# On the Parameterized Complexity of Reconfiguration of Connected Dominating Sets

## Daniel Lokshtanov
University of California Santa Barbara, CA, USA
daniello@ucsb.edu

## Amer E. Mouawad
Department of Computer Science, American University of Beirut, Lebanon
aa368@aub.edu.lb

## Fahad Panolan
Department of Computer Science and Engineering, IIT Hyderabad, India
fahad@cse.iith.ac.in

## Sebastian Siebertz
University of Bremen, Germany
siebertz@uni-bremen.de

──── **Abstract** ────

In a reconfiguration version of a decision problem $\mathcal{Q}$ the input is an instance of $\mathcal{Q}$ and two feasible solutions $S$ and $T$. The objective is to determine whether there exists a step-by-step transformation between $S$ and $T$ such that all intermediate steps also constitute feasible solutions. In this work, we study the parameterized complexity of the CONNECTED DOMINATING SET RECONFIGURATION problem (CDS-R). It was shown in previous work that the DOMINATING SET RECONFIGURATION problem (DS-R) parameterized by $k$, the maximum allowed size of a dominating set in a reconfiguration sequence, is fixed-parameter tractable on all graphs that exclude a biclique $K_{d,d}$ as a subgraph, for some constant $d \geq 1$. We show that the additional connectivity constraint makes the problem much harder, namely, that CDS-R is W[1]-hard parameterized by $k + \ell$, the maximum allowed size of a dominating set plus the length of the reconfiguration sequence, already on 5-degenerate graphs. On the positive side, we show that CDS-R parameterized by $k$ is fixed-parameter tractable, and in fact admits a polynomial kernel on planar graphs.

## 1 Introduction

In a decision problem $\mathcal{Q}$, we are usually asked to determine the existence of a feasible solution for an instance $\mathcal{I}$ of $\mathcal{Q}$. In a *reconfiguration version* of $\mathcal{Q}$, we are instead given a source feasible solution $S$ and a target feasible solution $T$ and we are asked to determine whether it is possible to transform $S$ into $T$ by a sequence of step-by-step transformations such that after each intermediate step we also maintain feasible solutions. Formally, we consider a graph, called the *reconfiguration graph*, that has one vertex for each feasible solution and where two vertices are connected by an edge if we allow the transformation between the two
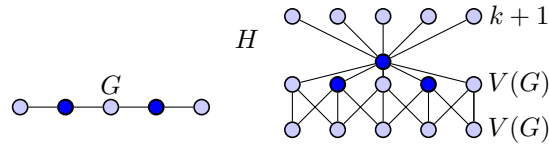
■ **Figure 1** A graph $G$ with a minimum dominating set of size $k = 2$ marked in dark blue and the graph $H$ obtained in the standard reduction from DOMINATING SET to CONNECTED DOMINATING SET. $G$ has a dominating set of size $k$ if and only if $H$ has a connected dominating set of size $k + 1$. If $p$ is equal to the pathwidth of $G$ then the pathwidth of $H$ is bounded by $2p + 1$.

corresponding solutions. We are then asked to determine whether $S$ and $T$ are connected in the reconfiguration graph, or even to compute a shortest path between them. Historically, the study of reconfiguration questions predates the field of computer science, as many classic one-player games can be formulated as such reachability questions [18, 20], e.g., the 15-puzzle and Rubik's cube. More recently, reconfiguration problems have emerged from computational problems in different areas such as graph theory [1, 16, 17], constraint satisfaction [12, 25] and computational geometry [5, 19, 23], and even quantum complexity theory [11]. Reconfiguration problems have been receiving considerable attention in recent literature, we refer the reader to [24, 28, 32] for an extensive overview.

In this work, we consider the CONNECTED DOMINATING SET RECONFIGURATION problem (CDS-R) in undirected graphs. A *dominating set* in a graph $G$ is a set $D \subseteq V(G)$ such that every vertex of $G$ lies either in $D$ or is adjacent to a vertex in $D$. A dominating set $D$ is a *connected dominating set* if the graph induced by $D$ is connected. The DOMINATING SET problem and its connected variant have many applications, including the modeling of facility location problems, routing problems, and many more.

We study CDS-R under the *Token Addition/Removal* model (TAR model). Suppose we are given a connected dominating set $D$ of a graph $G$, and imagine that a token is placed on each vertex in $D$. The TAR rule allows either the addition or removal of a single token at a time from $D$, if this results in a connected dominating set of size at most a given bound $k \geq 1$. A sequence $D_1, \ldots, D_\ell$ of connected dominating sets of a graph $G$ is called a *reconfiguration sequence* between $D_1$ and $D_\ell$ under TAR if the change from $D_i$ to $D_{i+1}$ respects the TAR rule, for $1 \leq i < \ell$. The *length* of the reconfiguration sequence is $\ell - 1$.

The (CONNECTED) DOMINATING SET RECONFIGURATION problem for TAR gets as input a graph $G$, two (connected) dominating sets $S$ and $T$ and an integer $k \geq 1$, and the task is to decide whether there exists a reconfiguration sequence between $S$ and $T$ under TAR using at most $k$ tokens.

Structural properties of the reconfiguration graph for $k$-dominating sets were studied in [14, 31]. The DOMINATING SET RECONFIGURATION problem was shown to be PSPACE-complete in [15], even on split graphs, bipartite graphs, planar graphs and graphs of bounded bandwidth. Both pathwidth and treewidth of a graph are bounded by its bandwidth, hence the DOMINATING SET RECONFIGURATION problem is PSPACE-complete on graphs of bounded pathwidth and treewidth. These hardness results motivated the study of the parameterized complexity of the problem. It was shown in [26] that the DOMINATING SET RECONFIGURATION problem is W[2]-hard when parameterized by $k + \ell$, where $k$ is the bound on the number of tokens and $\ell$ is the length of the reconfiguration sequence. However, the problem becomes fixed-parameter tractable on graphs that exclude a fixed complete bipartite graph $K_{d,d}$ as a subgraph, as shown in [22]. Such so-called *biclique-free* classes are very general sparse graph classes, including in particular the planar graphs, which are $K_{3,3}$-free.

In this work we study the complexity of CDS-R. The standard reduction from DOMINATING SET to CONNECTED DOMINATING SET shows that CDS-R is also PSPACE-complete, even on graphs of bounded pathwidth (Figure 1). We hence turn our attention to the parameterized complexity of the problem. We first show that the additional connectivity constraint makes the problem much harder, namely, that CDS-R parameterized by $k + \ell$ is W[1]-hard already on 5-degenerate graphs. As 5-degenerate graphs exclude the biclique $K_{6,6}$ as a subgraph, DOMINATING SET RECONFIGURATION is fixed-parameter tractable on much more general graph classes than its connected variant. To prove hardness we first introduce an auxiliary problem that we believe is of independent interest. In the COLORED CONNECTED SUBGRAPH problem we are given a graph $G$, an integer $k$, and a (not necessarily proper) coloring $c\colon V(G) \to C$, for some color set $C$ with $|C| \le k$. The question is whether $G$ contains a vertex subset $H$ on at most $k$ vertices such that $G[H]$ is connected and $H$ contains at least one vertex of every color in $C$ (i.e., $c(V(H)) = C$). The reconfiguration variant COLORED CONNECTED SUBGRAPH RECONFIGURATION (CCS-R) is defined as expected. We first prove that CCS-R reduces to CDS-R by a parameter preserving reduction (where $k + \ell$ is the parameter) and the degeneracy of the reduced graph is at most the degeneracy of the input graph plus one. We then prove that the known W[1]-hard problem MULTICOLORED CLIQUE (see [3] for definitions) reduces to CCS-R on 4-degenerate graphs. The last reduction has the additional property that for an input $(G, c, k)$ of MULTICOLORED CLIQUE the resulting instance of CCS-R admits either a reconfiguration sequence of length $\mathcal{O}(k^3)$, or no reconfiguration sequence at all. Hence, we derive that both CDS-R and CCS-R are W[1]-hard parameterized by $k + \ell$ on 5-degenerate and 4-degenerate graphs, respectively.

