')$ is determined as follows.*

1. *If S and S' are not locked in $G[S \cup S']$, then $d = |S \bigtriangleup S'|$.*
2. *If exactly one of S and S' is locked in $G[S \cup S']$, then $d = |S \bigtriangleup S'| + 2$.*
3. *If S and S' are locked in $G[S \cup S']$, then we have the following two cases.*
   a. *If there is $v \in V(G) \setminus (S \cup S')$ such that both $S + v$ and $S' + v$ are c-colorable in G, then $d = |S \bigtriangleup S'| + 2$.*
   b. *Otherwise, $d = |S \bigtriangleup S'| + 4$.*

▶ **Corollary 3.3.** *For $S \neq S'$, $S \leftrightsquigarrow S'$ if and only if none of S and S' is locked in G.*

Observe that $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S') \ge |S \bigtriangleup S'|$ for any pair of c-colorable sets S and S' in G. We use this fact implicitly in the following arguments.

▶ **Lemma 3.4** (Theorem 3.2 (1)). *Let G be an interval graph, and let S and S' be c-colorable sets of size at least k in G. If S and S' are not locked in $G[S \cup S']$, then* $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S') = |S \vartriangle S'|$.

Proof. We proceed by induction on $|S \vartriangle S'|$. The base case of $|S \vartriangle S'| = 0$ is trivial. Assume that $|S \vartriangle S'| > 0$ and that the statement is true if the symmetric difference is smaller.

We first consider the case where $|S| = k$. Since S is not locked in $G[S \cup S']$, S is not maximal in $G[S \cup S']$. Thus there is a vertex $v \in S' \setminus S$ such that $T := S + v$ is c-colorable. The set T is not locked in $G[T \cup S']$, $S \leftrightarrow T$, and $|T \vartriangle S'| = |S \vartriangle S'| - 1$. By the induction hypothesis, $\mathsf{dist}_{\mathsf{TAR}(k)}(T, S') = |T \vartriangle S'| = |S \vartriangle S'| - 1$. Hence, we have $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S') \le \mathsf{dist}_{\mathsf{TAR}(k)}(T, S') + 1 = |S \vartriangle S'|$. If $|S'| = k$, we can apply the same argument.

In the following, we assume that $|S| > k$ and $|S'| > k$. If $S \subseteq S'$, then we can add the elements of $S' \setminus S$ one-by-one in an arbitrary order to get a shortest reconfiguration sequence of length $|S' \setminus S| = |S \vartriangle S'|$. The case where $S' \subseteq S$ is the same.

We now consider the case where $S \not\subseteq S'$ and $S' \not\subseteq S$. Let $v \in S \setminus S'$ and $w \in S' \setminus S$ be vertices with the smallest right-end in each set. That is, $r_v = \min\{r_x : x \in S \setminus S'\}$ and $r_w = \min\{r_x : x \in S' \setminus S\}$. By symmetry, assume that $r_w \le r_v$. Let $u \in S \setminus S'$ be a vertex that minimizes l_u. (Note that u and v may be the same.) Now we have $r_w \le r_v \le r_u$. We set $T = S - u$ and $T' = S - u + w$. Clearly, $S \leftrightarrow T$. To apply the induction hypothesis, it suffices to show that T is not locked in $G[T \cup S']$. To this end, we prove that $T \leftrightarrow T'$. Suppose to the contrary that T' is not c-colorable; that is, T' contains a clique K of size $c + 1$. Since T does not contain such a large clique, K must include w. Let p be a clique point of K. If $p < l_u$, then K includes no vertex in $S \setminus S'$ as u has the minimum l_u in $S \setminus S'$. This contradicts the c-colorability of S' and thus $l_u \le p \le r_w \le r_u$. This implies that $K - w + u \subseteq S$ is a clique of size $c + 1$, a contradiction. Therefore, we can conclude that T' is c-colorable. Now, by the induction hypothesis, $\mathsf{dist}_{\mathsf{TAR}(k)}(T, S') = |T \vartriangle S'| = |S \vartriangle S'| - 1$, and thus $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S') \le \mathsf{dist}_{\mathsf{TAR}(k)}(T, S') + 1 = |S \vartriangle S'|$. ◀

▶ **Lemma 3.5** (Theorem 3.2 (2)). *Let G be an interval graph, and let S and S' be distinct c-colorable sets of size at least k in G. If S and S' are not locked in G, and exactly one of S and S' is locked in $G[S \cup S']$, then* $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S') = |S \vartriangle S'| + 2$.

