Colored Cut Games

Nils Morawietz[a,1,], Niels Grüttemeier[a], Christian Komusiewicz[a], Frank Sommer[a,2]

[a]*Philipps-Universität Marburg, Fachbereich Mathematik und Informatik, Marburg, Germany*

Abstract

In a graph $G = (V, E)$ with an edge coloring $\ell : E \to C$ and two distinguished vertices s and t, a colored (s,t)-cut is a set $\widetilde{C} \subseteq C$ such that deleting all edges with some color $c \in \tilde{C}$ from G disconnects s and t. Motivated by applications in the design of robust networks, we introduce *colored cut games*. In these games, an attacker and a defender choose colors to delete and to protect, respectively, in an alternating fashion. The attacker wants to achieve a colored (s,t)-cut and the defender wants to prevent this. First, we show that for an unbounded number of alternations, colored cut games are PSPACE-complete even on subcubic graphs. We then show that, even on subcubic graphs, colored cut games with i alternations are complete for classes in the polynomial hierarchy whose level depends on i. To complete the dichotomy, we show that all colored cut games are polynomial-time solvable on graphs with maximum degree at most 2.

Next, we show that all colored cut games admit a polynomial kernel for the parameter $k + \kappa_r$ where k denotes the total attacker budget and, for any constant r, κ_r is the number of vertex deletions that are necessary to transform G into a graph where the longest path has length at most r. For κ_1, which is the vertex cover number vc of the input graph, the kernel has size $\mathcal{O}(\mathrm{vc}^2 k^2)$. Moreover, we introduce an algorithm solving the most basic colored cut game, Colored (s,t)-Cut, in $2^{\mathrm{vc}+k} n^{\mathcal{O}(1)}$ time.

Keywords: Labeled Cut, Labeled Path, Network Robustness, Kernelization, PSPACE, Polynomial Hierarchy

---

⋆Some of the results of this work are also contained in the first author's Master thesis [24]

⋆⋆An extended abstract of this work appeared in the Proceedings of the 40th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'20), volume 182 of LIPIcs, pages 30:1-30:17. Schloss Dagstuhl - Leibniz- Zentrum für Informatik, 2020, held in Goa, India.

Email addresses: `morawietz@informatik.uni-marburg.de` (Nils Morawietz), `niegru@informatik.uni-marburg.de` (Niels Grüttemeier), `komusiewicz@informatik.uni-marburg.de` (Christian Komusiewicz), `fsommer@informatik.uni-marburg.de` (Frank Sommer)

[1]Supported by the Deutsche Forschungsgemeinschaft (DFG), project OPERAH, KO 3669/5-1.

[2]Partially supported by the Deutsche Forschungsgemeinschaft (DFG), project EAGR, KO 3669/6-1.

1. Introduction

Many classic computational graph problems are motivated by applications in network robustness. A famous example is the problem of computing a minimum cut between two given vertices s and t in a simple undirected graph $G = (V, E)$ [19, 14]. In some applications, a more realistic model for the robustness of a given network can be obtained by considering edge-colored graphs. Here, the input graph G comes with a coloring $\ell : E \to C$ of the edges, where C is the set of colors. For example, in multilayer networks a failure of some link in a basic network layer may result in a failure of many seemingly unrelated links in a virtual network layer, because all of the virtual links rely on paths in the basic network that use the failed link [9]. This can be modeled by assigning edge colors. A failure of the resource represented by a color c then destroys all edges with color c. Thus, whether a failure scenario disconnects two vertices s and t, depends directly on the colors of C that fail. This can be formalized as follows.

Definition 1.1. *Let G be an edge-colored graph and let s and t be two vertices in G. A color set $\tilde{C} \subseteq C$ is a* colored (s, t)-cut *in G if every (s, t)-path contains at least one edge that has a color of $\tilde{C}$.*

For example, the color set $\{2, 3, 4\}$ is a colored (s, t)-cut in Figure 1.

The size of the smallest colored (s, t)-cut then becomes an important network robustness parameter in scenarios modeled by colored graphs. Motivated by this fact, the problem of computing such a cut, called Colored (s, t)-Cut in the following, has been studied intensively [4, 9, 10, 17, 25, 31, 34]. In contrast to the classic problem on uncolored graphs, Colored (s, t)-Cut is NP-complete [9]. We may view Colored (s, t)-Cut as formulated from the perspective of an attacker whose aim is to disconnect s and t using a minimum number of edge colors. A related (s, t)-connectivity problem is Labeled Path, where we ask for a smallest color set $\tilde{C} \subseteq C$ such that there is an (s, t)-path whose edges are only colored with colors from $\tilde{C}$ [9, 17, 32]. Labeled Path is NP-complete in general [32]; when every edge color occurs at most once it is simply Shortest Path and thus solvable in polynomial time. We may view Labeled Path as formulated from the perspective of a defender who wants to secure a minimum number of edge colors in order to guarantee that s and t are connected.

We study colored cut games in which defender and attacker interact. This can be motivated from typical approaches in network security where a so-called red team (the attacker) plays against a so-called blue team (the defender) [23]. More precisely, we assume that there are two players that alternatingly choose colors. The colors chosen by the attacker are deleted from the graph while the colors chosen by the defender become safe which means that the attacker may not choose these colors in subsequent turns. In our model, in each turn the attacker and the defender have a fixed budget limiting the number of colors that they may choose, Figure 1 shows an example. We study different versions

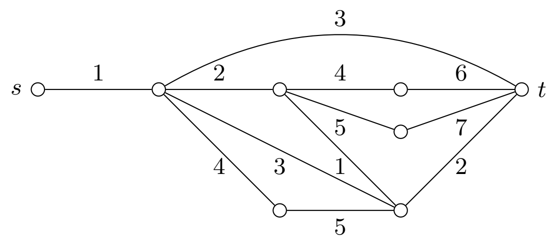

Figure 1: An edge-colored graph with seven colors. Consider a colored cut game of two rounds: In round one, the defender may protect one color and the attacker may attack two colors. In round two, the defender can protect two colors, and the attacker can attack one color. For example, the defender may protect color 1, then the attacker may delete colors 2 and 3, then in round two, the defender may protect colors 4 and 5. The resulting graph has two (s,t)-paths containing the colors $1,4,5,6$ and $1,4,5,7$, respectively. Since the attacker may now only attack either 6 or 7, the defender wins.

of this game. We distinguish, for example, whether the number of alternations between defender and attacker is constant or unbounded, whether the defender or the attacker starts, and whether we are interested in a winning strategy for the defender or the attacker. We refer to the family of these games as *colored cut games.*

COLORED (s,t)-CUT is the colored cut game where the attacker has one turn, the defender has none, and we ask if the attacker has a winning strategy. LABELED PATH can be seen as the colored cut game where the defender starts with a limited budget, followed by the attacker with unlimited budget, and we ask if the defender has a winning strategy. When the number of alternations between defender and attacker is unbounded, then we refer to the game as $(\mathrm{DA})^*$ COLORED (s,t)-CUT ROBUSTNESS ($(\mathrm{DA})^*$-CCR). A well-known special case of $(\mathrm{DA})^*$-CCR is the polynomial-time solvable SHANNON SWITCHING GAME [6, 8]. Here every edge color appears at most once and each player may choose one color in each turn.

Our Results. We study the complexity of colored cut games. In Section 3, we show that, in contrast to SHANNON SWITCHING GAME, $(\mathrm{DA})^*$-CCR is PSPACE-complete. Furthermore, in Section 4 we show that for an increasing but constant number of alternations between the agents, the colored cut games are complete for complexity classes of increasing levels of the polynomial hierarchy. The concrete level number depends directly on the number of alternations, assuming that the last turn is played by the attacker.

In Section 5, we study the complexity on restricted input instances. More precisely, in Section 5.1 we study how the structure of the input graph influences the complexity of the games. We show that all colored cut games are polynomial-time solvable on graphs with maximum degree at most 2 and hard for different levels of the polynomial hierarchy on bipartite planar subcubic graphs. Moreover, we show that two cases with restricted colorings for which

COLORED (s,t)-CUT can be solved in polynomial time are NP-hard already when the defender starts and attacker and defender have one turn.

Finally, in Section 6, we study the parameterized complexity of colored cut games. Our main result is a polynomial-size problem kernel for all colored cut games parameterized by $k + \kappa_r$. Here k is the total budget of the attacker and κ_r is the number of vertex deletions that are needed to transform the input graph G into a graph where the longest path has length at most r (thus, κ_1 is the vertex cover number vc of G). More precisely, we show that for every constant r we can reduce any instance of a colored cut game in polynomial time to one with $\mathcal{O}((\kappa_r)^2 k^{r+1})$ edges. This general kernelization result is somewhat surprising because for most parameters (including the vertex cover number vc, k, or $|C|$) even the basic colored cut games COLORED (s,t)-CUT and LABELED PATH are unlikely to admit a polynomial kernelization [17, 22, 25, 34]. In fact, the first nontrivial kernelization for COLORED (s,t)-CUT (with respect to a rather large parameter) was provided, to the best of our knowledge, in our companion work on COLORED (s,t)-CUT [25]. Finally, we present a direct FPT-algorithm solving COLORED (s,t)-CUT for the combined parameter $\kappa_r + k$ for each constant r. For $r = 1$, this algorithm has a running time of $2^{\text{vc}+k} n^{\mathcal{O}(1)}$.

One of the main tools in our hardness proofs and algorithms is the notion of colored-cut-equivalence (refer to Section 2 for a formal definition) which may be of general interest for the study of colored cuts in graphs.

2. Basic Definitions and Colored-Cut-Equivalence

We next describe the graph-theoretic notation used in this work, give the necessary definitions from parameterized complexity theory, define colored cut games formally, and present a condition for colored graphs to be considered equivalent in the context of colored cut games.

Notation. For integers j and k, $j \leq k$, we denote with $[j,k]$ the set $\{r \mid j \leq r \leq k\}$. For a set S and an integer k, we let $\binom{S}{k}$ denote the family of all size-k subsets of S. A *(simple undirected) graph* $G := (V, E)$ consists of a finite set of *vertices* $V(G) := V$ and a set of *edges* $E(G) := E \subseteq \binom{V}{2}$ and we denote $n := |V|$ and $m := |E|$. For $V' \subseteq V$, we denote with $G[V'] := (V', E \cap \binom{V'}{2})$ the *subgraph of* G *induced by* V' and with $G - V' := G[V \setminus V']$ the graph obtained from G by deleting V'. Analogously, we let $G - E' := (V, E \setminus E')$ denote the graph obtained by deleting the edge set $E' \subseteq E$. We denote with $N_G(v) := \{w \in V \mid \{v,w\} \in E\}$ the *neighborhood* of a vertex v in G and we denote with $\deg_G(v) := |N_G(v)|$ the *degree* of v in G. If G is clear from the context, we may omit the subscript. We denote the *maximum vertex degree* in a graph by Δ.

A sequence of vertices $P = (v_1, \ldots, v_k)$ is a *path* or (v_1, v_k)*-path* of length k in G if $\{v_i, v_{i+1}\} \in E(G)$ for all $1 \leq i < k$. If $v_i \neq v_j$ for all $i \neq j$, then we call P *vertex-simple*. If not mentioned otherwise, we only consider vertex-simple paths. We denote with $V(P)$ the vertices of P and with $E(P)$ the edges of P. Given two paths $P_1 = (v_1, \ldots, v_k)$ and $P_2 = (w_1, \ldots, w_r)$ in G

where $\{v_k, w_1\} \in E(G)$, we let $P_1 \cdot P_2 := (v_1, \ldots, v_k, w_1, \ldots, w_r)$ denote the *concatenation* of P_1 and P_2. If $v_k = w_1$, then we define the *merge* of P_1 and P_2 as $P_1 \multimap P_2 := (v_1, \ldots, v_k = w_1, \ldots, w_r)$. A subset $V' \subseteq V$ is a *connected component* if V' is a maximal set of vertices such that G contains at least one (u, v)-path for every pair of vertices $u \in V'$ and $v \in V'$.

Parameterized Complexity Theory. For the definition of classical complexity classes such as PSPACE or Σ_2^{P}, we refer to the textbook of Arora and Barak [2]. In the following, we give the central definitions of parameterized complexity theory that are relevant for this work; for an introduction to parameterized complexity theory and parameterized algorithms, we refer to the standard monographs [11, 13, 18, 27].

A *parameterized problem* is $L \subseteq \{0, 1\}^* \times \mathbb{N}$ and an instance of a parameterized problem (x, k) consists of an instance x of a decision problem and a *parameter* k. A *parameterized complexity class* $\mathcal{L}$ is a set of parameterized problems. We call a parameterized problem L *fixed-parameter tractable* if there is a computable function f such that for every instance $(x, k) \in \{0, 1\}^* \times \mathbb{N}$ it can be determined in $f(k) \cdot |x|^{\mathcal{O}(1)}$ time if $(x, k) \in L$. The class FPT contains exactly the parameterized problems that are fixed-parameter tractable. Furthermore, we call a parameterized problem L *slicewise polynomial* if there is a computable function f such that for every instance $(x, k) \in \{0, 1\}^* \times \mathbb{N}$ it can be determined in $|x|^{f(k)}$ time if $(x, k) \in L$. The class XP contains exactly the parameterized problems that are slicewise polynomial. Clearly, FPT is a subset of XP and it is widely assumed to be a proper subset.