The existence of a reconfiguration sequence of length at most $\ell$ with connected dominating sets of size at most $k$ can be expressed by a first-order formula of length depending only on $k$ and $\ell$. It follows from [13] that the problem is fixed-parameter tractable parameterized by $k + \ell$ on every nowhere dense graph class and the same is implied by [2] for every class of bounded cliquewidth. Nowhere dense graph classes are very general classes of uniformly sparse graphs, in particular the class of planar graphs is nowhere dense. Nowhere dense classes are themselves biclique-free, but are not necessarily degenerate. Hence, our hardness result on degenerate graphs essentially settles the question of fixed-parameter tractability for the parameter $k + \ell$ on sparse graph classes. It remains an interesting open problem to find dense graph classes beyond classes of bounded cliquewidth on which the problem is fixed-parameter tractable.

We then turn our attention to the smaller parameter $k$ alone. We show that CDS-R parameterized by $k$ is fixed-parameter tractable on the class of planar graphs. Our approach is as follows. We first compute a small *domination core* for $G$, a set of vertices that captures exactly the domination properties of $G$ for dominating sets of sizes not larger than $k$. The notion of a domination core was introduced in the study of the DISTANCE-$r$ DOMINATING SET problem on nowhere dense graph classes [4]. While the classification of interactions with the domination core would suffice to solve DOMINATING SET RECONFIGURATION on nowhere dense classes, additional difficulties arise for the connected variant. In a second step we use planarity to identify large subgraphs that have very simple interactions with the domination core and prove that they can be replaced by constant size gadgets such that the reconfiguration properties of $G$ are preserved.

Observe that CDS-R parameterized by $k$ is trivially fixed-parameter tractable on every class of bounded degree. The existence of a connected dominating set of size $k$ implies that the diameter of $G$ is bounded by $k + 2$, which in every bounded degree class implies a bound on the size of the graph depending only on the degree and $k$. We conjecture that CDS-R is fixed-parameter tractable parameterized by $k$ on every nowhere dense graph class. However, resolving this conjecture remains open for future work (see Figure 2).
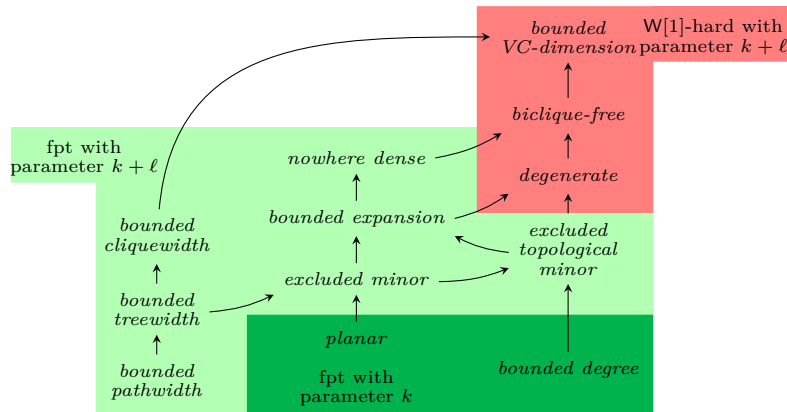
**Figure 2** The map of tractability for CONNECTED DOMINATING SET RECONFIGURATION. The classes colored in dark green admit an fpt algorithm with parameter $k$, the classes colored in light green admit an FPT algorithm with parameter $k + \ell$. On the classes colored in red the problem is W[1]-hard with respect to the parameter $k + \ell$.

The rest of the paper is organized as follows. We give background on graph theory and fix our notation in Section 2. We show hardness of CDS-R on degenerate graphs in Section 3 and show how to handle the planar case in Section 4. Due to space constraints proofs of results marked with a $\star$ are deferred to the full version of the paper.

## 2 Preliminaries

We denote the set of natural numbers by $\mathbb{N}$. For $n \in \mathbb{N}$, we let $[n] = \{1, 2, \ldots, n\}$. We assume that each graph $G$ is finite, simple, and undirected. We let $V(G)$ and $E(G)$ denote the vertex set and edge set of $G$, respectively. An edge between two vertices $u$ and $v$ in a graph is denoted by $\{u, v\}$ or $uv$. The *open neighborhood* of a vertex $v$ is denoted by $N_G(v) = \{u \mid \{u, v\} \in E(G)\}$ and the *closed neighborhood* by $N_G[v] = N_G(v) \cup \{v\}$. The degree of a vertex $v$, denoted $d_G(v)$, is $|N_G(v)|$. For a set of vertices $S \subseteq V(G)$, we define $N_G(S) = \{v \notin S \mid \{u, v\} \in E(G), u \in S\}$ and $N_G[S] = N_G(S) \cup S$. The subgraph of $G$ induced by $S$ is denoted by $G[S]$, where $G[S]$ has vertex set $S$ and edge set $\{\{u, v\} \in E(G) \mid u, v \in S\}$. We let $G - S = G[V(G) \setminus S]$. A graph $G$ is $d$-degenerate if every subgraph $H \subseteq G$ has a vertex of degree at most $d$. For a set $C$, we use $K[C]$ to denote the complete graph on vertex set $C$. For an integer $r \in \mathbb{N}$, an $r$-independent set in a graph $G$ is a subset $U \subseteq V(G)$ such that for any two distinct vertices $u, v \in U$, the distance between $u$ and $v$ in $G$ is more than $r$. An independent set in a graph is a 1-independent set. A subset of vertices $U$ in $G$ is called a separator in $G$ if $G - U$ is has more than one connected component. For $s, t \in V(G)$, we say $U$ is an $(s, t)$-separator in $G$ if there is no path from $s$ to $t$ in $G - U$.