Proof. Without loss of generality, assume that S is locked in $G[S \cup S']$. This implies that $|S| = k$. Since S is not locked in G, S is not maximal in G. Hence, there is a vertex $v \in V(G) \setminus (S \cup S')$ such that $T := S + v$ is a c-colorable set of G. Observe that T and S' are not locked in $G[T \cup S']$. Thus, by Theorem 3.2 (1), it holds that $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S') \le \mathsf{dist}_{\mathsf{TAR}(k)}(T, S') + 1 = |T \vartriangle S'| + 1 = |S \vartriangle S'| + 2$.

On the other hand, since S is locked in $G[S \cup S']$, every c-colorable set T of G with $S \leftrightarrow T$ contains a vertex in $V(G) \setminus (S \cup S')$. Thus $|T \vartriangle S'| = |S \vartriangle S'| + 1$ holds. This implies that $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S') \ge \min_{T\colon S \leftrightarrow T} |T \vartriangle S'| + 1 = |S \vartriangle S'| + 2$. ◀

▶ **Lemma 3.6** (Theorem 3.2 (3a)). *Let G be an interval graph, and let S and S' be distinct c-colorable sets of size at least k in G. Assume S and S' are locked in $G[S \cup S']$ but not in G. If there is a vertex $v \in V(G) \setminus (S \cup S')$ such that both $S + v$ and $S' + v$ are c-colorable in G, then* $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S') = |S \vartriangle S'| + 2$.

Proof. Let $v \in V(G) \setminus (S \cup S')$ be a vertex such that both $S + v$ and $S' + v$ are c-colorable in G. We have $S \leftrightarrow S + v$ and $S' \leftrightarrow S' + v$. Since $S + v$ and $S' + v$ are not locked in $G[S \cup S' + v]$, Theorem 3.2 (1) implies that $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S') \le \mathsf{dist}_{\mathsf{TAR}(k)}(S + v, S' + v) + 2 = |(S + v) \vartriangle (S' + v)| + 2 = |S \vartriangle S'| + 2$.

The lower bound $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S') \geq |S \vartriangle S'| + 2$ can be shown in exactly the same way as in the proof of Lemma 3.5. ◀

▶ **Lemma 3.7** (Theorem 3.2 (3b)). *Let G be an interval graph, and let S and S' be distinct c-colorable sets of size at least k in G. Assume S and S' are locked in $G[S \cup S']$ but not in G. If there is no vertex $v \in V(G) \setminus (S \cup S')$ such that both $S + v$ and $S' + v$ are c-colorable in G, then* $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S') = |S \vartriangle S'| + 4$.

Proof. Let $u, v \in V(G) \setminus (S \cup S')$ be distinct vertices such that $S + u$ and $S' + v$ are c-colorable in G. Since $S + u$ and $S' + v$ are not locked in $G[(S + u) \cup (S' + v)]$, Theorem 3.2 (1) implies that $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S') \leq \mathsf{dist}_{\mathsf{TAR}(k)}(S + u, S' + v) + 2 = |(S + u) \vartriangle (S' + v)| + 2 = |S \vartriangle S'| + 4$.

Since S and S' are locked in $G[S \cup S']$ and there is no vertex $v \in V(G) \setminus (S \cup S')$ such that both $S + v$ and $S' + v$ are c-colorable in G, we have $\min_{T\colon S \leftrightarrow T,\ T'\colon S' \leftrightarrow T'} |T \vartriangle T'| = |S \vartriangle S'| + 2$. This implies that $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S') \geq |S \vartriangle S'| + 4$. ◀

3.2 Computing the distance in linear time

We here explain how to check which case of Theorem 3.2 applies to a given instance in linear time.