Similar to classic complexity theory, we say that a *parameterized reduction* from a parameterized problem L_1 to a parameterized problem L_2 is an algorithm that transforms an instance $I_1 = (x_1, k_1)$ of L_1 into an instance $I_2 = (x_2, k_2)$ of L_2 and runs in $f(k_1) \cdot |x_1|^{\mathcal{O}(1)}$ time such that $I_1 \in L_1$ if and only if $I_2 \in L_2$ and $k_2 \leq g(k_1)$ for some computable functions f and g. Note that the parameter k_2 of I_2 only depends on k_1. Moreover, if g is a polynomial function, then the reduction is a *polynomial parameter transformation*.

The classes $\mathrm{W}[i], i \geq 0$, build the W*-hierarchy*. It holds that $\mathrm{FPT} = \mathrm{W}[0] \subseteq \mathrm{W}[1] \subseteq \cdots \subseteq \mathrm{XP}$ and it is widely assumed that these inclusions are proper [13]. A problem L' is W[i]-hard if there is a parameterized reduction from L' to L. Under the assumption $\mathrm{FPT} \neq \mathrm{W}[1]$, a parameterized problem L is *fixed-parameter intractable* if it is L is W[i]-hard for some $i \geq 1$. To show W[i]-hardness of a problem L, one may provide a parameterized reduction from some W[i]-hard problem L' to L. Furthermore, we say that a parameterized problem L is contained in $\mathrm{coW}[i], t \geq 0$, if $(\{0, 1\}^* \times \mathbb{N}) \setminus L \in \mathrm{W}[i]$. Similarly, L is coW[i]-hard if $(\{0, 1\}^* \times \mathbb{N}) \setminus L$ is W[i]-hard.

A *reduction to a problem kernel* for a parameterized problem L is a parameterized reduction from L to L that runs in polynomial time and transforms any instance (x_1, k_1) of L into an instance (x_2, k_2) of L such that $k_2 \leq k_1$ and $|x_2| \leq h(k_1)$ for some computable function h. In other words, we are able to find an equivalent instance (x_2, k_2) of L in polynomial time such that the size of (x_2, k_2) is upper-bounded by a computable function h only depending on k_1. We call h the *size of the kernel*.

A parameterized problem L admits a kernel if and only if L admits an FPT-algorithm [11]. Clearly, one is interested in finding kernels of small size for a given $L \in$ FPT. For some parameterized problems, however, one can show that it is not possible to find a kernel of polynomial size, unless NP $\subseteq$ coNP/poly [11], which is widely assumed to be false. Such (conditional) impossibility results may be transferred as follows.

If the unparameterized version of a parameterized problem L_1 is NP-hard, and there is a polynomial parameter transformation to another parameterized problem L_2 whose unparameterized version is contained in NP, then the existence of a polynomial kernel for L_2 implies the existence of a polynomial kernel for L_1. Hence, if a polynomial kernel for L_1 implies NP $\subseteq$ coNP/poly, then so does a polynomial kernel for L_2.

Colored Cut Games. An *edge-colored graph with terminals* or shortly a *colored graph* is a 5-tuple $\mathcal{H} := (G = (V, E), s, t, C, \ell)$ where G is an undirected graph, $s \in V$ and $t \in V$ are the *terminals*, C is a set of *colors* and $\ell : E \to C$ is an *edge coloring*. We denote with $|\mathcal{H}| := |G| + |C| + |\ell| = |V| + 2|E| + |C|$ the size of a colored graph.

For a graph $G = (V, E)$ and two vertices $s \in V$ and $t \in V$, we call an edge set $E' \subseteq E$ an (s,t)*-(edge-)cut* in G if s and t are in different connected components in $G - E'$. Let $\mathcal{H} = (G, s, t, C, \ell)$ be a colored graph. For a path P in G, we let $\ell(P) := \ell(E(P))$ denote the set of colors of the edges on this path. We say that $\tilde{C} \subseteq C$ is a *colored* (s,t)*-cut* in G if $\ell(P) \cap \tilde{C} \neq \emptyset$ for every (s,t)-path P in G. We say that $\tilde{C} \subseteq C$ is a *colored* (s,t)*-connector* in G if there is an (s,t)-path P in G with $\ell(P) \subseteq \tilde{C}$. We let $\mathcal{C}(\mathcal{H}) := \{\ell(P) \mid P \text{ is an } (s,t)\text{-path in } G\}$ denote the family of color sets of (s,t)-paths in G. Note that $\tilde{C} \subseteq C$ is a colored (s,t)-cut in G if and only if $\tilde{C} \cap C' \neq \emptyset$ for all $C' \in \mathcal{C}(\mathcal{H})$, that is, if $\tilde{C}$ is a hitting set for $\mathcal{C}(\mathcal{H})$. Moreover, $\tilde{C}$ is a colored (s,t)-connector in G if and only if there is a set $C' \in \mathcal{C}(\mathcal{H})$ such that $C' \subseteq \tilde{C}$.

We now formally define all colored cut games. Since the outcome of the game is decided after the last turn of the attacker, all colored cut games end with a turn of the attacker. In the most general problem variant, stated below, we allow an unbounded number of alternations between the defender D and the attacker A.

(DA)* Colored (s,t)-Cut Robustness ((DA)*-CCR)
Input: A colored graph $(G = (V, E), s, t, C, \ell)$, and two integer vectors $\vec{d} := (d_1, \ldots, d_i) \in \mathbb{N}^i$ and $\vec{a} := (a_1, \ldots, a_i) \in \mathbb{N}^i$ such that $\sum_{j=1}^{i}(d_j + a_j) \leq |C|$.
Question: $\exists D_1 \in \binom{C}{d_1}.\forall A_1 \in \binom{C \setminus D_1}{a_1}.\exists D_2 \in \binom{C \setminus (D_1 \cup A_1)}{d_2}. \cdots .\forall A_i \in \binom{C \setminus (\bigcup_{k=1}^{i-1}(D_k \cup A_k) \cup D_i)}{a_i}$: the set $\bigcup_{j=1}^{i} A_j$ is not a colored (s,t)-cut in G?

In $(\mathrm{DA})^*$-CCR we ask if the defender has a *winning strategy*. The case with a constant number $i \geq 1$ of turns is denoted by $(\mathrm{DA})^i$ Colored (s,t)-Cut Robustness ($(\mathrm{DA})^i$-CCR).

If the attacker starts the game, that is, if $d_1 = 0$, we denote the problems as $\mathrm{A(DA)}^i$ Colored (s,t)-Cut Robustness ($\mathrm{A(DA)}^i$-CCR) for all constant $i \geq 0$. We also define the complement problem $(\mathrm{DA})^*$ Colored (s,t)-Cut Vulnerability ($(\mathrm{DA})^*$-CCV) and the variants $(\mathrm{DA})^i$-CCV, and $\mathrm{A(DA)}^i$-CCV. In these problems we ask if there is a winning strategy for the attacker.

> $(\mathrm{DA})^*$ Colored (s,t)-Cut Vulnerability ($(\mathrm{DA})^*$-CCV)
> **Input:** An integer $i \geq 1$, a colored graph $(G = (V,E), s, t, C, \ell)$, and two integer vectors $\vec{d} := (d_1, \ldots, d_i), \vec{a} := (a_1, \ldots, a_i) \in \mathbb{N}^i$ such that $\sum_{k=1}^{i}(d_k + a_k) \leq |C|$.
> **Question:** $\forall D_1 \in \binom{C}{d_1}.\exists A_1 \in \binom{C \setminus D_1}{a_1}.\forall D_2 \in \binom{C \setminus (D_1 \cup A_1)}{d_2}.\cdots.\exists A_i \in \binom{C \setminus (\bigcup_{k=1}^{i-1}(D_k \cup A_k) \cup D_i)}{a_i}$: the set $\bigcup_{k=1}^{i} A_k$ is a colored (s,t)-cut in G?

We refer to all problems defined above as *colored cut games*.

Colored (s,t)-Cut is equivalent to $\mathrm{A(DA)}^0$-CCV and Labeled Path is the special case of $(\mathrm{DA})^1$-CCR where $a_1 = |C| - d_1$. Moreover, for all $i \geq 1$, $\mathrm{A(DA)}^{i-1}$-CCR is the special case of $(\mathrm{DA})^i$-CCR where the budget of the first defender turn is zero and $(\mathrm{DA})^i$-CCR is the special case of $\mathrm{A(DA)}^i$-CCR where the budget of the first attacker turn is zero. Hence, Colored (s,t)-Cut is a special case of $(\mathrm{DA})^i$-CCV and $\mathrm{A(DA)}^i$-CCV for every $i \geq 1$.

Colored-Cut-Equivalence. To argue concisely that two instances of one colored cut game are equivalent, we introduce the following definition.

Definition 2.1. *Two colored graphs $\mathcal{H} = (G, s, t, C, \ell)$ and $\mathcal{H}' = (G', s', t', C, \ell')$ are* colored-cut-equivalent *if for every $L_1 \in \mathcal{C}(\mathcal{H}) \cup \mathcal{C}(\mathcal{H}')$ there exists some $L_2 \in \mathcal{C}(\mathcal{H}) \cap \mathcal{C}(\mathcal{H}')$ such that $L_2 \subseteq L_1$.*

Thus, intuitively, only the color sets in $\mathcal{C}(\mathcal{H}) \cap \mathcal{C}(\mathcal{H}')$ are relevant for colored (s,t)-cuts. Observe, moreover, that $\mathcal{H}$ and $\mathcal{H}'$ are colored-cut-equivalent if for every (s,t)-path P in G there is an (s',t')-path P' in G' such that $\ell'(P') \subseteq \ell(P)$ and vice versa. The following lemma shows that Definition 2.1 gives us the intended property.

Lemma 2.2. *Let $\mathcal{H} = (G, s, t, C, \ell)$ and $\mathcal{H}' = (G', s', t', C, \ell')$ be two colored-cut-equivalent graphs, then $\tilde{C} \subseteq C$ is a colored (s,t)-cut in G if and only if $\tilde{C}$ is a colored (s',t')-cut in G'.*

Proof. Due to symmetry, it is sufficient to only show one direction. Let $\tilde{C}$ be a colored (s,t)-cut in G, then by definition $\tilde{C} \cap L_2 \neq \emptyset$ for all $L_2 \in \mathcal{C}(\mathcal{H}) \cap \mathcal{C}(\mathcal{H}')$. We show $\tilde{C} \cap L_1 \neq \emptyset$ for all $L_1 \in \mathcal{C}(\mathcal{H}')$. Let $L_1 \in \mathcal{C}(\mathcal{H}')$, then there is some $L_2 \in \mathcal{C}(\mathcal{H}) \cap \mathcal{C}(\mathcal{H}')$ with $L_2 \subseteq L_1$ since $\mathcal{H}$ and $\mathcal{H}'$ are colored-cut-equivalent. Hence, $L_1 \cap \tilde{C} \supseteq L_2 \cap \tilde{C} \neq \emptyset$ and therefore $\tilde{C}$ is a colored (s',t')-cut in G'. □

Thus, when considering two instances of any colored cut game with the same budget vectors, we obtain the following.

Corollary 2.3. *Two instances $I = (\mathcal{H}, \vec{d}, \vec{a})$ and $I' = (\mathcal{H}', \vec{d}, \vec{a})$ of any colored cut game are equivalent if $\mathcal{H}$ and $\mathcal{H}'$ are colored-cut-equivalent.*

The following lemmas will be useful for proving hardness on restricted input graphs. First, we show that one can easily transform non-bipartite input graphs to bipartite ones.

Lemma 2.4. *For every colored graph $\mathcal{H} = (G, s, t, C, \ell)$, one can compute in polynomial time a colored-cut-equivalent graph $\mathcal{H}' = (G', s', t', C, \ell')$ where G' is the bipartite graph obtained by subdividing each edge in G.*

Proof. We construct $\mathcal{H}'$ in polynomial time as follows: $V' := V \cup \{v_e \mid e \in E\}, E' := \{\{u, v_e\} \mid e \in E, u \in e\}, s' := s, t' := t$, and $\ell'(\{u, v_e\}) := \ell(e)$ for all $\{u, v_e\} \in E'$. That is, we subdivide every edge $\{u, w\} \in E$ into two edges $\{u, v_{\{u,w\}}\}$ and $\{v_{\{u,w\}}, w\}$ of the same color. Since all vertices $v \in V$ have only neighbors in $V' \setminus V$ and vice versa, we have that G' is bipartite.

By construction, $P = (u_1, \ldots, u_k)$ is an (s,t)-path in G if and only if $P' = (u_1, v_{\{u_1,u_2\}}, u_2, \ldots, v_{\{u_{k-1},u_k\}}, u_k)$ is an (s,t)-path in G'. Moreover, $\ell(P) = \ell'(P')$ and therefore $\mathcal{C}(\mathcal{H}) = \mathcal{C}(\mathcal{H}')$. This implies that $\mathcal{H}$ and $\mathcal{H}'$ are colored-cut-equivalent. □

Second, colored graphs of bounded maximum degree might be interesting. First, we show that if a given colored graph $\mathcal{H}$ has a colored (s,t)-cut of size one, that is, if there is a color α that occurs on every (s,t)-path in $\mathcal{H}$, then there is a colored-cut-equivalent graph with maximum degree 3.