## 3 Hardness on degenerate graphs

In this section we prove that CDS-R and CCS-R are W[1]-hard when parameterized by $k + \ell$ even on 5-degenerate and 4-degenerate graphs, respectively. Towards that, we first give a polynomial-time reduction from the W[1]-hard MULTICOLORED CLIQUE problem to CCS-R on 4-degenerate graphs with the property that for an input $(G, c, k)$ of MULTICOLORED CLIQUE the resulting instance of CCS-R admits either a reconfiguration sequence of length $\mathcal{O}(k^3)$ or no reconfiguration sequence at all. As a result, we conclude

that CCS-R is W[1]-hard when parameterized by $k + \ell$ on 4-degenerate graphs. Then, we give a parameter-preserving polynomial-time reduction from CCS-R to CDS-R. Let us first formally define the CCS problem.
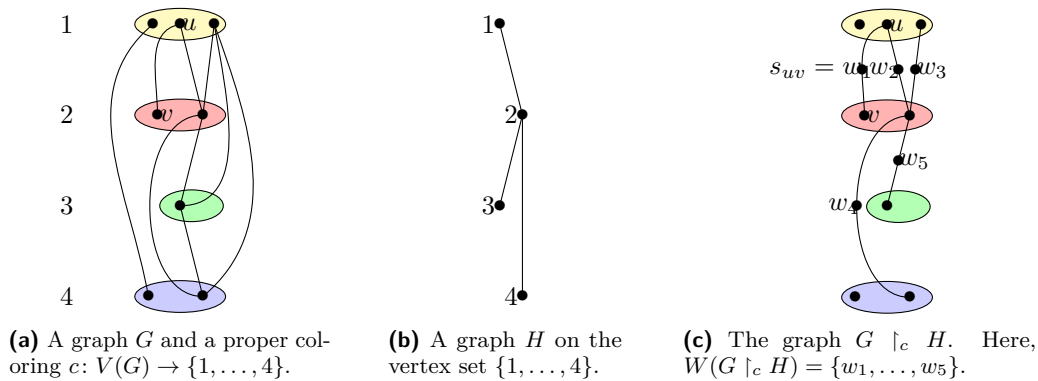
---

COLORED CONNECTED SUBGRAPH (CCS)                                                **Parameter:** $k$
**Input:** A graph $G$, a vertex-coloring $c: V(G) \to C$, and $k \in \mathbb{N}$ such that $|C| \leq k$
**Question:** Is there a vertex subset $S \subseteq V(G)$ of at most $k$ vertices with at least one vertex from every color class such that $G[S]$ is connected?

---

### Reduction from Multicolored Clique to CCS-R

We now present the reduction from MULTICOLORED CLIQUE to CCS-R, which we believe to be of independent interest. We can assume, without loss of generality, that for an input $(G, c, k)$ of MULTICOLORED CLIQUE, $G$ is connected and $c$ is a proper vertex-coloring, i.e., for any two distinct vertices $u, v \in V(G)$ with $c(u) = c(v)$ we have $\{u, v\} \notin E(G)$. Before we proceed let us define a graph operation.

▶ **Definition 3.1.** *Let $G$ be a graph and let $c: V(G) \to \{1, \ldots, k\}$ be a proper vertex coloring of $V(G)$. Let $H$ be a graph on the vertex set $\{1, \ldots, k\}$. We define the graph $G \upharpoonright_c H$ as follows. We remove all edges $\{u, v\} \in E(G)$ such that $c(u) = i$ and $c(v) = j$ and $\{i, j\} \notin E(H)$. We subdivide every remaining edge, i.e. for every remaining edge $\{u, v\}$ we introduce a new vertex $s_{uv}$, remove the edge $\{u, v\}$ and introduce instead the two edges $\{u, s_{uv}\}$ and $\{v, s_{uv}\}$. We write $W(G \upharpoonright_c H)$ for the set of all subdivision vertices $s_{uv}$ (see Figure 3).*

That is, to construct $G \upharpoonright_c H$, we first make a subgraph of $G$ by deleting the edges between different color classes if there are no edges between the "corresponding" vertices in $H$, and then subdivide the remaining edges. Let $(G, c, k)$ be the input instance of MULTICOLORED CLIQUE, where $G$ is a connected graph and $c$ is a proper $k$-vertex-coloring of $G$. We construct an instance $(H, \widehat{c}: V(H) \mapsto [k+1], Q_s, Q_t, 2k)$ of CCS-R ($Q_s$ and $Q_t$ are the source and target sets that we describe later). Note that the bound on the sizes of the solutions in the reconfiguration sequence is at most $2k$.



**(a)** A graph $G$ and a proper coloring $c: V(G) \to \{1, \ldots, 4\}$.

**(b)** A graph $H$ on the vertex set $\{1, \ldots, 4\}$.

**(c)** The graph $G \upharpoonright_c H$. Here, $W(G \upharpoonright_c H) = \{w_1, \ldots, w_5\}$.
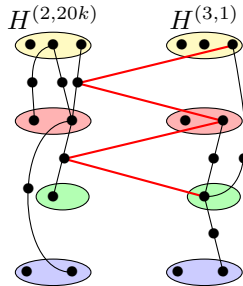
■ **Figure 3** Construction of $G \upharpoonright_c H$.

We first construct a routing gadget. For $1 \le i \le k$, let $T^i$ be the star with vertex set $\{1, \ldots, k\}$ having vertex $i$ as the center. For any $1 \le i \le k$ and $1 \le r \le 20k$, we let $H^{(i,r)}$ be a copy of the graph $G \upharpoonright_c T^i$. We let $c_{(i,r)}$ be the the partial vertex-coloring of $H^{(i,r)}$ that is naturally inherited from $G$. For an illustration, consider the input instance $(G, c, k)$ of MULTICOLORED CLIQUE depicted in Figure 3a. Then, $T^2$ is identical to the graph $H$ in Figure 3b and Figure 3c represents $H^{(2,r)} = G \upharpoonright_c T^2$, for any $1 \le r \le 20k$. Now, for $1 \le i \le k$ we define a graph $H^i$ as follows. We use $W(H^{(i,r)})$ to denote the set of subdivision vertices in $H^{(i,r)}$. For $1 \le r < 20k$ and all vertices $u, v$ in $V(H^{(i,r)}) \setminus W(H^{(i,r)})$, we connect the copy of the subdivision vertex $s_{uv}$ in $H^{(i,r)}$ (if it exists) with the copies of the vertices $u$ and $v$ in $H^{(i,r+1)}$ (see Figure 4 for an illustration of a portion of $H^1$). We use $W(H^i)$ to denote the set of subdivision vertices $\bigcup_{r \in [20k]} W(H^{(i,r)})$.
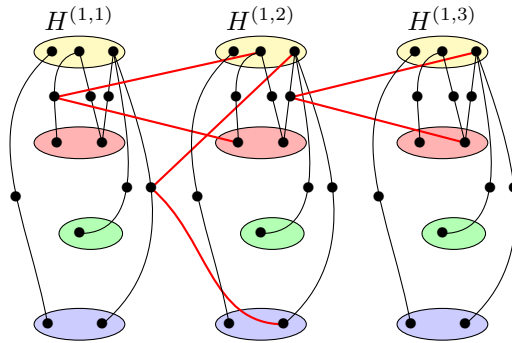


**Figure 4** Construction of $H^1$ from the instance $(G, c)$ depicted in Figure 3a. The red edges are some of the "crossing" edges but not all of them.

For each $1 \le i \le k$, we use $c_i$ to denote a coloring on $V(H^i)$ that is a union of $c_{(i,1)}, c_{(i,2)}, \ldots, c_{(i,20k)}$ and we color all the copies of the subdivision vertices using a new color $k + 1$. In other words, we know that for each $u \in V(H^i)$ we have $u \in V(H^{(i,r)})$, for some $r \in \{1, \ldots, 20k\}$. Hence, if $u \in V(H^{(i,r)}) \setminus W(H^{(i,r)})$ then we set $c_i(u) = c_{(i,r)}(u)$. For all $s_{uv} \in W(H^i)$, we set $c_i(s_{uv}) = k + 1$.

Now, define a graph $R$, which is super graph of $H^1 \cup \ldots \cup H^k$, as follows. For $1 \le i < k$ and all vertices $u$ and $v$, we connect the copy of the subdivision vertex $s_{uv}$ in $H^{(i,20k)}$ (if it exists) with the copies of the vertices $u$ and $v$ in $H^{(i+1,1)}$ (see Figure 5 for an illustration).