▶ **Lemma 3.8.** *Given an interval graph G and c-colorable sets S and S' in G, one can either find a vertex $v \notin S \cup S'$ such that $S + v$ and $S' + v$ are c-colorable or decide that no such vertex exists in linear time.*

Proof. Let $(M_1, \ldots, M_t)$ be a clique path of G. Recall that $M_1, \ldots, M_t$ are the maximal cliques of G. Thus, for every $T \subseteq V(G)$, the maximum clique size of $G[T]$ is equal to $\max_{1 \leq i \leq t} |T \cap M_i|$.

We compute $a_i^S = |S \cap M_i|$ for $1 \leq i \leq t$ as follows. Initialize all a_i^S to 0; for each $u \in S$, add 1 to all a_i^S with $l_u \leq i \leq r_u$. In the same way, we compute $a_i^{S'} = |S' \cap M_i|$ for $1 \leq i \leq t$. From the observation above, we can conclude that for each vertex $v \notin S \cup S'$, $S + v$ and $S' + v$ are c-colorable if and only if $a_i^S, a_i^{S'} < c$ for $l_v \leq i \leq r_v$.

The initialization and the test for all nonmembers of S can be done in time $O(\sum_{i=1}^{t} |M_i|)$. It suffices to show that $\sum_{i=1}^{t} |M_i| \leq \sum_{v \in V(G)} (\deg(v) + 1) = 2|E(G)| + |V(G)|$. Since $M_1 \not\subseteq M_2$, there is a vertex v with $l_v = r_v = 1$. Thus $|M_1| = \deg(v) + 1$. By induction on the number of vertices, our claim holds. ◀

By setting $S = S'$ in the lemma above, we have the following lemma.

▶ **Lemma 3.9.** *Given an interval graph G and a c-colorable set S in G, one can either find a vertex $v \notin S$ such that $S + v$ is c-colorable or decide that S is maximal in linear time.*

▶ **Corollary 3.10.** *Given an interval graph G and c-colorable sets S and S' in G, the distance* $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S')$ *can be computed in linear time.*

Proof. We first check whether S or S' is locked in G. If so, the distance is ∞. Otherwise, we check whether S and S' are locked in $G[S \cup S']$. If not both of them are locked in $G[S \cup S']$, then we can apply Theorem 3.2 (1) or (2) and determine the distance. If both S and S' are locked in $G[S \cup S']$, we find a vertex $v \notin S \cup S'$ such that both $S + v$ and $S' + v$ are c-colorable in G. Everything can be done in linear time by Lemmas 3.8 and 3.9. ◀

3.3 Finding a shortest reconfiguration sequence in linear time

Here we describe how we find an actual shortest reconfiguration sequence in linear time. To this end, we need to be careful about the representation of a reconfiguration sequence. If we always output the whole set, the total running time cannot be smaller than $k \cdot \mathsf{dist}_{\mathsf{TAR}(k)}(S, S')$. However, this product can be quadratic. To avoid this blow up, we output only the difference from the previous set. That is, if the current set is S and the next set is $S + v$ $(S - v)$, we output $+v$ $(-v$, resp.). We also fully use the reversible property of reconfiguration sequences and output them sometimes from left to right and sometimes from right to left. For example, we may output a reconfiguration sequence $\langle S_0, \dots, S_5 \rangle$ as first $S_0 \leftrightarrow S_1 \leftrightarrow S_2$, next $S_5 \leftrightarrow S_4 \leftrightarrow S_3$, then $S_2 \leftrightarrow S_3$. It is straightforward to output the sequence from left to right by using a linear-size buffer.

▶ **Theorem 3.11.** *Given an interval graph G and c-colorable sets S and S' in G, a $\mathsf{TAR}(k)$-sequence of length $\mathsf{dist}_{\mathsf{TAR}(k)}(S, S')$ can be computed in linear time.*

Proof. We first test which case of Theorem 3.2 applies to the given instance. This can be done in linear time as shown in the proof of Corollary 3.10. We reduce Cases (2) and (3) to Case (1). The reductions below can be done in linear time by using Lemmas 3.8 and 3.9.