Lemma 2.5. *Let $\mathcal{H} = (G, s, t, C, \ell)$ be a colored graph and let $\alpha \in C$ be a color that occurs on every (s,t)-path in $\mathcal{H}$, that is, if $\{\alpha\}$ is a colored (s,t)-cut in G. Then, one can compute in polynomial time a colored-cut-equivalent graph $\mathcal{H}' = (G', s', t', C, \ell')$ such that G' has a maximum degree of 3.*

Proof. We construct $\mathcal{H}'$ as follows. We start with an edgeless graph containing the vertices of V and add vertices and edges such that every vertex $v \in V$ is the root of some balanced binary tree T^v that has the leafs are $b^v_{u_1}, \ldots, b^v_{u_r}$ where $\{u_1, \ldots, u_r\} = N_G(v)$. Moreover, we assign the color α to all edges of these trees T^v with $v \in V$. Next, we add edges $\{b^v_w, b^w_v\}$ for all $\{v, w\} \in E$ and set $\ell'(\{b^v_w, b^w_v\}) := \ell(\{v, w\})$. Observe that $\mathcal{H}'$ can be constructed in polynomial time from $\mathcal{H}$. For every $v \in V$ and $x, y \in N(v)$ we define the (b^v_x, b^v_y)-path $P^v_{x,y}$ and the (v, b^v_x)-path P^v_x in G' in T^v. By construction, $\ell'(P^v_{x,y}) = \ell'(P^v_x) = \{\alpha\}$ for any $x, y \in N_G(v)$.

By construction, G' has a maximum degree of 3 so it remains to show that $\mathcal{H}$ and $\mathcal{H}'$ are colored-cut-equivalent. To this end, we prove that for every $L_{\mathcal{H}'} \in \mathcal{C}(\mathcal{H}')$, there is some $L_{\mathcal{H}} \in \mathcal{C}(\mathcal{H})$ such that $L_{\mathcal{H}} \subseteq L_{\mathcal{H}'}$ and vice versa.

First, we show that $\mathcal{C}(\mathcal{H}) \subseteq \mathcal{C}(\mathcal{H}')$ which implies that for every $L_{\mathcal{H}} \in \mathcal{C}(\mathcal{H})$, there is some $L_{\mathcal{H}'} \in \mathcal{C}(\mathcal{H}')$ such that $L_{\mathcal{H}'} \subseteq L_{\mathcal{H}}$. Let $P = (v_0, \ldots, v_r)$ be an (s,t)-path in G for some $r \geq 1$. Let $\overleftarrow{P^{v_r}_{v_{r-1}}}$ be the reverse path of $P^{v_r}_{v_{r-1}}$, then $P' = P^{v_0}_{v_1} \cdot P^{v_1}_{v_0,v_2} \cdot \ldots \cdot P^{v_r-1}_{v_{r-2},v_r} \cdot \overleftarrow{P^{v_r}_{v_{r-1}}}$ is an (s,t)-path in G' and $\ell'(P') \supseteq \bigcup_{j=0}^{r-1} \ell'(\{b^{v_j}_{v_{j+1}}, b^{v_{j+1}}_{v_j}\}) = \bigcup_{j=0}^{r-1} \ell(\{v_j, v_{j+1}\}) = \ell(P)$. By construction, every other edge in $E(P')$ is colored in α. Recall that $\alpha \in L$ for all $L \in \mathcal{C}(\mathcal{H})$. Hence, $\alpha \in \ell(P)$ and therefore $\ell'(P') = \ell(P)$.

Finally, we show that for every $L_{\mathcal{H}'} \in \mathcal{C}(\mathcal{H}')$ there is some $L_{\mathcal{H}} \in \mathcal{C}(\mathcal{H})$ such that $L_{\mathcal{H}} \subseteq L_{\mathcal{H}'}$. Let P' be an (s,t)-path in G'. Then, we know by construction that $P' = P^{v_0}_{v_1} \cdot P^{v_1}_{v_0,v_2} \cdot \ldots \cdot P^{v_r-1}_{v_{r-2},v_r} \cdot \overleftarrow{P^{v_r}_{v_{r-1}}}$ for $v_j \in V$ and where $\overleftarrow{P^{v_r}_{v_{r-1}}}$ is the reverse path of $P^{v_r}_{v_{r-1}}$. Then, $P = (v_0, \ldots, v_r)$ is an (s,t)-path in G and $\ell(P) \subseteq \ell'(P')$. Note that P might not be vertex-simple but then there exists a vertex-simple (s,t)-path $\tilde{P}$ in G with $\ell(\tilde{P}) \subseteq \ell(P) \subseteq \ell'(P')$. Note that if G is planar, we can also construct G' planar. □

Ideally, we would like to drop the restriction that every colored (s,t)-cut contains α. We show that this is not always possible. More precisely, we show that if $\mathcal{H}$ has no colored (s,t)-cut of size at most 3, then there is no colored-cut-equivalent graph for $\mathcal{H}$ of degree at most 3.

Lemma 2.6. *Let $\mathcal{H} = (G, s, t, C, \ell)$ be a colored graph and let $x \leq \max(3, |C|)$. If there is no colored (s,t)-cut of size at most x in G, then every colored graph $\mathcal{H}'$ which is colored-cut-equivalent with $\mathcal{H}$ has degree at least $x+1$.*

Proof. Assume towards a contradiction that there is a colored graph $\mathcal{H}' = (G', s', t', C, \ell')$ which is colored-cut-equivalent with $\mathcal{H}$ and has degree at most x. Let $\tilde{C}$ be the color set of edges incident with s' in G'. Since G' has maximum degree at most x, $|\tilde{C}| \leq x$. Note that $\tilde{C}$ is a colored (s',t')-cut in G'. Since G has no colored (s,t)-cut of size at most x, $\tilde{C}$ is not a colored (s,t)-cut in G. This contradicts the fact that $\mathcal{H}$ and $\mathcal{H}'$ are colored-cut-equivalent. □

We remark that by a small modification of the construction of Lemma 2.5, one can obtain a colored-cut-equivalent graph of degree $\max(3, x)$, where x is the size of a smallest colored (s,t)-cut in G. This construction is, however, not necessary for our purposes and thus omitted.

3. General Hardness Results

We first show that colored cut games are PSPACE-complete if the number of alternations between attacker and defender is unbounded by reducing from the PSPACE-complete Maker-Breaker problem [29].

(DA)*-CCR can also be seen as a Shannon Switching Game [16]: two players play against each other and every agent selects an unselected color in each turn. The game ends when there is no unselected color remaining and the attacker wins if he selected a colored (s,t)-cut. This is the case if and only if the defender has not selected a colored (s,t)-connector, since at the end of the game every color is selected by either the attacker or the defender and we ask if the defender has a winning strategy. In the classical Shannon

Switching Game, every agent selects an edge each turn instead of a color, which is polynomial-time-solvable [6, 8]. In contrast to that, we will show that (DA)*-CCR is PSPACE-complete.

Theorem 3.1. (DA)*-CCR *and* (DA)*-CCV *are* PSPACE-*complete on planar graphs even if each budget is one.*

Proof. (DA)*-CCR and (DA)*-CCV can be solved within polynomial space by a simple search tree algorithm that alternately chooses the colors for the defender and the attacker. Thus, it remains to show PSPACE-hardness. To this end we give a polynomial-time reduction from a competitive version of Hitting Set which is PSPACE-complete [29, 28].

> Maker-Breaker
> **Input:** A universe $\mathcal{U}$ with $|\mathcal{U}| = 2i$ and a collection $\mathcal{F}$ of non-empty subsets of $\mathcal{U}$.
> **Question:** $\forall d_1 \in \mathcal{U}.\exists a_1 \in \mathcal{U} \setminus \{d_1\}.\forall d_2 \in \mathcal{U} \setminus \{d_1, a_1\}. \cdots .\exists a_i \in \mathcal{U} \setminus (\{d_j, a_j \mid 1 \leq j < i\} \cup \{d_i\})$: does $F \cap \{a_j \mid 1 \leq j \leq i\} \neq \emptyset$ hold for all $F \in \mathcal{F}$?

This problem can be seen as a game between two agents where every agent selects an unselected element of the universe in each turn. The game ends when all elements of the universe are selected and the second player wins if he intersects every subset $F \in \mathcal{F}$ with the elements he chose. Otherwise, the first player wins. The definition asks if the second player has a winning strategy.

Given an instance $I = (\mathcal{U}, \mathcal{F})$ of Maker-Breaker, we construct an equivalent instance $I' = (G = (V, E), s, t, C, \ell)$ of (DA)*-CCV in polynomial time.

We set $C := \mathcal{U}$ and start with an empty graph only containing distinct vertices s and t. For every $F \in \mathcal{F}$ we add an (s, t)-path P_F such that $\ell(P_F) = F$ and where all vertices of P_F except s and t are new.

Thus, for every (s, t)-path P in G there is an $F \in \mathcal{F}$ such that $\ell(P) = F$. Consequently, $A \subseteq \mathcal{U}$ intersects every $F \in \mathcal{F}$ if and only if A is a colored (s, t)-cut in G.

Hence, a winning strategy for the attacker in the (DA)*-CCV instance I' is also a winning strategy for the second player in the Maker-Breaker instance I and vice versa. Therefore, I is a yes-instance of Maker-Breaker if and only if I' is a yes-instance of (DA)*-CCV.

Note that the constructed graph is planar. Since PSPACE-complete problems is closed under complement, (DA)*-CCR on planar graphs with unit budget in every turn is also PSPACE-complete. □

4. Complexity for constant number of alternations

Next, we analyze the complexity of (DA)^i-CCR and A(DA)^i-CCR. Recall that (DA)^i-CCR asks if the defender has a winning strategy where the defender starts and both agents have exactly i turns for some constant i.

Lemma 4.1. *For all $i \geq 1$, $(\mathrm{DA})^i$-CCV is Π^{P}_{2i}-hard and $(\mathrm{DA})^i$-CCR is Σ^{P}_{2i}-hard even on planar graphs.*

To prove Lemma 4.1, we reduce from the Π^{P}_{2i}-hard problem QSAT_{2i} [2] which we will state using the following notation. For a set of boolean variables Z, we define the set of *literals* $\mathcal{L}(Z) := Z \cup \{\neg z \mid z \in Z\}$. A subset of literals $\tilde{Z} \subseteq \mathcal{L}(Z)$ is an *assignment* of Z if $|\{z, \neg z\} \cap \tilde{Z}| = 1$ for all $z \in Z$. For a subset $X \subseteq Z$ of variables we denote with $\tau_Z(X) := X \cup \{\neg z \mid z \in Z \setminus X\}$, the assignment of Z where all variables of X occur positively and all variables of $Z \setminus X$ occur negatively. Given an assignment $\tilde{Z}$ and a *clause* $\phi \in \binom{\mathcal{L}(Z)}{3}$ we say that $\tilde{Z}$ *satisfies* ϕ (denoted by $\tilde{Z} \models \phi$) if $\phi \cap \tilde{Z} \neq \emptyset$. Analogously, $\tilde{Z}$ satisfies a set $\Phi \subseteq \binom{\mathcal{L}(Z)}{3}$ of clauses (denoted by $\tilde{Z} \models \Phi$) if $\tilde{Z} \models \phi$ for all $\phi \in \Phi$.

> QSAT_{2i}
> **Input:** A set Φ of clauses in 3-CNF over the variables Z and a partition $(X_1, Y_1, \ldots, X_i, Y_i)$ of Z.
> **Question:** Is it true that $\forall \tilde{X}_1 \subseteq X_1.\exists \tilde{Y}_1 \subseteq Y_1. \cdots .\forall \tilde{X}_i \subseteq X_i.\exists \tilde{Y}_i \subseteq Y_i : \tau_Z(\tilde{X}_1 \cup \tilde{Y}_1 \cup \cdots \cup \tilde{X}_i \cup \tilde{Y}_i) \models \Phi$?

QSAT_{2i} can be seen as a two-player game where Player 1 and Player 2 choose an assignment for X_j and Y_j, respectively, in their jth turn. We ask if Player 2 has a winning strategy, that is, if the combined assignment satisfies Φ.

Proof of Lemma 4.1. Let $I' = (Z, \Phi, X_1, Y_1, \ldots, X_i, Y_i)$ be a QSAT_{2i}-instance. We construct an instance $I = (G = (V, E), s, t, C, \ell, (d_1, \ldots, d_i), (a_1, \ldots, a_i))$ of $(\mathrm{DA})^i$-CCV as follows. Let $X_j = \{x^j_k \mid 1 \leq k \leq |X_j|\}, Y_j = \{y^j_k \mid 1 \leq k \leq |Y_j|\}$ for all $j \in [1, i]$ and let $\mathcal{L} := \mathcal{L}(Z)$. We can assume without loss of generality that $|X_j| \geq 2$ for all $j \in [2, i]$ and $|Y_j| \geq 2$ for all $j \in [1, i]$.