**Figure 5** Illustration of the subgraph of $R$ induced on $V(H^{(2,20k)}) \cup V(H^{3,1})$ constructed from the instance $(G, c, k)$ depicted in Figure 3a. The red edge are some of the "crossing edges".

We additionally introduce two subgraphs $H^0$ and $H^{k+1}$. The graph $H^0$ is obtained by subdividing each edge of a star on vertex set $\{v_1, \ldots, v_k\}$ centered at $v_1$. Here we use $w_2, \ldots, w_k$ to denote the subdivision vertices. Similarly, the graph $H^{k+1}$ is obtained by subdividing each

edge of star on $\{x_1, \ldots, x_k\}$ centered at $x_k$. Here $y_1, \ldots, y_{k-1}$ denote the subdivision vertices. Let $c_0$ and $c_{k+1}$ be the colorings on $\{v_1, \ldots, v_k, w_2, \ldots, w_k\}$ and $\{x_1, \ldots, x_k, y_1, \ldots, y_{k-1}\}$, respectively, defined as follows. For all $1 \leq i \leq k$, $c_0(v_i) = i$ and $c_{k+1}(x_i) = i$. For all $2 \leq i \leq k$, $c_0(w_i) = k + 1$ and for all $1 \leq i \leq k - 1$, $c_{k+1}(y_i) = k + 1$. Observe that we may interpret $H^0$ as $K[\{v_1, \ldots, v_k\}] \upharpoonright_{c_0} T^0$ and $H^{k+1}$ as $K[\{x_1, \ldots, x_k\}] \upharpoonright_{c_{k+1}} T^{k+1}$, where $T^0$ and $T^{k+1}$ are two trees on vertex set $\{1, \ldots, k\}$, with $E(T^0) = \{\{1, i\} : 2 \leq i \leq k\}$ and $E(T^{k+1}) = \{\{k, i\} : 1 \leq i \leq k - 1\}$.

Finally, for each $2 \leq i \leq k$, we connect the "subdivision vertex" $w_i$ (adjacent to $v_1$ and $v_i$) to all vertices $v \in V(H^{(1,1)})$ colored 1 or $i$, i.e., with $c_{(1,1)}(v) \in \{1, i\}$. For each subdivision vertex $s_{ab} \in W(H^{(k,20k)})$, we connect $s_{ab}$ to $x_k$ and $x_i$, where $k = c_k(a) = c_{(k,20k)}(a)$ and $i = c_k(b) = c_{(k,20k)}(b)$. Recall that $s_{ab}$ is adjacent to a vertex of color $k$ and a vertex of color $i$, for some $i < k$. This completes the construction of $H$ (see Figure 6). We define $\widehat{c} : V(H) \mapsto [k + 1]$ to be the union of $c_0, \ldots, c_{k+1}$. We define the starting configuration $Q_s$ as the set $\{v_1, \ldots, v_k, w_2, \ldots, w_k\}$ and the target configuration $Q_t$ as the set $\{x_1, \ldots, x_k, y_1, \ldots, y_{k-1}\}$.
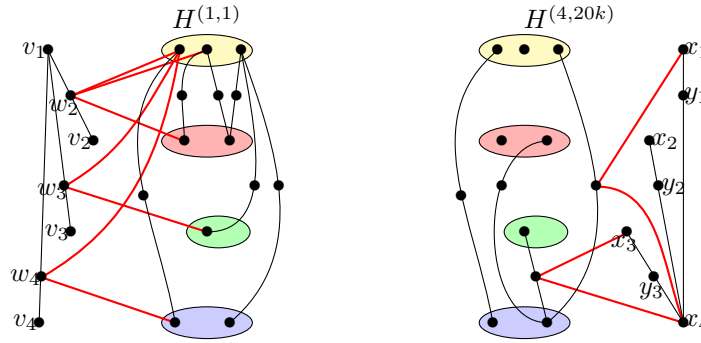


**Figure 6** Illustration of connection between $H^0$ and $R$, and $H^{k+1}$ and $R$ from the instance $(G, c, k)$ depicted in Figure 3a. The red edge are some of the "crossing edges" between $H^0$ and $H^1$, and $H^k$ and $H^{k+1}$.

▶ **Proposition 3.2.** *The sets $Q_s$ and $Q_t$ are solutions of size $2k - 1$ of the CCS instance $(H, \widehat{c}, 2k)$.*

We now consider the instance $(H, \widehat{c}, Q_s, Q_t, 2k)$ of the CCS-R problem. Before we analyze the reconfiguration properties of $H$, let us verify that $H$ is 4-degenerate.

▶ **Lemma 3.3.** *The graph $H$ is 4-degenerate.*

**Proof.** We iteratively remove minimum degree vertices and show that we can always remove a vertex of degree at most 4 in each step.

- Every subdivision vertex $w \in W(H^i)$ for $1 \leq i \leq k$ has degree at most 4; it has 4 neighbors in $V(H^i) \cup V(H^{i+1})$.
- After removal of all subdivision vertices the degree of the remaining vertices of each $H^i$ is at most one. That is, a vertex in $H^{(1,1)}$ may have a neighbor in $\{w_2, \ldots, w_k\}$.
- After the removal of $V(H^1) \cup \ldots V(H^k)$, the degree of all vertices except $v_1$ and $x_k$ is at most 2.
- Finally we remove $v_1$ and $x_k$.

This completes the proof. ◀

▶ **Lemma 3.4** (⋆). *If there exists a $k$-colored clique in $G$ then there is reconfiguration sequence of length $\mathcal{O}(k^3)$ from $Q_s$ to $Q_t$ in $(H, \widehat{c}, 2k)$.*

**Proof sketch.** Informally, in every step our solution consists of vertices corresponding to a clique and at most $k$ vertices from the subdivision vertices present in the clique. The reconfiguration sequence checks whether all the edges between the clique vertices are present.

We aim to shift the connected vertices of $Q_s$ through the subgraphs $H^1, \ldots, H^k$ (in that order) to maintain connectivity and eventually shift all the tokens to $Q_t$. For each $u_i \in V(G)$, $1 \leq j \leq k$ and $1 \leq r \leq 20k$, we use $u_i^{(j,r)}$ to denote the copy of $u_i$ in $H^{(j,r)}$. Let $C = \{u_1, \ldots, u_k\}$ be a $k$-colored clique in $G$ such that $c(u_i) = i$, for all $1 \leq i \leq k$. To prove the lemma, we need to define a reconfiguration sequence starting from $Q_s$ and ending at $Q_t$ such that the cardinality of any solution in the sequence is at most $2k$. First we define $k$ "colored" trees $\widehat{T}_1, \ldots, \widehat{T}_k$ each on $2k - 1$ vertices, and then prove that there are reconfiguration sequences from $Q_s$ to $V(\widehat{T}_1)$, $V(\widehat{T}_i)$ to $V(\widehat{T}_{i+1})$ for all $1 \leq i < k$, and $V(\widehat{T}_k)$ to $Q_t$.