Assume first that Case (2) applies; that is, S is locked but S' is not in $G[S \cup S']$. We find a vertex $v \in V(G) \setminus (S \cup S')$ such that $S + v$ is a c-colorable set of G. We then add v to S. As we saw in the proof of Lemma 3.5, this is a valid step in a shortest reconfiguration sequence. Furthermore, after this step, S and S' are not locked in $G[S \cup S']$.

Next assume that Case (3) applies; that is, both S and S' are locked in $G[S \cup S']$. We find vertices $u, v \notin S \cup S'$ such that $S + u$ and $S' + v$ are c-colorable in G. In Case (3a), we further ask that $u = v$. We then add u to S and v to S'. The proofs of Lemmas 3.6 and 3.7 imply that these are valid steps in a shortest reconfiguration sequence, and that S and S' are no longer locked in $G[S \cup S']$ after these steps.

We now handle Case (1), where S and S' are not locked in $G[S \cup S']$. Assume that $S \not\subseteq S'$ and $S' \not\subseteq S$ since otherwise finding a shortest sequence is trivial. We first compute two orderings of the vertices in $S \cup S'$: nondecreasing orderings of left-ends l_v and of right-ends r_v. Such orderings can be constructed in linear time from a clique path. We maintain information for each vertex v whether $v \in S \setminus S'$, $v \in S' \setminus S$, or $v \notin S \mathbin{\triangle} S'$. Using this information, we can also maintain vertices of the smallest left-end and of the smallest right-end in each of $S \setminus S'$ and $S' \setminus S$.

Let $v \in S \setminus S'$ and $w \in S' \setminus S$ be vertices with the smallest right-end in each set. By symmetry, assume that $r_w \le r_v$. Let $u \in S \setminus S'$ be a vertex that minimizes l_u. As shown in the proof of Lemma 3.4, $S \leftrightarrow (S - u) \leftrightarrow (S - u + w)$ under $\mathsf{TAR}(k)$ and $|S \mathbin{\triangle} S'| = |(S - u + w) \mathbin{\triangle} S'| + 2$. We output the two steps $S - u$ and $S - u + w$.

We then set $S := S - u + w$ and update the information as $u, w \notin S \mathbin{\triangle} S'$ anymore. We also have to maintain the vertices of the smallest left- and right-ends in each $S \setminus S'$ and $S' \setminus S$. Let $w' \in S' \setminus S$ be a vertex with the smallest right-end. The vertex w can be found by sweeping the nondecreasing ordering of the right-ends from the position of w to the right. The vertex $u' \in S \setminus S'$ with the smallest left-end can be found in an analogous way. Although a single update can take super constant steps, it sums up to a linear number of steps in total since it can be seen as a single left-to-right scan of each nondecreasing ordering. Therefore, the total running time is linear. ◀

4 Split graphs

For split graphs, we consider two cases. In the first case, we assume that c is a fixed constant, and show that the problem under TAR (and TJ) can be solved in $O(n^{c+1})$ time. The second case is the general problem having c as a part of input. We show that in this case the problem is PSPACE-complete under all reconfiguration rules.

4.1 Polynomial-time algorithm for fixed c

Let $G = (K, I; E)$ be a split graph, where K is a clique and I is an independent set. For $C \subseteq K$ with $|C| \le c$, we define T_C as follows:

$$T_C = \begin{cases} C \cup I & \text{if } |C| < c, \\ C \cup I \setminus \{u \in I : C \subseteq N_G(u)\} & \text{if } |C| = c. \end{cases}$$

We can see that T_C is c-colorable for every $C \subseteq K$ with $|C| \le c$ as follows. Every clique $K' \subseteq T_C$ includes at most $|C| \le c$ vertices in C and at most one vertex in I. Since a vertex in $T_C \cap I$ has fewer than c neighbors in C, the maximum clique size of $G[T_C]$ is at most c.

▶ **Lemma 4.1.** *If S is a c-colorable set of G with $|S| \ge k$, then $S \leftrightsquigarrow T_{S \cap K}$ under* TAR(k).