We set $C := \mathcal{L}$ and force the defender and the attacker to choose an assignment of the variables of X_j and $X_j \cup Y_j$, respectively, in their jth turn, otherwise they will lose.

The graph consists of three parts: the variable gadgets for the defender, the variable gadgets for the attacker and a gadget for the evaluation of the clauses. To this end, we define $G := (V, E)$ with $V := V_d \cup V_a \cup V_\Phi$ and $E := E_d \cup E_a \cup E_\Phi$ where V_d and E_d form the variable gadget of the defender gadget, V_a and E_a form the variable gadget of the attacker, and V_Φ and E_Φ form the gadget for the evaluation of the clauses. First, we introduce the variable gadget for the defender, shown in Figure 2:

- $V_d := \bigcup_{j=1}^i \{r^j_0\} \cup \{r^j_k, \top^j_k, \bot^j_k \mid 1 \leq k \leq |X_j|\}$
- $E_d := \bigcup_{j=1}^i \Big\{\{r^j_{k-1}, \top^j_k\}, \{r^j_{k-1}, \bot^j_k\}, \{\top^j_k, r^j_k\}, \{\bot^j_k, r^j_k\} \mid 1 \leq k \leq |X_j|\Big\}$,

and set the colors:

- $\ell(\{r^j_{k-1}, \top^j_k\}) := \ell(\{\top^j_k, r^j_k\}) := x^j_k$, $\ell(\{r^j_{k-1}, \bot^j_k\}) := \ell(\{\bot^j_k, r^j_k\}) := \neg x^j_k$,

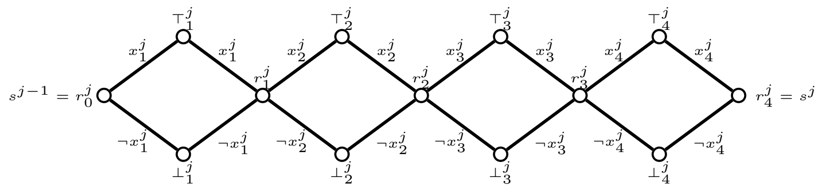

Figure 2: The gadget for the defender for the variables of X_j with $|X_j| = 4$.

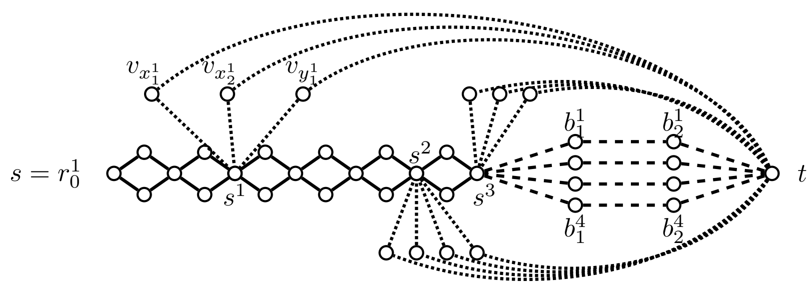

Figure 3: The final graph of the construction for the graph $G_D = (V_D, E_D)$ of Figure 2 with $|\Phi| = 4$. Solid edges belong to E_d, dotted edges belong to E_a, and dashed edges belong to E_Φ. The gadgets for the clauses are connected with s^3 and t.

where $r^j_{|X_j|}$ is identified with r_0^{j+1} for all $1 <= j < i$. The idea of the vertices $\top_k^j$ and $\bot_k^j$ is to create for every literal of variable x_k^j a path from vertex r_{k-1}^j to r_k^j. In the following, let $s := s^0 := r_0^1$ and $s^j := r^j_{|X_j|}$ for all $j \in [1, i]$. The vertex s_j is a common vertex of the gadgets for the attacker and defender. The idea is that the defender has to choose in his jth turn an assignment of the variables of X_j, or otherwise the attacker wins by taking at most two colors in his next turn. Next, we define the gadgets for the attacker:

- $V_a := \{t\} \cup \{v_x \mid x \in Z\}$
- $E_a := \bigcup_{j=1}^{i} \big\{\{s^j, v_x\}, \{v_x, t\} \mid x \in X_j \cup Y_j\big\}$,

and set the colors:

- $\ell(\{s^j, v_x\}) := x$, and $\ell(\{v_x, t\}) := \neg x$ for all $j \in [1, i], x \in X_j \cup Y_j$.

The idea is that the set of colors the attacker chooses in his jth turn is an assignment of the variables of $X_j \cup Y_j$, or otherwise the defender wins by choosing two colors in his next turn. Since a player can only choose colors that were not chosen before, the assignment for the variables of X_j of the attacker is the complement of the assignment on the variables of X_j of the defender.

Finally, we define the gadget for evaluating the clauses. To this end, we fix an ordering on every clause $\phi_j \in \Phi$ and denote with $\phi_j(y)$ the yth literal of ϕ_j. The clause gadget is then defined as:

- $V_\Phi := \{b_1^j, b_2^j \mid 1 \leq j \leq |\Phi|\}$
- $E_\Phi := \{\{s^i, b_1^j\}, \{b_1^j, b_2^j\}, \{b_2^j, t\} \mid 1 \leq j \leq |\Phi|\}$,

and we set the colors:

- $\ell(\{s^i, b_1^j\}) := \phi_j(1)$, $\ell(\{b_1^j, b_2^j\}) := \phi_j(2)$, and $\ell(\{b_2^j, t\}) := \phi_j(3)$.

That is, for every $\phi \in \Phi$, we added a new (s^i, t)-path P with $\ell(P) = \phi$. The final graph can be seen in Figure 3. Note that the constructed graphs is planar. We set $d_j := |X_j|$ and $a_j := |X_j| + |Y_j|$ for all $j \in [1, i]$. This completes the construction of I.

In the following, we let $G_\Phi := G[V_\Phi \cup \{s^i, t\}]$ denote the subgraph induced by the edges of E_Φ. Note that for every (s^i, t)-path P in G_Φ there is a clause $\phi \in \Phi$ such that $\ell(P) = \phi$. Before we show the equivalence between I and I', we make some observations on colored (s, t)-cuts and colored (s, t)-connectors in I.

Claim 4.2. *Let $\tilde{C} \subseteq \mathcal{L}$ be an assignment of Z, then $\tilde{C}$ is a colored (s^i, t)-cut in G_Φ if and only if $\tilde{C} \models \Phi$.*

Proof. Note that the set of (s^i, t)-paths in G_Φ is $\{P_j := (s^i, b_1^j, b_2^j, t) \mid 1 \leq j \leq |\Phi|\}$. By definition, $\tilde{C}$ is a colored (s^i, t)-cut in G_Φ if and only if $\tilde{C} \cap \ell(P_j) \neq \emptyset$ for all $j \in [1, |\Phi|]$. Since $\ell(P_j) = \phi_j$ for all $j \in [1, |\Phi|]$, this is the case if and only if $\tilde{C} \cap \ell(P_j) = \tilde{C} \cap \phi_j \neq \emptyset$ which is by definition the case if and only if $\tilde{C} \models \Phi$. ■

The following definition establishes the link between sensible choices of color sets for the defender and partial assignments for variables in Φ.

Definition 4.3. *We call a set of colors $D_k \subseteq C, k \in [1, i]$,* nice *if D_k is an assignment for X_k. Analogously, we call a set of colors $A_k \subseteq C, k \in [1, i]$,* nice *if A_k is an assignment for $X_k \cup Y_k$.*

Based on the defined budgets, if $\{z, \neg z\} \subseteq D_j$ for some variable $z \in X_j$, then there is a variable $z' \in X_j$ such that $\{z', \neg z'\} \cap D_j = \emptyset$. Hence, if D_j is not nice, then there is $z' \in X_j$ such that $\{z', \neg z'\} \cap D_j = \emptyset$. Clearly, the same also holds for the sets of colors that the attacker chooses.

First, we show, that if the defender chooses nice sets of colors in his first k turns, then he has completed a colored (s, s^k)-connector.

Claim 4.4. *Let $k \in [1, i]$ and $\tilde{D}_k := \bigcup_{j=1}^{k} D_j$ such that D_j is nice for all $j \in [1, k]$, then there is an (s, s^k)-path P in G with $\ell(P) \subseteq \tilde{D}_k$.*

Proof. We show that there is an (s^{j-1}, s^j)-path P^j in G with $\ell(P^j) \subseteq D_j$ if D_j is nice for any $j \in [1, k]$.

By construction, there are (r_{q-1}^j, r_q^j)-paths $P_\top = (r_{q-1}^j, \top_q^j, r_q^j)$ and $P_\bot = (r_{q-1}^j, \bot_q^j, r_q^j)$ in G with $\ell(P_\top) = \{x_q^j\}$ and $\ell(P_\bot) = \{\neg x_q^j\}$, respectively for all $q \in [1, |X_j|]$. Since $D_j \cap \mathcal{L}$ is an assignment for X_j, it follows that either $x_q^j \in D_j$, or $\neg x_q^j \in D_j$. Therefore, there is an (r_{q-1}^j, r_q^j)-path P_k^j in G with $\ell(P_q^j) \subseteq D_j \subseteq \tilde{D}_j$. Hence, there is an $(r_0^j = s^{j-1}, s^j)$-path P^j in G with $\ell(P^j) \subseteq D_j$.

Thus, there is an (s, s^k)-path P in G with $\ell(P) \subseteq \tilde{D}_k$ if D_j is nice for all $j \in [1, k]$. ■

Next, we describe some (s, t)-connectors in G assuming that the defender has chosen only sets of nice colors in his first j turns.

Claim 4.5. *Let $k \in [1, i]$ and $\tilde{D}_k := \bigcup_{j=1}^{k} D_j$ such that D_j is nice for all $j \in [1, k]$, then $\tilde{D}_k \cup \{x, \neg x\}$ for any $x \in X_k \cup Y_k$ is a colored (s, t)-connectors in G.*

Proof. By Claim 4.4, there is an (s, s^k)-path P in G with $\ell(P) \subseteq \tilde{D}_k$. Let $x \in X_k \cup Y_k$, the path $P' = (r^k_{|X_k|}, v_x, t)$ is in G and $\ell(P') = \{x, \neg x\}$. Hence, for $P'' := P \multimap P'$ it holds that $\ell(P'') = \ell(P) \cup \{x, \neg x\} \subseteq \tilde{D}_k \cup \{x, \neg x\}$. ■

Intuitively, the claim show that, if the defender has chosen nice sets of colors in his first k turns and the attacker chooses a set of colors in turn k that is not nice, then the defender can win the game in turn $k + 1$. The winning strategy for this will be shown in the proof of the equivalence between I and I'.

Second, we show, that if the attacker chooses nice color sets in his first k turns and the defender does not chose a nice color set in his $(k+1)$th turn, then the attacker can win in his $(k + 1)$th turn.

Claim 4.6. *Let $k \in [1, i]$ and $\tilde{A}_k := \bigcup_{j=1}^{k-1} A_j$ such that A_j is nice for all $j \in [1, k-1]$, then for every (s, t)-path P in G with $\ell(P) \cap \tilde{A}_k = \emptyset$ it holds that $s^j \in V(P)$ for all $j, 1 \le j \le k$.*

In other words, every (s, t)-path that is not destroyed after removing the edges colored in $\ell^{-1}(\tilde{A}_k)$ has to contain the vertex s^k.

Proof. We show this statement by induction over k. By construction, $s^0, s^1 \in V(P)$ for every (s, t)-path P in G. Hence, the Claim holds for $k = 1$.

Assume that the statement is true for some $k' = k - 1$, $k \in [2, i]$. We show that the statement is true for k. Assume towards a contradiction that there is an (s, t)-path P in G with $\ell(P) \cap \tilde{A}_k = \emptyset$ and $s^k \notin V(P)$. By the induction hypothesis, we know that $s^\ell \in V(P)$ for $\ell \le k'$. Note that by construction for every $(s^{k'}, t)$-path P^k with $r^k_1 \in V(P^k)$ it holds that $s^k \in V(P^k)$. Assume towards a contradiction that $r^k_1 \notin V(P)$ and therefore, $V(P) \cap \{v_x \mid x \in X_{k'} \cup Y_{k'}\} \neq \emptyset$. Then, there is some $x \in X_{k'} \cup Y_{k'}$ such that $\{\{s^{k'}, v_x\}, \{v_x, t\}\} \subseteq E(P)$ and therefore $\{x, \neg x\} \subseteq \ell(P)$. Since $A_{k'}$ is nice, it follows that $A_{k'} \cap \{x, \neg x\} \neq \emptyset$ and hence $\ell(P) \cap A_{k'} \neq \emptyset$ which is a contradiction. ■

Claim 4.7. *Let $k \in [1, i]$ and $\tilde{A}_k := \bigcup_{j=1}^{k-1} A_j$ such that A_j is nice for all $j \in [1, k-1]$, then $\tilde{A}_k \cup \{x, \neg x\}$ for any $x \in X_k$ is a colored (s, t)-cuts in G.*

Proof. Let $k \in [2, i]$ and assume that A_j is nice for all $j \in [1, k-1]$. By Claim 4.6 we have $s^{k-1}, s^k \in V(P)$ for every (s, t)-path P in G with $\ell(P) \cap \tilde{A}_k = \emptyset$. To prove the statement, we show that for every (s^{k-1}, s^k)-path P^k in G with $t \notin V(P^k)$, $\ell(P^k) \cap \{x, \neg x\} \neq \emptyset$ for all $x \in X_k$. We can assume that $t \notin V(P^k)$

since otherwise there is an (s,t)-path P in G with $\ell(P) \cap \tilde{A}_k = \emptyset$ and $s^k \notin V(P)$ which is impossible due to Claim 4.6.