We start by defining $\widehat{T}_1, \ldots, \widehat{T}_k$. For each $1 \leq i \leq k$, $C_i = \{u_1^{(i,1)}, \ldots, u_k^{(i,1)}\}$ and $S_i = \{z \in V(H^{(i,1)}): N_{H^{(i,1)}}(z) \cap C_i = 2\}$. That is, for each $1 \leq j \leq k$ and $j \neq i$, $s_{u_i^{(i,1)} u_j^{(i,1)}} \in S_i$ (the subdivision vertex on the edge $u_i^{(i,1)} u_j^{(i,1)}$ is in $S_i$), and $|S_i| = k - 1$. In other words, $C_i$ contains the copies of the vertices of the clique $C$ in $H^{(i,1)}$ and $S_i$ contains subdivision vertices corresponding to $k - 1$ edges in the clique incident on the $i$th colored vertex of the clique, such that $H[C_i \cup S_i]$ is a tree. Now, define $\widehat{T}_i = H[C_i \cup S_i]$. It is easy to verify that $\widehat{c}(C_i \cup S_i) = \{1, \ldots, k + 1\}$ and hence $C_i \cup S_i = V(\widehat{T}_i)$ is a solution to the CCS instance $(H, \widehat{c}, 2k)$. Let $T_s = H[Q_s]$ and $T_t = H[Q_t]$. Note that $T_s$ and $T_t$ are trees on $2k - 1$ vertices.

**Case 1: Reconfiguration from $Q_s$ to $V(\widehat{T}_1)$.** Informally, we move to $\widehat{T}_1$ by adding a token on $u_i^{(1,1)}$ and then removing tokens from $v_i$ for $i$ in the order $2, \ldots, k, 1$ (for a total of $2k$ token additions/removals). Finally, we move the tokens from $\{w_2, \ldots, w_{k-1}\}$ to $S_1$ in $2(k-1)$ steps. The length of the reconfiguration sequence is $2k + 2(k-1) = 4k - 2$.

**Case 2: Reconfiguration from $V(\widehat{T}_i)$ to $V(\widehat{T}_{i+1})$.** First we define $20k$ trees $P_1, \ldots P_{20k}$, each on $2k - 1$ vertices such that for all $1 \leq r \leq 20k$, (i) $V(P_r) \subseteq V(H^{(i,r)})$, and (ii) $\widehat{T}_i = P_1$. Then we give a reconfiguration sequence from $V(P_r)$ to $V(P_{r+1})$ for all $r \in [20k - 1]$ and a reconfiguration sequence from $V(P_{20k})$ to $V(\widehat{T}_{i+1})$.

Recall that $C = \{u_1, \ldots, u_k\}$ is a $k$-colored clique in $G$ such that $c(u_i) = i$ for all $1 \leq i \leq k$. For $1 \leq r \leq 20k$, let $C_i^r = \{u_1^{(i,r)}, \ldots, u_k^{(i,r)}\}$ and $S_i^r = \{z \in V(H^{(i,r)}): N_{H^{(i,r)}}(z) \cap C_i^r = 2\}$. That is, for each $1 \leq j \leq k$ and $j \neq i$, $s_{u_i^{(i,r)} u_j^{(i,r)}} \in S_i^r$ (i.e, the subdivision vertex on the edge $u_i^{(i,r)} u_j^{(i,r)}$ is in $S_i^r$) and $|S_i^r| = k - 1$. Let $P_r = H[C_i^r \cup S_i^r]$. Notice that for all $r \in [20k]$, $P_r$ is a tree on $2k - 1$ vertices. Moreover, for each $1 \leq r \leq 20k$, $V(P_r)$ is a solution to the CCS instance $(H, \widehat{c}, 2k)$. By arguments similar to those given for Case 1, one can prove that there is a reconfiguration sequence of length $4k - 2$ from $V(P_r)$ to $V(P_{r+1})$, for all $1 \leq r < 20k$.

For the reconfiguration sequence from $V(P_{20k})$ to $V(\widehat{T}_{i+1})$ we refer the reader to the complete proof in the full version of the paper.

**Case 3: Reconfiguration from $V(\widehat{T}_k)$ to $V(T_t)$.** The arguments for this case are similar to those given in Case 1, we therefore omit the details.                            ◀

▶ **Lemma 3.5** (⋆). *If there is a reconfiguration sequence from $Q_s$ to $Q_t$ then there is a $k$-colored clique in $G$.*

▶ **Theorem 3.6.** *CCS-R parameterized by $k + \ell$ is W[1]-hard on 4-degenerate graphs.*

**Reduction from CCS-R to CDS-R**

We give a polynomial-time parameter-preserving reduction from CCS-R to CDS-R that is fairly straightforward. Let $(G, c, Q_s, Q_t, k)$ be an instance of CCS-R. Let $c \colon V(G) \mapsto \{1, \ldots, k'\}$, where $k' \leq k$. We construct a graph $H$ as follows. For each $1 \leq i \leq k'$, we add a vertex $d_i$ and connect $d_i$ to all the vertices in $c^{-1}(i)$. Next, for each $1 \leq i \leq k'$, we add a pendant vertex $x_i$ (i.e., $\{d_i, x_i\}$ is an edge). Let $D = \{d_1, \ldots, d_{k'}\}$. We output $(H, Q_s \cup D, Q_t \cup D, k + k')$ as the new CDS-R instance.

▶ **Lemma 3.7.** *If $G$ is a $d$-degenerate graph then $H$ is a $(d + 1)$-degenerate graph.*

**Proof.** For each vertex $v \in V(G)$, $d_H(v) = d_G(v) + 1$. Thus, after removing $V(G)$ and $\{x_i \colon 1 \leq i \leq k'\}$, the remaining graph is edgeless. ◀

It is easy to verify that for any reconfiguration sequence $Q_s = R_1, \ldots, R_\ell = Q_t$ of the instance $(G, c, Q_s, Q_t, k)$ of CCS-R, $Q_s \cup D = R_1 \cup D, \ldots, R_\ell \cup D = Q_t \cup D$ is a reconfiguration sequence of the instance $(H, Q_s \cup D, Q_t \cup D, k + k')$ of CDS-R. Now we prove the reverse direction.

▶ **Lemma 3.8.** *If $(H, Q_s \cup D, Q_t \cup D, k + k')$ is a yes-instance then $(G, c, Q_s, Q_t, k)$ is a yes-instance.*

**Proof.** Notice that the set $D$ is contained in any connected dominating set of $H$. Moreover for any minimal connected dominating set $Z$ in $H$, $Z \cap \{x_i \colon 1 \leq i \leq k'\} = \emptyset$, $H[Z \setminus D]$ is connected, and $Z \setminus D$ contains a vertex from $c^{-1}(i)$ for all $1 \leq i \leq k'$ (recall that $G$ is a subgraph of $H$). Therefore, by deleting $D$ from each set in a reconfiguration sequence of $(H, Q_s \cup D, Q_t \cup D, k + k')$, we get a valid reconfiguration sequence of $(G, c, Q_s, Q_t, k)$. This completes the proof. ◀

Thus, by Theorem 3.6, we have the following theorem.

▶ **Theorem 3.9.** *CDS-R parameterized by $k + \ell$ is W[1]-hard on 5-degenerate graphs.*

## 4 Fixed-parameter tractability on planar graphs

This section is devoted to proving that CDS-R under TAR parameterized by $k$ is fixed-parameter tractable on planar graphs. In fact, we show that the problem admits a polynomial kernel. Recall that a kernel for a parameterized problem $\mathcal{Q}$ is a polynomial-time algorithm that computes for each instance $(I, k)$ of $\mathcal{Q}$ an equivalent instance $(I', k')$ with $|I'| + k' \leq f(k)$ for some computable function $f$. The kernel is polynomial if the function $f$ is polynomial. We prove that for every instance $(G, S, T, k)$ of CDS-R, with $G$ planar, we can compute in polynomial time an instance $(G', S, T, k)$ where $|V(G')| \leq h(k)$ for some polynomial $h$, $G'$ planar, and where there exists a reconfiguration sequence under TAR from $S$ to $T$ in $G$ (using at most $k$ tokens) if and only if such a sequence exists in $G'$.