Proof. Note that $T_{S\cap K}$ is c-colorable since S is c-colorable and thus $|S \cap K| \le c$. We now show that $S \subseteq T_{S\cap K}$, which implies $S \leftrightsquigarrow T_{S\cap K}$.

If $|S \cap K| < c$, then $T_{S\cap K} = (S \cap K) \cup I$, and thus $S \subseteq T_{S\cap K}$. If $|S \cap K| = c$, then $S \cap \{u \in I : (S \cap K) \subseteq N(u)\} = \emptyset$ since S is c-colorable. Thus it holds that $S \subseteq (S \cap K) \cup (I \setminus \{u \in I : (S \cap K) \subseteq N(u)\}) = T_{S\cap K}$. ◀

By the reversibility of reconfiguration sequences, we can reduce the problem as follows.

▶ **Corollary 4.2.** *If S and S' are c-colorable sets of G with $|S| \ge k$ and $|S'| \ge k$, then $S \leftrightsquigarrow S'$ under* TAR(k) *if and only if $T_{S\cap K} \leftrightsquigarrow T_{S'\cap K}$ under* TAR(k).

Now we state the crucial lemma for solving the reduced problem.

▶ **Lemma 4.3.** *Let $C \subseteq K$ and $v \in K \setminus C$. If T_C and T_{C+v} are c-colorable sets of size at least k, then $T_C \leftrightsquigarrow T_{C+v}$ under* TAR(k) *if and only if $|T_{C+v}| \ge k + 1$.*

Proof. To prove the if part, assume that $|T_{C+v}| \ge k + 1$. Then, $T_{C+v} \leftrightarrow T_{C+v} - v$. Since $(T_{C+v} - v) \cap K = C$, it holds that $T_{C+v} - v \leftrightsquigarrow T_C$ by Lemma 4.1. Thus we have $T_{C+v} \leftrightsquigarrow T_C$.

To prove the only-if part, assume that $|T_{C+v}| = k$. If $|C + v| = c$, then T_{C+v} is a maximal c-colorable set and no other c-colorable set of size at least k can be reached from T_{C+v}. Assume that $|C + v| < c$, and hence $|C| < c$. Then, $T_{C+v} = (C + v) \cup I$ and $T_C = C \cup I$. Therefore, we have $|T_C| = |T_{C+v} - v| = k - 1$, a contradiction. ◀

Combining the arguments in this subsection, we are now ready to present a polynomial-time algorithm.

▶ **Theorem 4.4.** *Given an n-vertex split graph $G = (K, I; E)$ and c-colorable sets S and S' of size at least k in G, it can be decided whether $S \leftrightsquigarrow S'$ under* TAR(k) *in time $O(n^{c+1})$.*

Proof. We construct a graph $H = (\mathcal{K}, \mathcal{E})$ from $G = (K, I; E)$ as follows:

$$\mathcal{K} = \{C \subseteq K : |C| \le c \text{ and } |T_C| \ge k\},$$
$$\mathcal{E} = \{\{C, C + v\} : C, C + v \in \mathcal{K} \text{ and } |T_{C+v}| \ge k + 1\}.$$

For each $C \subseteq K$ with $|C| < c$, the size $|T_C| = |C| + |I|$ can be computed in constant time (assuming that we know the size $|I|$ in advance). If $|C| = c$, then we need to compute the size of $\{u \in I : C \subseteq N(u)\}$. This can be done in time $O(n)$ for each C. In H, each $C \in \mathcal{K}$ is adjacent to at most $|C|$ subsets of C: if $|T_C| > k$, then C is adjacent to all $C - v$ with $v \in C$; otherwise, C has no edge to its subsets. This can be computed in time $O(n)$ for each C. In total, the graph H with $O(n^c)$ vertices and $O(n^c)$ edges can be constructed in time $O(n^{c+1})$. For $C, C' \in \mathcal{K}$, one can decide whether H has a C–C' path in time $O(n^c)$.