By construction, every (s^{k-1}, s^k)-path P^k in G with $t \notin V(P^k)$ contains all vertices $r_0^k, \dots, r_{|X_k|}^k$. Thus, $\ell(P^k) \cap \{x, \neg x\} \neq \emptyset$ for all $x \in X_k$. ■

Intuitively, the claim show that, if the attacker has chosen nice sets of colors in his first $k-1$ turns and the defender chooses a set of colors in turn k that is not nice, then the attacker can win the game in turn k. The winning strategy for this will be shown in the proof of the equivalence between I and I'.

If all colors of each agent are nice in every turn, then we can show that the attacker has completed a colored (s,t)-cut if and only if the he chooses a satisfying assignment for Φ.

Claim 4.8. *Let D_j, A_j be nice for all $j \in [1,i]$ and $\tilde{A} := \bigcup_{j=1}^{i} A_j$, then $\tilde{A}$ is a colored (s,t)-cut in G if and only if $\tilde{A}$ is a colored (s^i,t)-cut in G_Φ.*

Proof. By Claim 4.6, $s^i \in V(P)$ for every (s,t)-path P in G with $\ell(P) \cap \tilde{A} = \emptyset$. Since A_i is nice, by construction, all (s^i,t)-paths $P' = (s^i, w, t)$ with $w \in V_a$ are cut. Thus, $b_1^q \in V(P)$ for some $q \in [1, |\Phi|]$. By Claim 4.4 there is an (s,s^i)-path P_d in G with $\ell(P_d) \subseteq \tilde{D}_i$. Hence, s and s^i cannot be separated by the attacker anymore. Therefore, $\tilde{A}$ is a colored (s,t)-cut in G if and only if $\tilde{A}$ is a colored (s^i,t)-cut in G_Φ. Recall that $\ell(E_\Phi) \subseteq \mathcal{L}$. Hence, $\tilde{A}$ is a colored (s,t)-cut in G if and only if $\tilde{A} \cap \mathcal{L}$ is a colored (s^i,t)-cut in G_Φ. ■

Next, we show that the QSAT_{2i} instance is a yes-instance if and only if the constructed $(\mathrm{DA})^i$-CCV instance is a yes-instance.

($\Rightarrow$) Assume that $\forall \tilde{X}_1 \subseteq X_1 . \exists \tilde{Y}_1 \subseteq Y_1 . \cdots . \forall \tilde{X}_i \subseteq X_i . \exists \tilde{Y}_i \subseteq Y_i . \tau_Z(\tilde{X}_1 \cup \tilde{Y}_1 \cup \cdots \cup \tilde{X}_i \cup \tilde{Y}_i) \models \Phi$ is true. Then, there are functions $f_k : \mathbb{P}(\bigcup_{j=1}^{k} \tilde{X}_j) \to \mathbb{P}(Y_k)$ for all $k \in [1,i]$ such that $\forall \tilde{X}_1 \subseteq X_1 . \cdots . \forall \tilde{X}_i \subseteq X_i . \tau_Z(\tilde{X}_1 \cup f_1(\tilde{X}_1) \cup \cdots \cup \tilde{X}_i \cup f_i(\bigcup_{k=1}^{i} \tilde{X}_k)) \models \Phi$ is true [3]. Herein, $\mathbb{P}(S)$ denotes the powerset of S. The functions $f_1, \dots, f_i$ are called *Skolem functions* and can be seen as the winning strategy of Player 2 in the QSAT_{2i} instance. We will use these functions to describe a winning strategy for the attacker in the $(\mathrm{DA})^i$-CCV instance iteratively.

Let D_1 be the set of colors the defender chooses in his first turn. Assume that D_1 is not nice, then $\{x, \neg x\} \cap D_1 = \emptyset$ for some $x \in X_1$. By Claim 4.6, the sets $\{x, \neg x\}$, with $x \in X_1$ are all colored (s,t)-cuts in G. Since $a_1 \geq 2$ the attacker has a winning strategy. So, we assume that D_1 is nice. Then, D_1 is an assignment for X_1. Let $\overline{D}_1 := X_1 \setminus D_1$, that is, the complement assignment of $D_1 \cap X_1$. We set $A_1 := \tau_{X_1 \cup Y_1}(\overline{D}_1 \cup f_1(\overline{D}_1))$ which is nice and disjoint from D_1.

Let $j \in [2,i]$ such that D_r and A_r are nice for all $r \in [1, j-1]$. Let D_j be the set of colors the defender chooses in his jth turn. Assume that D_j is not nice, then $\{x, \neg x\} \cap D_j = \emptyset$ for some $x \in X_j$. With Claim 4.6 we know that $\tilde{A}_{j-1} \cup \{x, \neg x\}, x \in X_j$ is a colored (s,t)-cut in G. Since $a_j \geq 2$, the attacker has a winning strategy. So, we assume that D_j is nice. Then, D_j is an assignment for X_j. Let $\overline{D}_r := X_r \setminus D_r$, that is, the complement assignment of D_r for all $r \in [1, j]$. We set $A_j := \tau_{X_j \cup Y_j}(\overline{D}_j \cup f_j(\bigcup_{r=1}^{j} \overline{D}_r))$. Observe that A_j is also nice and therefore D_r and A_r are nice for all $r \in [1, j]$.

Hence, we can assume that D_j is nice and A_j is defined as described for all $j \in [1, i]$. We show that $\tilde{A}_i := \bigcup_{j=1}^{i} A_j$ is a colored (s, t)-cut in G. By Claim 4.8, $\tilde{A}_i$ is a colored (s, t)-cut in G if $\tilde{A}_i \cap \mathcal{L}$ is a colored (s^i, t)-cut in G_Φ. By Claim 4.2 this is the case if $\tilde{A}_i$ is a satisfying assignment for Φ. Since we assumed that $\forall \tilde{X}_1 \subseteq X_1. \cdots .\forall \tilde{X}_i \subseteq X_i.\tau_Z(\tilde{X}_1 \cup f_1(\tilde{X}_1) \cup \cdots \cup \tilde{X}_i \cup f_i(\bigcup_{k=1}^{i} \tilde{X}_k)) \models \Phi$ is true, it follows that $\tilde{A}_i = \tau_Z(\overline{D}_1 \cup f_1(\overline{D}_1) \cup \cdots \cup \overline{D}_i \cup f_i(\bigcup_{k=1}^{i} \overline{D}_k)) \models \Phi$. Therefore, $\tilde{A}_i$ is a colored (s, t)-cut in G. Hence, the attacker has a winning strategy.

($\Leftarrow$) We show this direction by contraposition. In the following, we assume that $\forall \tilde{X}_1 \subseteq X_1.\exists \tilde{Y}_1 \subseteq Y_1. \cdots .\forall \tilde{X}_i \subseteq X_i.\exists \tilde{Y}_i \subseteq Y_i.\tau_Z(\tilde{X}_1 \cup \tilde{Y}_1 \cup \cdots \cup \tilde{X}_i \cup \tilde{Y}_i) \models \Phi$ is false. Therefore, $\exists \tilde{X}_1 \subseteq X_1.\forall \tilde{Y}_1 \subseteq Y_1. \cdots .\exists \tilde{X}_i \subseteq X_i.\forall \tilde{Y}_i \subseteq Y_i.\tau_Z(\tilde{X}_1 \cup \tilde{Y}_1 \cup \cdots \cup \tilde{X}_i \cup \tilde{Y}_i) \not\models \Phi$ is true. Then, there are functions $g_k : \mathbb{P}(\bigcup_{j=1}^{k-1} \tilde{Y}_j) \to \mathbb{P}(X_k)$ for all $k \in [1, i]$ such that $\forall \tilde{Y}_1 \subseteq Y_1. \cdots .\forall \tilde{Y}_i \subseteq Y_i.\tau_Z(g_1(\emptyset) \cup \tilde{Y}_1 \cup \cdots \cup g_i(\bigcup_{j=1}^{i-1} \tilde{Y}_j) \cup \tilde{Y}_i) \not\models \Phi$ is true [3]. Note that $g_1 : \{\emptyset\} \to \mathbb{P}(X_1)$, that is, the empty set is the only possible argument for g_1. The functions $g_1, \ldots, g_i$ can be seen as the winning strategy of Player 1 in the QSAT$_{2i}$ instance. We will use these functions to describe a winning strategy for the defender in the (DA)^i-CCV instance iteratively and therefore show that the attacker has no winning strategy.

Let $\overline{A}_1 := g_1(\emptyset)$ and set $D_1 := \tau_{X_1}(\overline{A}_1)$. Note that D_1 is nice. Let A_1 be the set of colors the attacker chooses in his first turn. Assume that A_1 is not nice, then $\{x, \neg x\} \cap A_1 = \emptyset$ for some $x \in X_1 \cup Y_1$. By Claim 4.5, $D_1 \cup \{x, \neg x\}, x \in X_1 \cup Y_1$ is a colored (s, t)-connector in G. Hence, the defender has a winning strategy. So, we assume that A_1 is nice. Therefore, A_1 is an assignment for $X_1 \cup Y_1$. Since D_1 is also nice, D_1 is an assignment for X_1 and $D_1 \cap A_1 = \emptyset$. Hence, $A_1 \cap X_1 = X_1 \setminus \overline{A}_1 = g_1(\emptyset)$, that is, the attacker is forced to chose $g_1(\emptyset)$ as his assignment for X_1.

Let $j \in [2, i]$ such that D_r and A_r are nice for all $r \in [1, j-1]$. For all $r \in [1, j-1]$, let $\tilde{Y}_r := A_r \cap Y_r$ for be the corresponding assignment of Y_r chosen by A_r. Let $\overline{A}_j := g_j(\bigcup_{r=1}^{j-1} \tilde{Y}_r)$ and set $D_j := \tau_{X_j}(X_j \setminus \overline{A}_j)$. Observe that D_j is nice. Now, let A_j be the set of colors the attacker chooses in his jth turn. Assume that A_j is not nice, then $\{x, \neg x\} \cap A_j = \emptyset$ for some $x \in X_j \cup Y_j$. By Claim 4.5, $\tilde{D}_j \cup \{x, \neg x\}, x \in X_j \cup Y_j$ is a colored (s, t)-connector in G. If $i = j$, then the attacker cannot complete a colored (s, t)-cut anymore and otherwise $d_{j+1} \geq 2$ by definition. In both cases the defender has a winning strategy. So, we assume that A_j is nice. Therefore, A_j is an assignment for $X_j \cup Y_j$. Since D_j is also nice, D_j is an assignment for X_j and $D_j \cap A_j = \emptyset$. Hence, $A_j \cap X_j = X_j \setminus \overline{A}_j = g_j(\bigcup_{r=1}^{j-1} \tilde{Y}_r)$, that is, the attacker is forced to pick $g_j(\bigcup_{r=1}^{j-1} \tilde{Y}_r)$ as his assignment for X_j.

Thus, assume that D_j is defined as described above and that A_j is nice for all $j \in [1, i]$. We show that $\tilde{A}_i := \bigcup_{j=1}^{i} A_j$ is not a colored (s, t)-cut in G. By

Claim 4.8, $\tilde{A}_i$ is a colored (s,t)-cut in G if and only if $\tilde{A}_i$ is a colored (s^i,t)-cut in G_Φ. By Claim 4.2 this is the case if an only if $\tilde{A}_i \cap \mathcal{L}$ is a satisfying assignment for Φ. Since we assumed that $\forall \tilde{Y}_1 \subseteq Y_1.\cdots.\forall \tilde{Y}_i \subseteq Y_i.\tau_Z(g_1(\emptyset) \cup \tilde{Y}_1 \cup \cdots \cup g_i(\bigcup_{j=1}^{i-1} \tilde{Y}_j) \cup \tilde{Y}_i) \not\models \Phi$ is true, it follows that $\tilde{A}_i \cap \mathcal{L} = \tau_Z(g_1(\emptyset) \cup \tilde{Y}_1 \cup \cdots \cup g_i(\bigcup_{r=1}^{i-1} \tilde{Y}_r) \cup \tilde{Y}_i) \not\models \Phi$. Therefore, $\tilde{A}_i$ is not a colored (s,t)-cut in G. Hence, the defender has a winning strategy and therefore the attacker cannot have a winning strategy.

Hence, I is a yes-instance of $(\mathrm{DA})^i$-CCV if and only if I' is a yes-instance of QSAT_{2i}. Therefore, $(\mathrm{DA})^i$-CCV is Π_{2i}^{P}-hard. Since $(\mathrm{DA})^i$-CCR is the complement problem of $(\mathrm{DA})^i$-CCV, it follows that $(\mathrm{DA})^i$-CCR is Σ_{2i}^{P}-hard. This completes the proof of Lemma 4.1. □

Next, we analyze the computational complexity of $\mathrm{A}(\mathrm{DA})^i$-CCR.