Our approach is as follows. We first compute a small *domination core* for $G$, that is, a set of vertices that captures exactly the domination properties of $G$ for dominating sets of sizes not larger than $k$. While the classification of interactions with the domination core would suffice to solve DOMINATING SET RECONFIGURATION, additional difficulties arise for the connected variant. In a second step we use planarity to identify large subgraphs that have very simple interactions with the domination core and prove that they can be replaced by constant size gadgets such that the reconfiguration properties of $G$ are preserved.

## 4.1   Domination cores

▶ **Definition 4.1.** *Let $G$ be a graph and let $k \geq 1$ be an integer. A $k$-domination core is a subset $C \subseteq V(G)$ of vertices such that every set $X \subseteq V(G)$ of size at most $k$ that dominates $C$ also dominates $G$.*

It is not difficult to see that DOMINATING SET is fixed-parameter tractable on all graphs that admit a $k$-domination core of size at most $f(k)$ that is computable in time $g(k) \cdot n^c$, for any computable functions $f, g$ and constant $c$. This approach was first used (implicitly) in [4] to solve DISTANCE-$r$ DOMINATING SET on nowhere dense graph classes. In case $k$ is the size of a minimum (distance-$r$) dominating set, one can establish the existence of a linear size $k$-domination core on classes of bounded expansion [6] (including the class of planar graphs) and a polynomial size (in fact an almost linear size) $k$-domination core on nowhere dense graph classes [8, 21]. If $k$ is not minimum, there exist classes of bounded expansion such that a $k$-domination core must have at least quadratic size [7]. The most general graph classes that admit $k$-domination cores are given in [9]. Moreover, DOMINATING SET RECONFIGURATION and DISTANCE-$r$ DOMINATING SET RECONFIGURATION are fixed-parameter tractable on all graphs that admit small (distance-$r$) $k$-domination cores [22, 30].

▶ **Lemma 4.2.** *There exists a polynomial $h$ such that for all $k \geq 1$, every planar graph $G$ admits a polynomial-time computable $k$-domination core of size at most $h(k)$.*

The lemma is implied by Theorem 1.6 of [21] by the fact that planar graphs are nowhere dense. We want to stress again that the polynomial size of the $k$-domination core results from the fact that $k$ may not be the size of a minimum dominating set, if $k$ is minimum we can find a linear size core. Explicit bounds on the degree of the polynomial can be derived from [27, 29], but we refrain from doing so to not disturb the flow of ideas.

The following lemma is immediate from the definition of a $k$-domination core.

▶ **Lemma 4.3.** *If $C$ is a $k$-domination core and $D$ is a dominating set of size at most $k$ that contains a vertex set $W \subset D$ such that $N[D] \cap C = N[D \setminus W] \cap C = C$, then $D \setminus W$ is also a dominating set.*

▶ **Definition 4.4.** *Let $G$ be a graph and let $A \subseteq V(G)$. The projection of a vertex $v \in V(G) \setminus A$ into $A$ is the set $N(v) \cap A$. If two vertices $u, v$ have the same projection into $A$ we write $u \sim_A v$.*

Obviously, the relation $\sim_A$ is an equivalence relation. The following lemma is folklore, one possible reference is [10].

▶ **Lemma 4.5.** *Let $G$ be a planar graph and let $A \subseteq V(G)$. Then there exists a constant $c$ such that there are at most $c \cdot |A|$ different projections to $A$, that is, the equivalence relation $\sim_A$ has at most $c \cdot |A|$ equivalence classes.*

## 4.2   Reduction rules

Let $G$ be an embedded planar graph. We say that a vertex $v$ *touches* a face $f$ if $v$ is drawn inside $f$ or belongs to the boundary of $f$ or is adjacent to a vertex on the boundary of $f$. We fix two connected dominating sets $S$ and $T$ of size at most $k$. We will present a sequence of lemmas, each of which implies a polynomial-time computable reduction rule that allows us to transform $G$ to a planar graph $G'$ that inherits its embedding from $G$, with $S, T \subseteq V(G')$ and that has the same reconfiguration properties with respect to $S$ and $T$ as $G$. To not overload

notation, after stating a lemma with a reduction rule, we assume that the reduction rule is applied until this is no longer possible and call the resulting graph again $G$. We also assume that whenever one or more of our reduction rules are applicable, then they are applied in the order presented. We will guarantee that $S$ and $T$ will always be connected dominating sets of size at most $k$, hence, after each application of a reduction rule, we can recompute a $k$-domination core in polynomial time. This yields only polynomial overhead and allows us to assume that we always have marked a $k$-domination core $C$ of size at most $h(k)$ as described in Lemma 4.2. This allows us to state the lemmas as if $G$ and $C$ are fixed. Without loss of generality we assume that $C$ contains $S$ and $T$.

▶ **Definition 4.6.** *A set $W \subseteq V(G) \setminus C$ of vertices is* irrelevant *if there is a reconfiguration sequence from $S$ to $T$ in $G$ if and only if there is a reconfiguration sequence from $S$ to $T$ in $G - W$.*

▶ **Definition 4.7.** *Let $u, v \in V(G)$ be distinct vertices. We call the set $D(u, v) := (N(u) \cap N(v)) \cup \{u, v\}$ the* diamond *induced by $u$ and $v$. We call $|N(u) \cap N(v)|$ the* thickness *of $D(u, v)$.*

▶ **Lemma 4.8.** *If $G$ contains a diamond $D(u, v)$ of thickness greater than $3k$, then at least one of $u$ or $v$ must be occupied by a token in every reconfiguration sequence from $S$ to $T$.*

**Proof.** Assume $S = S_1, \ldots, S_t = T$ is a reconfiguration sequence from $S$ to $T$ and $u, v \notin S_i$ for some $1 \le i \le t$. Then every $s \in S_i$ can dominate at most 3 vertices of $N(u) \cap N(v)$:
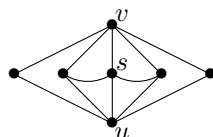


**Figure 7** A vertex $s \in S_i$ can dominate at most 3 vertices of $N(u) \cap N(v)$.

otherwise $u, v, s$ together with 3 vertices of $N(u) \cap N(v)$ different from $u, v$ and $s$ would form a complete bipartite graph $K_{3,3}$.                                                                             ◀

▶ **Lemma 4.9.** *If $G$ contains a diamond $D(u, v)$ of thickness greater than $3k$, then we can remove all internal edges in $D(u, v)$, i.e., edges with both endpoints in $N(u) \cap N(v)$.*

**Proof.** Assume $S = S_1, \ldots, S_t = T$ is a reconfiguration sequence from $S$ to $T$. According to Lemma 4.8, for each $1 \le i \le t$, $S_i \cap \{u, v\} \ne \emptyset$. Hence all vertices of $N(u) \cap N(v)$ are always dominated by at least one of $u$ or $v$, say by $u$. Moreover, having tokens on more than one vertex of $N(u) \cap N(v)$ will never create connectivity via internal edges that is not already there via edges incident on $u$. In other words, for any connected dominating set $S$ of $G$, if an edge $yz$ is used for connectivity, where $y, z \in N(u) \cap N(v)$, then the edge can be replaced by the path $yuz$ or the path $yvz$ (depending on which of $u$ or $v$ is in $S$).                                        ◀

As described earlier, we now apply the reduction rule of Lemma 4.9 until this is no longer possible, and name the resulting graph again $G$. As we did not make use of the properties of a $k$-domination core in the lemma, it is sufficient to recompute a $k$-domination core $C$ after applying the reduction rule exhaustively. In the following it may be necessary to recompute it after each application of a reduction rule. We will not mention these steps explicitly in the following.