Let $C := S \cap K$ and $C' := S' \cap K$. Now, by Corollary 4.2, it suffices to show that $T_C \leftrightsquigarrow T_{C'}$ if and only if there is a path between C and C' in H.

Assume that $T_C \leftrightsquigarrow T_{C'}$. Let $\langle S_1 = T_C, S_2, \ldots, S_p = T_{C'} \rangle$ be a reconfiguration sequence from T_C to $T_{C'}$, and let $C_i = S_i \cap K$ for $1 \le i \le p$. Observe that $|C_i \vartriangle C_{i+1}| \le 1$ for each $1 \le i < p$. If $C_i \ne C_{i+1}$, Corollary 4.2 and Lemma 4.3 imply that $\{C_i, C_{i+1}\} \in \mathcal{E}$. Since $C_1 = C$ and $C_p = C'$, we can conclude that H has a C–C' path.

Next assume that there is a path between C and C' in H. Let $(C_1 = C, C_2, \ldots, C_q = C')$ be such a path. Lemma 4.3 and the definition of H together imply that $T_{C_i} \leftrightsquigarrow T_{C_{i+1}}$ for each $1 \le i < q$. Since $C_1 = C$ and $C_q = C'$, we have $T_C \leftrightsquigarrow T_{C'}$. ◀

4.2 PSPACE-completeness when c is a part of input

Here we show the PSPACE-completeness when c is a part of input. (The proof if omitted in this version.)

▶ **Theorem 4.5.** *Given a split graph and c-colorable sets S and S' of size k in the graph, it is PSPACE-complete to decide whether $S \leftrightsquigarrow S'$ under any of* TS, TJ, *and* TAR$(k-1)$.

5 Concluding remarks

We show that Colorable Set Reconfiguration under TAR/TJ is linear-time solvable on interval graphs. Our results give a sharp contrast of the computational complexity with respect to graph classes, while some cases are left unanswered. One of the main unsettled cases is $\mathrm{CSR}_{\mathsf{TAR}}$ with fixed $c > 1$ for chordal graphs (see Table 1). In particular, what is the complexity of $\mathrm{CSR}_{\mathsf{TAR}}$ with $c = 2$ for chordal graphs? This problem is equivalent to the reconfiguration of feedback vertex sets under TAR on chordal graphs. It would be also interesting to study the shortest variant on split graphs with a constant c.

Our positive results for $\mathrm{CSR}_{\mathsf{TAR}}$ on interval graphs and split graphs (Theorems 3.11 and 4.4) do not imply analogous results for $\mathrm{CSR}_{\mathsf{TS}}$. The complexity of $\mathrm{CSR}_{\mathsf{TS}}$ is not settled for these graph classes even with a fixed constant c. It was only recently shown that if $c = 1$, then $\mathrm{CSR}_{\mathsf{TS}}$ can be solved in polynomial time for interval graphs [4]. For $c \ge 2$, $\mathrm{CSR}_{\mathsf{TS}}$ on interval graphs is left unsettled. For split graphs, although co-NP-hardness of a related problem is known [4], $\mathrm{CSR}_{\mathsf{TS}}$ is not solved for all $c \ge 1$.

References

1 Louigi Addario-Berry, W. Sean Kennedy, Andrew D. King, Zhentao Li, and Bruce A. Reed. Finding a maximum-weight induced k-partite subgraph of an i-triangulated graph. *Discrete Applied Mathematics*, 158(7):765–770, 2010. doi:10.1016/j.dam.2008.08.020.

2 Eyjólfur Ingi Ásgeirsson, Magnús M. Halldórsson, and Tigran Tonoyan. Universal framework for wireless scheduling problems. In Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl, editors, *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10-14, 2017, Warsaw, Poland*, volume 80

of *LIPIcs*, pages 129:1–129:15. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017. doi:10.4230/LIPIcs.ICALP.2017.129.

3 Matthias Bentert, René van Bevern, and Rolf Niedermeier. (Wireless) Scheduling, graph classes, and c-colorable subgraphs. *CoRR*, abs/1712.06481, 2017. arXiv:1712.06481.