Corollary 4.9. *For all $i \geq 0$, $\mathrm{A}(\mathrm{DA})^i$-CCR is Π_{2i+1}^{P}-hard.*

Proof. We show this statement by a polynomial-time reduction from the $\Sigma_{2i+1}^{\mathrm{P}}$-complete problem QSAT_{2i+1} [2] to $\mathrm{A}(\mathrm{DA})^i$-CCV. Recall that QSAT_{2i+1} is the special case of QSAT_{2i+2} where $X_1 = \emptyset$. In other words, QSAT_{2i+1} starts and ends with an existential quantified set.

Let $I := (Z, \Phi, Y_1, \ldots, X_{i+1}, Y_{i+1})$ be an instance of QSAT_{2i+1}. Clearly, I is equivalent to the instance $I_2 := (Z, \Phi, X_1 = \emptyset, Y_1, \ldots, X_{i+1}, Y_{i+1})$ of QSAT_{2i+2}. Therefore, we may use the reduction in the proof of Lemma 4.1 to get an equivalent instance $I_2' = (G = (V,E), s, t, C, \ell, (d_1, \ldots, d_{i+1}), (a_1, \ldots, a_{i+1}))$ of $(\mathrm{DA})^{i+1}$-CCV. Note that by construction of I_2' we get that $d_1 = |X_1| = 0$.

Now, let $I' := (G = (V,E), s, t, C, \ell, (d_2, \ldots, d_{i+1}), (a_1, \ldots, a_{i+1}))$ be an $\mathrm{A}(\mathrm{DA})^i$-CCV instance. Clearly, I_2' is equivalent to I'. We thus constructed for I an equivalent instance I' of $\mathrm{A}(\mathrm{DA})^i$-CCV in polynomial time. Hence, $\mathrm{A}(\mathrm{DA})^i$-CCV is $\Sigma_{2i+1}^{\mathrm{P}}$-hard and $\mathrm{A}(\mathrm{DA})^i$-CCR is Π_{2i+1}^{P}-hard. □

Due to Lemma 4.1 and Corollary 4.9, it is sufficient to show that $\mathrm{A}(\mathrm{DA})^i$-CCR is in the class Π_{2i+1}^{P} and that $(\mathrm{DA})^i$-CCR is in the class Σ_{2i}^{P} to show the following theorem.

Theorem 4.10. *For all $i \geq 0$, $\mathrm{A}(\mathrm{DA})^i$-CCR is Π_{2i+1}^{P}-complete and for all $i \geq 1$, $(\mathrm{DA})^i$-CCR is Σ_{2i}^{P}-complete.*

Proof. By Lemma 4.1 and Corollary 4.9, $(\mathrm{DA})^i$-CCV is Π_{2i}^{P}-hard and $\mathrm{A}(\mathrm{DA})^i$-CCV is $\Sigma_{2i+1}^{\mathrm{P}}$-hard. Hence, it remains to show that $(\mathrm{DA})^i$-CCV $\in \Pi_{2i}^{\mathrm{P}}$ and $\mathrm{A}(\mathrm{DA})^i$-CCV $\in \Sigma_{2i+1}^{\mathrm{P}}$. We show this statement by induction over i.

Since $\mathrm{A}(\mathrm{DA})^0$-CCV and Colored (s,t)-Cut are equivalent, we get that $\mathrm{A}(\mathrm{DA})^0$-CCV is NP $= \Sigma_1^{\mathrm{P}}$-complete and $\mathrm{A}(\mathrm{DA})^0$-CCR is Π_1^{P}-complete [9]. Hence, the statement holds for $i = 0$.

Assume that the statement is true for $j-1 \geq 0$. We show that the statement is also true for j. To this end, we show an inductive steps in which we first show that the statement is true for $(\mathrm{DA})^j$-CCR if it is true for $\mathrm{A}(\mathrm{DA})^{j-1}$-CCV and afterwards that the statement is true for $\mathrm{A}(\mathrm{DA})^j$-CCV since it is rue for $(\mathrm{DA})^j$-CCR.

First, we show that the statement is true for $(\mathrm{DA})^j$-CCR if it is true for $\mathrm{A}(\mathrm{DA})^{j-1}$-CCV. Let $I = (G, s, t, C, \ell, (d_1, \ldots, d_j), (a_1, \ldots, a_j))$ be an instance of $(\mathrm{DA})^j$-CCR and let $D_1 \in \binom{C}{d_1}$. Clearly, the attacker is not able to separate vertices anymore that are connected with an edge colored in D_1. We can compute the graph G' where we remove all edges of $\ell^{-1}(D_1)$ and identify u and v for all $\{u, v\} \in E, \ell(\{u, v\}) \in D_1$ in polynomial time. Note that this graph might have parallel edges. Hence, we also subdivide every edge e into two new edges e_1' and e_2' and set $\ell'(e_1') := \ell'(e_2') := \ell(e)$. Next, we can use an oracle to solve the instance $I' = (G', s, t, C \setminus D_1, \ell', (d_2, \ldots, d_j), (a_1, \ldots, a_j))$ of $\mathrm{A}(\mathrm{DA})^{j-1}$-CCR. Since $\mathrm{A}(\mathrm{DA})^{j-1}$-CCR is Π_{2j-1}^{P}-complete as shown above, it follows that $(\mathrm{DA})^j$-CCR is Σ_{2j}^{P}-complete and $(\mathrm{DA})^j$-CCV is Π_{2j}^{P}-complete.

Second, we show that the statement is true for $\mathrm{A}(\mathrm{DA})^j$-CCV if it is true for $(\mathrm{DA})^j$-CCR. Let $I = (G, s, t, C, \ell, (d_2, \ldots, d_{j+1}), (a_1, \ldots, a_{j+1}))$ be an instance of $\mathrm{A}(\mathrm{DA})^j$-CCV and let $A_1 \in \binom{C}{a_1}$. We can compute the graph $G' := G - \ell^{-1}(A_1)$, set $C' := C \setminus A_1$, and set $\ell'(e) := \ell(e)$ for all $e \in E(G), \ell(e) \notin A_1$ in polynomial time. Next, we can use an oracle to solve the instance $I' = (G', s, t, C', \ell, (d_2, \ldots, d_{j+1}), (a_2, \ldots, a_{j+1}))$ of $(\mathrm{DA})^j$-CCV. Since $(\mathrm{DA})^j$-CCV is Π_{2j}^{P}-complete due to the induction hypothesis, it follows that $\mathrm{A}(\mathrm{DA})^j$-CCV is $\Sigma_{2j+1}^{\mathrm{P}}$-complete and $\mathrm{A}(\mathrm{DA})^j$-CCR is Π_{2j+1}^{P}-complete. $\square$

5. Restricted Instances

We now take a closer look at the classic complexity of $(\mathrm{DA})^i$, $\mathrm{A}(\mathrm{DA})^i$, and $(\mathrm{DA})^*$-CCR on restricted instances. First, we investigate graph classes like subcubic and planar graphs. Second, we consider two restricted classes of colored graphs for which Colored (s,t)-Cut is polynomial-time-solvable and show that DA-CCR is NP-complete on these restricted colored graphs.

5.1. Computational Complexity on Restricted Graph Classes

First, we show that the classic complexity of all colored cut games is the same even on bipartite planar graphs. Second, we show that $(\mathrm{DA})^i$-CCR, $\mathrm{A}(\mathrm{DA})^i$-CCR, $i \geq 1$, and $(\mathrm{DA})^*$-CCR can be solved in polynomial time on graphs with maximum degree at most 2 but cannot be solved in polynomial time on graphs with maximum degree at least 3, unless P = NP.

By Corollary 2.3 and Proposition 2.4, we can find for each instance I of a colored cut game an equivalent instance I' of the same colored cut game such that the graph of I' is bipartite planar if the graph of I is planar. Hence, the hardness results of Theorem 4.10 and Theorem 3.1 imply the following.

Theorem 5.1. *Even on bipartite planar graphs, for all $i \geq 1$, $(\mathrm{DA})^i$-CCR is Σ_{2i}^{P}-complete and for all $i \geq 0$, $\mathrm{A}(\mathrm{DA})^i$-CCR is Π_{2i+1}^{P}-complete. Furthermore, $(\mathrm{DA})^*$-CCR and $(\mathrm{DA})^*$-CCV are PSPACE-complete even on planar graphs.*

In the following, we focus on graphs with a constant maximum degree. To prove that $(\mathrm{DA})^i$-CCR, $\mathrm{A}(\mathrm{DA})^i$-CCR, with $i \geq 1$ and $(\mathrm{DA})^*$-CCR can be solved in polynomial time on graphs with maximum degree at most 2, we first prove the following lemmas.

Intuitively, the first lemma states that the outcome of a colored cut game is determined at the latest at the turn in which the total attacker budget so far is at least $|\mathcal{C}(\mathcal{H})|$, the number of different color sets of vertex-simple (s,t)-paths in G.

Lemma 5.2. *Let $I = (G, s, t, C, \ell, (d_1, \ldots, d_i), (a_1, \ldots, a_i))$ be an instance of $(\mathrm{DA})^i$-CCR and if there is some $j \leq i$ such that $\sum_{r=1}^{j} a_r \geq |\mathcal{C}(I)|$, that is, the sum of the budget of the attacker in the first j turns is at least the number of color sets of vertex-simple (s,t)-paths in G, then I is a yes-instance for $(\mathrm{DA})^i$-CCR if and only if the instance $I' = (G, s, t, C, \ell, (d_1, \ldots, d_j), (a_1, \ldots, a_j))$ of $(\mathrm{DA})^j$-CCR is a yes-instance.*

Proof. The attacker can intersect at least one $L \in \mathcal{C}(I)$ for each color he chooses. Hence, after choosing at most $|\mathcal{C}(I)|$ colors, he has completed a colored (s,t)-cut, unless the defender has completed a colored (s,t)-connector before. Thus, the outcome of the game is determined at the latest after the attacker's jth turn. □

Intuitively, the second lemma states that the player with the winning strategy can be determined efficiently if the total attacker budget is at most one.

Lemma 5.3. $(\mathrm{DA})^*$-CCR *can be solved in $\mathcal{O}(i+|C|\cdot(n+m))$ time if $\sum_{j=1}^{i} a_j \leq 1$ for the total attacker budget.*

Proof. First, we show that DA-CCR can be solved in $\mathcal{O}(|C|(n+m))$ time if $a_1 \leq 1$. Let $I = (G = (V, E), s, t, C, \ell, d_1, a_1)$ be an instance of DA-CCR. We can assume that s and t are in the same connected component in G, since otherwise I is a trivial no-instance of DA-CCR. Hence, if $a_1 = 0$, I is a trivial yes-instance of DA-CCR. If $a_1 = 1$, we can compute the set of all colored (s,t)-cuts of size exactly one, that is, $A \subseteq C$ such that $\{\alpha\}$ is a colored (s,t)-cut in G for all $\alpha \in A$. This can be done in $\mathcal{O}(|C| \cdot (n+m))$ time.

We show that I is a yes-instance if and only if $d \geq |A|$. If the attacker is able to choose some $\alpha \in A$ in his turn, then he has picked a colored (s,t)-cut and therefore, the defender will lose the game. Therefore, I is a no-instance if $d < |A|$. If $d \geq |A|$, then I is a yes-instance, since the defender can choose $D_1 \supseteq A$ and therefore, there is no colored (s,t)-cut of size one left.

Next, we show that $(\mathrm{DA})^*$-CCR can be solved in $\mathcal{O}(i + |C| \cdot (n+m))$ time if $\sum_{j=1}^{i} a_j \leq 1$. Let $I = (G = (V, E), s, t, C, \ell, (d_1, \ldots, d_i), (a_1, \ldots, a_i))$ be an instance of $(\mathrm{DA})^*$-CCR with $\sum_{j=1}^{i} a_j \leq 1$. Recall that if $\sum_{j=1}^{i} a_j = 0$, the defender wins if and only if s and t are in the same connected component in G. Thus, assume that $\sum_{j=1}^{i} a_j = 1$. Hence, there is some x with $x \in [1, i]$, such that $a_x = 1$ and $a_y = 0$ for all $y \in [1, i]$ with $y \neq x$. Recall that the defender cannot change the outcome of the game after the attacker has performed his last turn. This also holds for the last turn in which the attacker is able to select a set of colors of size at least one. Therefore, I is a yes-instance, if the $(\text{DA})^x$-CCR instance $\tilde{I} := (G, s, t, C, \ell, (d_1, \dots, d_x), (a_1, \dots, a_x))$ is a yes-instance. Note that $a_1 = a_2 = \dots = a_{x-1} = 0$. Hence, $\tilde{I}$ is a yes-instance, if the DA-CCR instance $I' := (G, s, t, C, \ell, d, a)$ is a yes-instance where $d := \sum_{j=1}^{x} d_j$ and $a = 1$. Recall that we can solve I' in $\mathcal{O}(|C|(n+m))$ time and therefore we can solve the equivalent instance I in $\mathcal{O}(i + |C|(n+m))$ time. The additional summand i in the running time, comes from finding the turn x where $a_x = 1$.