▶ **Lemma 4.10** (⋆). *If $G$ contains a diamond $D(u,v)$ of thickness greater than $4|C| + 3k + 1$ then $G$ contains an irrelevant vertex.*

We may in the following assume that $G$ does not contain diamonds of thickness greater than $4|C| + 3k + 1$.

▶ **Corollary 4.11.** *If a vertex $v \in V(G)$ has degree greater than $(4|C| + 3k + 1) \cdot k$, then the token on $v$ is never lifted throughout a reconfiguration sequence.*

**Proof.** Assume $S = S_1, \ldots, S_t = T$ is a reconfiguration sequence from $S$ to $T$ in $G$ and assume there is $S_i$ with $v \notin S_i$. The dominating set $S_i$ has at most $k$ vertices and must dominate $N(v)$. Hence, there must be one vertex $u \in S_i$ that dominates at least a $1/k$ fraction of $N(v)$, which is larger than $4|C| + 3k + 1$. Then there is a diamond $D(u,v)$ of thickness greater than $4|C| + 3k + 1$, which does not exist after application of the reduction rule of Lemma 4.10. ◀

According to Corollary 4.11, the only vertices that can have high degree after applying the reduction rules are vertices that are never lifted throughout a reconfiguration sequence. This gives rise to another reduction rule that is similar to the rule of Lemma 4.9.

▶ **Lemma 4.12.** *Assume $v$ is a vertex of degree greater than $(4|C| + 3k + 1) \cdot k$. Then we may remove all edges with both endpoints in $N(v)$.*

**Proof.** Let $G'$ be the graph obtained from $G$ by removing all edges with both endpoints in $N(v)$. We claim that reconfiguration between $S$ and $T$ is possible in $G$ if and only if it is possible in $G'$. The fact that $S$ and $T$ are in fact connected dominating sets in $G'$ is implied by the argument below.

Assume $S = S_1, \ldots, S_t = T$ is a reconfiguration sequence from $S$ to $T$ in $G$. We claim that the same sequence is a reconfiguration sequence in $G'$. According to Corollary 4.11, $v \in S_i$ for all $1 \le i \le t$. This implies that $S_i$ is connected in $G'$ for all $1 \le i \le t$, as all $x, y \in S_i$ that are no longer connected by an edge in $G'$ but were connected in $G$ are connected via a path of length 2 using the vertex $v$. It is also easy to see that $S_i$ is a dominating set in $G'$, as all vertices that are no longer dominated by $s \in S_i$ in $G$ are still dominated by $v$. Observe that this in particular implies that $S$ and $T$ are connected dominating sets in $G'$. Vice versa, if $S = S_1, \ldots, S_t = T$ is a reconfiguration sequence from $S$ to $T$ in $G'$, this is trivially also a reconfiguration sequence in $G$. ◀

The following reduction rule is obvious.

▶ **Lemma 4.13.** *If a vertex $v$ has more than $k + 1$ pendant neighbours, i.e., neighbors of degree exactly one, then it suffices to retain exactly $k + 1$ of them in the graph.*

▶ **Lemma 4.14.** *There are at most $c|C| \cdot (4|C| + 3k + 1)$ vertices of $V(G) \setminus C$ that have 2 neighbours in $C$, where $c$ is the constant of Lemma 4.5.*

**Proof.** According to Lemma 4.5 there are at most $c|C|$ different projections to $C$. Each projection class that has at least 3 representatives has size at most 2, as otherwise we would find a $K_{3,3}$ as a subgraph, contradicting the planarity of $G$. Consider a class with a projection of size 2 into $C$. Denote these two vertices of $C$ by $u$ and $v$. If this class has more than $4|C| + 3k + 1$ representatives, then $D(u,v)$ is a diamond of thickness greater than $4|C| + 3k + 1$, which cannot exist after exhaustive application of the reduction rule of Lemma 4.10. ◀

We now come to the description of our final reduction rule. Let $D$ denote the set of vertices containing both $C$ and all vertices of $V(G) \setminus C$ having at least two neighbors in $C$. In other words, $V(G) \setminus D$ contains all those vertices in $V(G) \setminus C$ that have exactly one neighbor in $C$. According to Lemma 4.14 at most $c|C| \cdot (4|C| + 3k + 1)$ vertices have two neighbors in $C$, hence $|D| \leq c|C| \cdot (4|C| + 3k + 1) + |C| =: p$.

▶ **Lemma 4.15** (⋆). *Assume there are two vertices $u$ and $v$ with degree greater than $4p + (4|C| + 3k + 1) \cdot k + 1$. Let $\mathcal{P}$ be a maximum set of vertex-disjoint paths of length at least $2$ that run between $u$ and $v$ using only vertices in $V(G) \setminus D$. If $|\mathcal{P}| > 4p + (4|C| + 3k + 1) \cdot k + 1$, then there is $G'$ such that the instances $(G, S, T, k)$ and $(G', S, T, k)$ are equivalent, $G'$ is planar, and $|V(G')| < |V(G)|$.*

▶ **Theorem 4.16.** *CDS-R under TAR parameterized by $k$ admits a polynomial kernel on planar graphs.*

**Proof.** Our kernelization algorithm starts by computing (in polynomial time) a $k$-domination core $C$ of size at most $h(k)$ as described in Lemma 4.2. Without loss of generality we assume that $C$ contains $S$ and $T$. After each application of a reduction rule, we recompute the core, giving a polynomial blow-up of the running time. We are left to prove that each reduction rule can be implemented in polynomial time and that we end up with a polynomial number of vertices. It is clear that the reduction rules of Lemma 4.10, Lemma 4.12 and Lemma 4.13 can easily be implemented in polynomial time. The reduction rule of Lemma 4.15 is slightly more involved, however, we can use a standard maximum-flow algorithm to compute in polynomial time a maximum set of vertex-disjoint paths in a subgraph of $G$. It remains to bound the size of $G$. Recall that we call $D$ the set of all vertices $C$ and of all vertices of $V(G) \setminus C$ that have at least $2$ neighbors in $C$. It follows from Lemma 4.14 that $D$ has size at most $c|C| \cdot (4|C| + 3k + 1) + |C| =: p$, where $c$ is the constant of Lemma 4.5. We are left to bound the number of vertices in $V(G) \setminus C$ having exactly one neighbour in $C$ (recall that each vertex in $V(G) \setminus C$ has at least one neighbour in $S \cup T \subseteq C$).