4 Marthe Bonamy and Nicolas Bousquet. Token sliding on chordal graphs. In Hans L. Bodlaender and Gerhard J. Woeginger, editors, *Graph-Theoretic Concepts in Computer Science - 43rd International Workshop, WG 2017, Eindhoven, The Netherlands, June 21-23, 2017, Revised Selected Papers*, volume 10520 of *Lecture Notes in Computer Science*, pages 127–139. Springer, 2017. doi:10.1007/978-3-319-68705-6_10.

5 Paul S. Bonsma, Marcin Kaminski, and Marcin Wrochna. Reconfiguring independent sets in claw-free graphs. In R. Ravi and Inge Li Gørtz, editors, *Algorithm Theory - SWAT 2014 - 14th Scandinavian Symposium and Workshops, Copenhagen, Denmark, July 2-4, 2014. Proceedings*, volume 8503 of *Lecture Notes in Computer Science*, pages 86–97. Springer, 2014. doi:10.1007/978-3-319-08404-6_8.

6 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.

7 Samuel Fiorini, R. Krithika, N. S. Narayanaswamy, and Venkatesh Raman. LP approaches to improved approximation for clique transversal in perfect graphs. In Andreas S. Schulz and Dorothea Wagner, editors, *Algorithms - ESA 2014 - 22th Annual European Symposium, Wroclaw, Poland, September 8-10, 2014. Proceedings*, volume 8737 of *Lecture Notes in Computer Science*, pages 430–442. Springer, 2014. doi:10.1007/978-3-662-44777-2_36.

8 Delbert R. Fulkerson and Oliver A. Gross. Incidence matrices and interval graphs. *Pac. J. Math.*, 15(3):835–855, 1965. doi:10.2140/pjm.1965.15.835.

9 M. R. Garey, David S. Johnson, and Larry J. Stockmeyer. Some simplified np-complete graph problems. *Theor. Comput. Sci.*, 1(3):237–267, 1976. doi:10.1016/0304-3975(76)90059-1.

10 Paul C. Gilmore and Alan J. Hoffman. A characterization of comparability graphs and of interval graphs. *Canad. J. Math.*, 16:539–548, 1964. doi:10.4153/CJM-1964-055-5.

11 Martin Charles Golumbic. *Algorithmic Graph Theory and Perfect Graphs*, volume 57 of *Annals of Discrete Mathematics*. North Holland, second edition, 2004.

12 Martin Grötschel, László Lovász, and Alexander Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer-Verlag, 1988.

13 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.

14 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.

15 Marcin Kaminski, Paul Medvedev, and Martin Milanic. Complexity of independent set reconfigurability problems. *Theor. Comput. Sci.*, 439:9–15, 2012. doi:10.1016/j.tcs.2012.03.004.

16 R. Krithika and N. S. Narayanaswamy. Parameterized algorithms for (r, l)-partization. *J. Graph Algorithms Appl.*, 17(2):129–146, 2013. doi:10.7155/jgaa.00288.

17 Daniel Lokshtanov and Amer E. Mouawad. The complexity of independent set reconfiguration on bipartite graphs. In Artur Czumaj, editor, *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018*, pages 185–195. SIAM, 2018. doi:10.1137/1.9781611975031.13.

18 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.

19 Naomi Nishimura. Introduction to reconfiguration. *Preprints*, 2017. 2017090055. doi:10.20944/preprints201709.0055.v1.

20 Jeremy P. Spinrad. *Efficient Graph Representations.* Fields Institute monographs. American Mathematical Society, 2003.

21 Ryuhei Uehara and Yushi Uno. On computing longest paths in small graph classes. *Int. J. Found. Comput. Sci.*, 18(5):911–930, 2007. doi:10.1142/S0129054107005054.

22 Jan van den Heuvel. The complexity of change. In Simon R. Blackburn, Stefanie Gerke, and Mark Wildon, editors, *Surveys in Combinatorics 2013*, volume 409 of *London Mathematical Society Lecture Note Series*, pages 127–160. Cambridge University Press, 2013. doi:10.1017/CBO9781139506748.005.