Thus, we conclude that $(\text{DA})^*$-CCR can be solved in $\mathcal{O}(i + |C|(n+m))$ time if $\sum_{j=1}^{i} a_j \leq 1$. □

Observe that we can assume that for each turn either the budget of the defender or the attacker is at least one. Thus, we can assume that $i \leq 2|C|$ and hence the running time of Lemma 5.3 is $\mathcal{O}(|C| \cdot (n+m))$. Since $(\text{DA})^i$-CCR and $\text{A}(\text{DA})^i$-CCR are special cases of $(\text{DA})^*$-CCR, $(\text{DA})^i$-CCR and $\text{A}(\text{DA})^i$-CCR can also be solved in $\mathcal{O}(|C|(n+m))$ time if $\sum_{j=1}^{i} a_j \leq 1$.

Lemma 5.4. $(\text{DA})^i$-CCR*,* $\text{A}(\text{DA})^i$-CCR *with* $i \geq 1$*, and* $(\text{DA})^*$-CCR *can be solved in polynomial time on graphs with maximum degree at most two.*

Proof. Let $\mathcal{H} = (G, s, t, C, \ell)$ be a colored graph where G has maximum degree at most two. Consequently, there are at most two (s,t)-paths in G and therefore $|\mathcal{C}(\mathcal{H})| \leq 2$ and can be computed in polynomial time.

We prove the statement by an induction over i. To this end, we will show the base case for $i = 1$ and $i = 2$.

First, let $I_1 = (\mathcal{H}, d, a)$ be an instance of $(\text{DA})^1$-CCR. We show that we can solve I_1 in polynomial time. Note that I_1 is a no-instance if $\mathcal{C}(\mathcal{H}) = \emptyset$, that is, if s and t are not in the same connected component in G. Hence, we can assume that $\mathcal{C}(\mathcal{H}) \neq \emptyset$. By Lemma 5.3, we can solve I_1 in polynomial time if $a \leq 1$. Therefore, assume that $a \geq 2$. Since $a \geq 2$ and $|\mathcal{C}(\mathcal{H})| \leq 2$, the attacker can win the game if the defender has not picked a colored (s,t)-connector in his turn. Hence, I_1 is a yes-instance if and only if $d \geq \min\{|L| \mid L \in \mathcal{C}(\mathcal{H})\}$ which can be checked in polynomial time.

Next, let $I_2 = (\mathcal{H}, (d_1, d_2), (a_1, a_2))$ be an instance of $(\text{DA})^2$-CCR. We show that we can solve I_2 in polynomial time. Note that I_2 is a no-instances if $\mathcal{C}(\mathcal{H}) = \emptyset$, that is, if s and t are not in the same connected component in G. Hence, we can assume that $\mathcal{C}(\mathcal{H}) \neq \emptyset$. By Lemma 5.3, we can solve I_2 in polynomial time if $a_1 + a_2 \leq 1$. Therefore, assume that $a_1 + a_2 \geq 2$ and $\mathcal{C}(\mathcal{H}) \neq \emptyset$. If $a_1 \geq |\mathcal{C}(\mathcal{H})|$ then I_2 is equivalent to the $(\text{DA})^1$-CCR instance $(\mathcal{H}, d_1, a_1)$ due to Lemma 5.2 and therefore can be solved in polynomial time. Moreover, if $a_1 = 0$, I_2 is equivalent to the $(\text{DA})^1$-CCR instance $(\mathcal{H}, d_1 + d_2, a_2)$ and therefore can be solved in polynomial time. Hence, we can assume that $a_1 = 1, a_2 \geq 1$, and $|\mathcal{C}(\mathcal{H})| = 2$. Let $\mathcal{C}(\mathcal{H}) = \{L_1, L_2\}$ and assume that $|L_1| \leq |L_2|$. Note that the defender can win the game in his first turn if $d \geq |L_1|$. If the defender does not choose D_1 such that $D_1 \supseteq L_1 \cap L_2$, then the attacker can win the game in his first turn by taking only one color $\alpha \in (L_1 \cap L_2) \setminus D_1$. Therefore, I_2 is a no-instance if $d_1 < |L_1 \cap L_2|$. Thus, assume that $|L_1 \cap L_2| \leq d_1 < |L_1|$. Since D_1 has to be a superset of $L_1 \cap L_2$, we can reduce the instance such that $L_1 \cap L_2 = \emptyset$ by decreasing d_1 by $|L_1 \cap L_2|$ and merging the endpoints of edges $e \in E$ with $\ell(e) \in L_1 \cap L_2$. Note that the attacker can pick $\alpha \in L_i, i \in \{1,2\}$, in his first turn where $|L_i \setminus D_1| = \min(|L_1 \setminus D_1|, |L_2 \setminus D_1|)$, that is, the set that is closest to being fully chosen by the defender. If $d_1 + d_2 < |L_2|$ then the defender loses the game since the attacker can choose $\alpha \in L_1$ in his first turn and the defender cannot complete a colored (s,t)-connector, since $|L_2| - |D_1| > d_2$. Thus, assume that $d_1 \geq |L_2| - d_2, a_1 = 1, a_2 \geq 1$, and $L_1 \cap L_2 = \emptyset$.

If $d_1 \geq |L_1| - d_2 + |L_2| - d_2$ then the defender can choose D_1 such that $|L_1 \setminus D_1| \leq d_2$ and $|L_2 \setminus D_1| \leq d_2$. By the fact that $a_1 = 1$ and $L_1 \cap L_2 = \emptyset$, the attacker can cut at most one path with the color α in his first turn. Therefore, in the second turn of the defender there is an $i \in \{1,2\}$ such that $\alpha \notin L_i$ and $|L_i \setminus D_1| \leq d_2$. Hence, the defender can win by choosing $D_2 \supseteq (L_i \setminus D_1)$: We have $L_i \subseteq D_1 \cup D_2$ and therefore $D_1 \cup D_2$ is a colored (s,t)-connector and I_2 is a yes-instance.

Otherwise, if $d_1 < |L_1| - d_2 + |L_2| - d_2$ then the defender is not able to choose a set D_1 such that $|L_1 \setminus D_1| \leq d_2$ and $|L_2 \setminus D_1| \leq d_2$. Assume without loss of generality that $|L_2 \setminus D_1| > d_2$, then the attacker can choose $\alpha \in (L_1 \setminus D_1)$. Hence, the defender can now only win by choosing D_2 such that $(D_1 \cup D_2) \supseteq L_2$ which is not possible since $|L_2 \setminus D_1| > d_2 = |D_2|$. Therefore, the defender loses and I_2 is a no-instance.

Note that we checked all possibilities and that all checks in this algorithm can be done in polynomial time and therefore, we can solve I_2 in polynomial time. This completes the base case.

Let $j \geq 3$ and assume that the statement is true for $j-1$, we show that the statement is also true for j. Let $I = (G, s, t, C, \ell, (d_1, \ldots, d_j), (a_1, \ldots, a_j))$ be an instance of $(\mathrm{DA})^j$-CCR. If $a_1 = 0$ or $a_2 = 0$, then we can construct an equivalent instance I' of $(\mathrm{DA})^{j-1}$-CCR in polynomial time. Due to the induction hypothesis, we can solve I' and therefore I in polynomial time. Hence, assume that $a_1 \neq 0$ and $a_2 \neq 0$. So, $a_1 + a_2 \geq 2 \geq |\mathcal{C}(I)|$ and therefore I is equivalent to the $(\mathrm{DA})^2$-CCR instance $I_2 = (G, s, t, C, \ell, (d_1, d_2), (a_1, a_2))$ due to Lemma 5.2. Since we can solve the base case I_2 in polynomial time, we can also solve I, in polynomial time.

The statement for $\mathrm{A(DA)}^i$-CCR follows analogously. □

We obtain a dichotomy by showing that, for all $i \geq 1$, $(\mathrm{DA})^i$-CCR and $\mathrm{A(DA)}^i$-CCR are Σ_{2i}^{P}-hard on graphs with maximum degree at least 3.

Lemma 5.5. *For all $i \geq 1$, $(\mathrm{DA})^i$-CCR and $\mathrm{A(DA)}^i$-CCR are Σ_{2i}^{P}-hard even on subcubic graphs and $(\mathrm{DA})^*$-CCR is PSPACE-hard even on subcubic graphs.*

Proof. Since $(\mathrm{DA})^i$-CCR is a special case of $\mathrm{A(DA)}^i$-CCR, we only have to show that $(\mathrm{DA})^i$-CCR is Σ_{2i}^{P}-hard even on subcubic graphs.

Let $I' = (G' = (V', E'), s', t', C', \ell', (d'_1, \ldots, d'_i), (a'_1, \ldots, a'_i))$ with $a'_1 \geq 1$ be an instance of $(\mathrm{DA})^i$-CCR. We construct an instance I of $(\mathrm{DA})^i$-CCR with maximum degree 3 as follows: We add a new color α, a new vertex s, set $t := t', V := V' \cup \{s\}, E := E' \cup \{\{s, s'\}\}, C := C' \cup \{\alpha\}, \ell(\{s, s'\}) :=$

$\alpha, d_1 := d'_1 + 1, a_1 := a'_1, d_j := d'_j$, and $a_j := a'_j$ for all $j \in [2, i]$. Furthermore, let $\mathcal{H} = (G = (V, E), s, t, C, \ell)$ be the corresponding colored graph. Recall that with Lemma 2.5 we can construct a colored-cut-equivalent graph with maximum degree 3 since α occurs in every (s, t)-path. Now, the equivalence follows from Corollary 2.3. □

Theorem 5.6 follows directly from Lemmas 5.4 and 5.5.

Theorem 5.6. *Let* $i \geq 1$*. The problems* $(\mathrm{DA})^i$-CCR*,* $\mathrm{A(DA)}^i$-CCR*, and* $(\mathrm{DA})^*$-CCR *can be solved in polynomial time on graphs with a maximum degree at most* 2*. On bipartite planar subcubic graphs with a maximum degree* 3*,* $(\mathrm{DA})^i$-CCR *and* $\mathrm{A(DA)}^i$-CCR *are* Σ_{2i}^{P}*-hard and* $(\mathrm{DA})^*$-CCR *is* PSPACE*-hard.*

5.2. Restricted Colorings

We now analyze the complexity of $(\mathrm{DA})^1$-CCR on instances where the graph may be arbitrary but the coloring is restricted. First, we consider the case where every color appears in at most two (s, t)-paths. In this case, Colored (s, t)-Cut can be solved in polynomial time [9, 19, 30]. We show that $(\mathrm{DA})^1$-CCR is NP-complete. Hence, for any $i \geq 1$, $(\mathrm{DA})^i$-CCR and $\mathrm{A(DA)}^i$-CCR cannot be solved in polynomial time on these restricted colored graphs, unless P = NP. Since there is only one turn per agent, the budget-vectors are only single integers, and we use the notation $a := a_1$ and $d := d_1$.

Theorem 5.7. DA-CCR *is* NP*-complete and* W[1]*-hard when parameterized by* d *even if every color appears in at most two* (s, t)*-paths.*

Proof. First, we show that DA-CCR is contained in NP if every color appears in at most two (s, t)-paths. Let $I = (G = (V, E), s, t, C, \ell, d, a)$ be a yes-instance of DA-CCR where every color appears in at most two (s, t)-paths and let $D_1 \subseteq S$ with $|D_1| \leq d$ be a winning strategy of the defender. We define a graph G' where we identify the vertices $u, v \in V$ if there is an (u, v)-path P in G with $\ell(P) \subseteq D_1$. Since the resulting graph might have parallel edges, we also subdivide every edge. In the resulting colored graph, every color still appears in at most two (s, t)-paths [25]. Thus, we can check in polynomial time if there is a colored (s, t)-cut of size at most a in the constructed colored graph [30] and therefore verify that D_1 is a winning strategy for the defender. Hence, DA-CCR is contained in NP if every color appears in at most two (s, t)-paths.

Second, we show that DA-CCR is NP-hard even if every color appears in at most two (s, t)-paths by giving a polynomial-time reduction from the NP-complete problem Matching Interdiction [33].

> Matching Interdiction
> **Input:** A graph $G = (V, E)$ and integers b and r.
> **Question:** Is there a subset $S \subseteq E$ with $|S| \leq b$ such that the maximum matching in $G - S$ has size at most r?

In other words, we ask if there is a set $S \subseteq E$ such that there are no $r+1$ disjoint edges in $G - S$. Given an instance $I = (G = (V, E), b, r)$ of MATCHING INTERDICTION, we build in polynomial time an equivalent instance $I' = (G' = (V', E'), s, t, C, \ell, d, a)$ of DA-CCR where every color appears in at most two (s,t)-paths. Since the maximum matching in G has size at most $|V|/2$, I is a yes-instance if $r \geq |V|/2$. Hence, we can assume without loss of generality that $r \leq |V|/2 - 1$.