Let $p' = (4p + (4|C| + 3k + 1) \cdot k + 1) \cdot (4|C| + 3k + 1) \cdot k + k + 1$, which is still a polynomial in $k$. Towards a contradiction, assume that there exists an equivalence class $Q$ in $\sim_C$ with a projection of size one containing more than $p'$ vertices. Let $u \in C$ denote the projection of the aforementioned class. Due to Lemma 4.13, we know that at most $k + 1$ of the vertices in $Q$ are pendant, i.e., adjacent to only $u$ in $G$. Since we cannot apply the reduction rule of Lemma 4.12 any more, we know that there are no edges with both endpoints in $Q$. Hence, all but $k + 1$ vertices of $Q$ must be adjacent to at least one other vertex in $V(G) \setminus C$. Let $R = N_G(Q) \setminus \{u\}$ denote this set of neighbours. No vertex in $R$ can be adjacent to more than $4|C| + 3k + 1$ vertices of $Q$, as we cannot apply the reduction rule of Lemma 4.10. The vertices of $R$ must be dominated by $S$, and cannot be dominated by $u$, as otherwise two neighbours of $u$ would be connected. Hence, there is $v \in S$ different from $u$ that dominates at least a $1/k$ fraction of $R$. This implies the existence of at least $4p + (4|C| + 3k + 1) \cdot k + 1$ vertex-disjoint paths of length at least $2$ that run between $u$ and $v$. But in this case, the reduction rule of Lemma 4.15 is applicable. Therefore, we conclude that $Q$ cannot exist, obtaining a bound on the size of all equivalence classes of $\sim_C$, as needed. ◀

───── **References** ─────

**1**    Luis Cereceda, Jan van den Heuvel, and Matthew Johnson. Connectedness of the graph of vertex-colourings. *Discrete Mathematics*, 308(56):913–919, 2008.

**2**    Bruno Courcelle, Johann A Makowsky, and Udi Rotics. Linear time solvable optimization problems on graphs of bounded clique-width. *Theory of Computing Systems*, 33(2):125–150, 2000.

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

**4**    Anuj Dawar and Stephan Kreutzer. Domination problems in nowhere-dense classes. In *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2009*, pages 157–168, 2009.

**5**    Erik D. Demaine and Joseph O'Rourke. *Geometric folding algorithms - linkages, origami, polyhedra.* Cambridge University Press, 2007.

**6**    Pål Grønås Drange, Markus Sortland Dregi, Fedor V. Fomin, Stephan Kreutzer, Daniel Lokshtanov, Marcin Pilipczuk, Michal Pilipczuk, Felix Reidl, Fernando Sánchez Villaamil, Saket Saurabh, Sebastian Siebertz, and Somnath Sikdar. Kernelization and sparseness: the case of dominating set. In *33rd Symposium on Theoretical Aspects of Computer Science, STACS 2016*, pages 31:1–31:14, 2016.

**7**    Eduard Eiben, Mithilesh Kumar, Amer E. Mouawad, Fahad Panolan, and Sebastian Siebertz. Lossy kernels for connected dominating set on sparse graphs. In *35th Symposium on Theoretical Aspects of Computer Science, STACS 2018*, pages 29:1–29:15, 2018.

**8**    Kord Eickmeyer, Archontia C. Giannopoulou, Stephan Kreutzer, O-joung Kwon, Michal Pilipczuk, Roman Rabinovich, and Sebastian Siebertz. Neighborhood complexity and kernelization for nowhere dense classes of graphs. In *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017*, pages 63:1–63:14, 2017.

**9**    Grzegorz Fabianski, Michal Pilipczuk, Sebastian Siebertz, and Szymon Toruńczyk. Progressive algorithms for domination and independence. In *36th International Symposium on Theoretical Aspects of Computer Science, STACS 2019*, pages 27:1–27:16, 2019.

**10**   Jakub Gajarský, Petr Hlinený, Jan Obdržálek, Sebastian Ordyniak, Felix Reidl, Peter Rossmanith, Fernando Sánchez Villaamil, and Somnath Sikdar. Kernelization using structural parameters on sparse graph classes. *J. Comput. Syst. Sci.*, 84:219–242, 2017.

**11**   Sevag Gharibian and Jamie Sikora. Ground state connectivity of local hamiltonians. In *Proceedings of the 42nd International Colloquium on Automata, Languages, and Programming, ICALP 2015*, pages 617–628, 2015.

**12**   Parikshit Gopalan, Phokion G. Kolaitis, Elitza N. Maneva, and Christos H. Papadimitriou. The connectivity of Boolean satisfiability: computational and structural dichotomies. *SIAM Journal on Computing*, 38(6):2330–2355, 2009.

**13**   Martin Grohe, Stephan Kreutzer, and Sebastian Siebertz. Deciding first-order properties of nowhere dense graphs. *Journal of the ACM (JACM)*, 64(3):17, 2017.

**14**   Ruth Haas and Karen Seyffarth. The k-dominating graph. *Graphs and Combinatorics*, 30(3):609–617, 2014.

**15**   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`.

**16**   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. *Theoretical Computer Science*, 412(12-14):1054–1065, 2011.

**17**   Takehiro Ito, Marcin Kamiński, and Erik D. Demaine. Reconfiguration of list edge-colorings in a graph. *Discrete Applied Mathematics*, 160(15):2199–2207, 2012.

**18**   Wm. Woolsey Johnson and William E. Story. Notes on the "15" puzzle. *American Journal of Mathematics*, 2(4):397–404, 1879.

**19**   Iyad A. Kanj and Ge Xia. Flip distance is in FPT time o(n+ k * cˆk). In *32nd International Symposium on Theoretical Aspects of Computer Science, STACS 2015*, pages 500–512, 2015.

**20**   Graham Kendall, Andrew J. Parkes, and Kristian Spoerer. A survey of NP-complete puzzles. *ICGA Journal*, pages 13–34, 2008.

**21**   Stephan Kreutzer, Roman Rabinovich, and Sebastian Siebertz. Polynomial kernels and wideness properties of nowhere dense graph classes. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017*, pages 1533–1545, 2017.

**22**   Daniel Lokshtanov, Amer E. Mouawad, Fahad Panolan, M. S. Ramanujan, and Saket Saurabh. Reconfiguration on sparse graphs. *J. Comput. Syst. Sci.*, 95:122–131, 2018.

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

**24**   Amer E. Mouawad. *On Reconfiguration Problems: Structure and Tractability*. PhD thesis, University of Waterloo, 2015.

**25**   Amer E. Mouawad, Naomi Nishimura, Vinayak Pathak, and Venkatesh Raman. Shortest reconfiguration paths in the solution space of boolean formulas. In *Automata, Languages, and Programming - 42nd International Colloquium, ICALP 2015, Kyoto, Japan, July 6-10, 2015, Proceedings, Part I*, pages 985–996, 2015.

**26**   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.

**27**   Wojciech Nadara, Marcin Pilipczuk, Roman Rabinovich, Felix Reidl, and Sebastian Siebertz. Empirical evaluation of approximation algorithms for generalized graph coloring and uniform quasi-wideness. In *17th International Symposium on Experimental Algorithms, SEA 2018*, pages 14:1–14:16, 2018.

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

**29**   Michał Pilipczuk, Sebastian Siebertz, and Szymon Toruńczyk. On the number of types in sparse graphs. In *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 799–808. ACM, 2018.

**30**   Sebastian Siebertz. Reconfiguration on nowhere dense graph classes. *Electr. J. Comb.*, 25(3):P3.24, 2018.

**31**   Akira Suzuki, Amer E. Mouawad, and Naomi Nishimura. Reconfiguration of dominating sets. *Journal of Combinatorial Optimization*, 32(4):1182–1195, 2016.

**32**   Jan van den Heuvel. The complexity of change. *Surveys in combinatorics*, 409(2013):127–160, 2013.