We start with an empty graph G', set $d := b, a := |V| - r - 1, C := \{\alpha_j^v \mid v \in V, 0 \leq j \leq d\} \cup E$ and add vertices s and t to G'. Furthermore, we add for every $v \in V$ an (s,t)-path P_v in G' such that $\ell(P_v) = \{\alpha_j^v \mid 0 \leq j \leq d\} \cup \{e \in E \mid v \in e\}$.

Note that for distinct vertices $v, w \in V$, $\ell(P_v) \cap \ell(P_w) = \{\{v, w\}\}$ if $\{v, w\} \in E$ and $\ell(P_v) \cap \ell(P_w) = \emptyset$, otherwise. Moreover, every color $e \in E$ appears on exactly two (s,t)-paths in G' and every color $\alpha \in C \setminus E$ appears on exactly one (s,t)-path. By construction, there are exactly $|V|$ many (s,t)-paths in G' and all of them have pairwise different sets of colors. Hence, $|\mathcal{C}(I')| = |V|$. The idea of this construction is that the defender is not able to choose a colored (s,t)-connector since each (s,t)-path contains at least $d+1$ different colors and therefore he only has a winning strategy if he is able to reduce the size of the maximum matching in G. We now give the formal proof of this intuition. That is, we show that I is a yes-instance of MATCHING INTERDICTION if and only if I' is a yes-instance of DA-CCR.

($\Rightarrow$) Let $S \subseteq E$, such that there is no matching of size $r+1$ in $G - S$. We will show that there is no colored (s,t)-cut $A_1 \subseteq C \setminus S$ of size at most a in G'. Assume towards a contradiction that there is a colored (s,t)-cut $A_1 \subseteq C \setminus S$ of size at most a in G'. Recall that every color appears in at most two (s,t)-paths in G'. Since $|A_1| \leq a = |V| - r - 1$ and $|\mathcal{C}(I')| = |V|$, there is a set of colors $R \subseteq A_1$ of size at least $r+1$ such that every color $\alpha \in R$ appears in two (s,t)-paths and for all distinct colors $\alpha, \beta \in R$, there is no (s,t)-path P' in G' with $\{\alpha, \beta\} \subseteq \ell(P')$. By construction, only the colors $E \subseteq C$ appear in exactly two (s,t)-paths and therefore $R \subseteq E$. For every pair of distinct edges $e_1 := \{u_1, w_1\} \in R$ and $e_2 := \{u_2, w_2\} \in R$ in I (these are distinct colors in I'), it holds that $\ell(E(P_{u_1}) \cup E(P_{w_1})) \cap \ell(E(P_{u_2}) \cup E(P_{w_2})) = \emptyset$. Hence $e_1 \cap e_2 = \emptyset$ and therefore R is a matching of size $r+1$ in $G - S$, a contradiction.

($\Leftarrow$) Let $D_1 \subseteq C$ be a color set of size at most d such that there is no colored (s,t)-cut $A_1 \subseteq C \setminus D_1$ in G' of size at most a in G'. By construction, there is no colored (s,t)-connector of size at most d in G' and, therefore, for every $v \in V$, there is some $\beta_v \in \ell(P_v) \setminus D_1$. We will show that $G - (D_1 \cap E)$ has no matching of size $r+1$. Assume towards a contradiction that there is a matching M of size at most $r+1$ in $G - (D_1 \cap E)$. Then, $A_1 := M \cup \{\beta_v \mid v \in V \setminus (\bigcup_{e \in M} e)\}$ has size at most $r + 1 + |V| - 2(r+1) = |V| - (r+1) = a$ and $A_1 \cap D_1 = \emptyset$. By construction, A_1 is a colored (s,t)-cut in G', since for every (s,t)-path P_v with $v \in V$ it holds that either $\beta_v \in A_1$ or $\ell(P_v) \cap M \neq \emptyset$, a contradiction.

Clearly, this is a parameterized reduction from MATCHING INTERDICTION parameterized by b to DA-CCR parameterized by d. Since MATCHING INTER-

DICTION parameterized by b is W[1]-hard [33], DA-CCR parameterized by d is W[1]-hard even if every color appears in at most two (s,t)-paths □

Second, we consider the case when no two edges have the same color or, in other words, the case of uncolored input graphs. In this case, COLORED (s,t)-CUT can be solved in polynomial time since it is is equivalent to the classical MIN (s,t)-CUT problem [19]. Consequently, if we restrict DA-CCR to uncolored graphs, we ask if there is a set D of edges of size d such that every (s,t)-edge-cut distinct from D has size at least $a+1$. This exact problem has been considered for both weighted and unweighted graphs in our companion work [20] under the name MIN (s,t)-CUT PREVENTION.

Theorem 5.8 ([20])**.** MIN (s,t)-CUT PREVENTION *is* NP*-complete and* W[1]*-hard when parameterized by d.*

This implies the following for games with a fixed number of alternations.

Theorem 5.9. *For $i \geq 1$,* $(\mathrm{DA})^i$-CCR *and* $\mathrm{A(DA)}^i$-CCR *are both* NP*-hard and* W[1]*-hard when parameterized by $d := \sum_{j=1}^{i} d_j$ on uncolored graphs.*

Consequently, COLORED (s,t)-CUT and its complement problem are the only colored cut games that can be solved in polynomial time on uncolored graphs or if every color appears in at most two (s,t)-paths, unless P = NP.

6. Parameterized Complexity of Colored Cut Games

Finally, we analyze the parameterized complexity of the colored cut games. First, we investigate budget-related parameters. Second, we present polynomial kernels for all colored cut games. Since COLORED (s,t)-CUT does not admit a polynomial kernel when parameterized by $|C|$, the kernels use parameters larger than $|C|$.

6.1. Parameterization by the Total Budget $b(I)$ and $|C|$

For an instance $I = (\mathcal{H}, \vec{d}, \vec{a})$ of a colored cut game we denote with

$$b(I) := \sum_{x=1}^{i} (d_x + a_x)$$

the sum of all budgets and with

$$k := \sum_{x=1}^{i} a_x$$

the total budget of the attacker. First, we investigate the parameter $b(I)$. COLORED (s,t)-CUT is W[2]-hard when parameterized by $k = b(I)$ [9]. Similar to COLORED (s,t)-CUT, we will show that all the colored cut games do not admit an FPT-algorithm when parameterized by $b(I)$. Moreover, we show that all colored cut games are in FPT and do not admit polynomial kernels when parameterized by $|C|$.

Proposition 6.1. $(\mathrm{DA})^i$-CCR*,* $i \geq 1$*,* $\mathrm{A}(\mathrm{DA})^i$-CCR*,* $i \geq 0$*, and* $(\mathrm{DA})^*$-CCR *parameterized by* $b(I)$ *are* coW[2]*-hard and can be solved in* $\mathcal{O}(|C|^{b(I)}(n+m))$ *time.*

Proof. First, we describe the XP-algorithm for $(\mathrm{DA})^*$-CCR. By using an and-or tree of depth $2i+1$ that branches into all possible sets of unchosen colors of size equal to the budget of the current turn, we can determine in time $\mathcal{O}(\binom{|C|}{d_1}\binom{|C|-d_1}{a_1}\dots\binom{|C|-\sum_{j=1}^{i-1}(d_j+a_j)-d_i}{a_i}(n+m)) \subseteq \mathcal{O}(|C|^{b(I)}(n+m))$ if the defender has a winning strategy. Since $(\mathrm{DA})^i$-CCR and $\mathrm{A}(\mathrm{DA})^i$-CCR are special cases of $(\mathrm{DA})^*$-CCR, the same algorithm works for $(\mathrm{DA})^i$-CCR and $\mathrm{A}(\mathrm{DA})^i$-CCR as well.

Second, we show the coW[2]-hardness. Let $I' = (G, s, t, C, \ell, k)$ be a COLORED (s,t)-CUT instance. Since COLORED (s,t)-CUT is a special case of $(\mathrm{DA})^i$-CCV,$\mathrm{A}(\mathrm{DA})^i$-CCV, and $(\mathrm{DA})^*$-CCV where all budgets except a_1 are set to zero, we can give a trivial reduction from COLORED (s,t)-CUT to any of these problems where $b(I) = a_1 = k$. The statement follows since COLORED (s,t)-CUT is W[2]-hard parameterized by k [17, 30]. The coW[2]-hardness for $(\mathrm{DA})^i$-CCR (and generalizations) follows from considering the complement problem. □

By definition, $b(I) \leq |C|$. Hence, the algorithm of Proposition 6.1 with a running time of $\mathcal{O}(|C|^{b(I)}(n+m))$ also implies an FPT-algorithm for the parameter $|C|$.

Corollary 6.2. $(\mathrm{DA})^i$-CCR*,* $\mathrm{A}(\mathrm{DA})^{i-1}$-CCR*,* $i \geq 1$*, and* $(\mathrm{DA})^*$-CCR *can be solved in time* $\mathcal{O}(\min(|C|^{|C|}, 2^{2i|C|})(n+m))$ *and do not admit a polynomial kernel when parameterized by* $|C|$*, unless* NP $\subseteq$ coNP/poly.

Proof. With the algorithm described in Proposition 6.1 and the fact that $b(I) \leq |C|$, a running time of $\mathcal{O}(|C|^{|C|}(n+m))$ follows directly. Moreover, $\binom{|C|}{j} \leq 2^{|C|}$ for every $j \in [0, |C|]$ and therefore, I can be solved in $\mathcal{O}(2^{2i|C|}(n+m)) = \mathcal{O}((2^{|C|})^{2i}(n+m)) \supseteq \mathcal{O}(\binom{|C|}{d_1}\binom{|C|-d_1}{a_1}\dots\binom{|C|-\sum_{j=1}^{i-1}(d_j+a_j)-d_i}{a_i}(n+m))$ time.

Unless NP $\subseteq$ coNP/poly, COLORED (s,t)-CUT parameterized by $|C|$ does not admit a polynomial compression [30] and therefore also no polynomial kernel [11]. Since COLORED (s,t)-CUT is a special case of all colored cut games where we ask if the attacker has a winning strategy, none of these colored cut games admits a polynomial kernel when parameterized by $|C|$, unless NP $\subseteq$ coNP/poly. □

For DA-CCR and DA-CCV, we can improve the running time.

Proposition 6.3. DA-CCR *can be solved in* $\mathcal{O}(2^{|C|}(n+m))$ *time.*

Proof. Recall that in DA-CCR we ask if there is a set $D_1 \subseteq C$ of size at most d such that there is no colored (s,t)-cut $A_1 \subseteq (C \setminus D_1)$ of size at most a in G. In the following, we call a set $\tilde{D} \subseteq C$ *safe* if $\tilde{D}$ has size at least a and if there is no colored (s,t)-cut $A_1 \subseteq \tilde{D}$ of size at most a in G, that is, if the defender chooses all colors in $C \setminus \tilde{D}$, the attacker is not able to select a colored (s,t)-cut of size at most a. In other words, the defender wins if and only if there is a safe set $\tilde{D} \subseteq C$ of size at least $|C| - d$. We describe an algorithm that runs in $\mathcal{O}(2^{|C|}(n+m))$ time and checks if there is a safe set $\tilde{D} \subseteq C$ of size at least $|C| - d$.

The algorithm computes iteratively the sets S_j of all safe sets of colors $\tilde{D}_j \in \binom{C}{j}$ of size exactly j for every $j \in [a, |C| - d]$. Clearly, the defender has a winning strategy if $S_{|C|-d} \neq \emptyset$. We can compute the set S_a in $\mathcal{O}(\binom{|C|}{a}(n+m))$ time by checking for every $\tilde{D}_a \in \binom{C}{a}$ in $\mathcal{O}(|C|(n+m))$ time if $\tilde{D}_a$ is not a colored (s,t)-cut in G. Next, we use the set S_j to compute the set S_{j+1} for every $j \in [a, |C| - d - 1]$. By definition, $\tilde{D}$ with $|\tilde{D}| > a$ is safe if and only if there is no $D' \subset \tilde{D}$ with $|D'| = a$ such that D' is not safe. Therefore, $\tilde{D}_{j+1} \in \binom{C}{j+1}$ is safe if every $\tilde{D}_j \in \binom{\tilde{D}_{j+1}}{j}$ is safe. Hence, the algorithm checks for every $\tilde{D}_{j+1} \in \binom{C}{j+1}$ in $\mathcal{O}(|C|)$ time if every $\tilde{D}_j \in \binom{\tilde{D}_{j+1}}{j}$ is safe. Therefore, we can compute S_{j+1} in $\mathcal{O}(\binom{|C|}{j+1}|C|)$ time. Consequently, the algorithm checks if $S_{|C|-d} \neq \emptyset$ in $\mathcal{O}(\sum_{j=a}^{|C|-d} \binom{|C|}{j}(|C| + n + m)) \subseteq \mathcal{O}(2^{|C|}(n+m))$ time. □

6.2. Polynomial Kernels for a Family of Combined Parameters

Second, we investigate colored cut games from the viewpoint of kernelization. As shown above, natural parameterizations by $b(I)$ or $|C|$ will not give a kernel. Moreover, Colored (s,t)-Cut is NP-hard even if the vertex cover number of the input graph is at most two [31]. Hence, for most structural graph parameters there is little hope to obtain polynomial kernels. We will show that all colored cut games admit polynomial kernels when parameterized by the total attacker budget k *and* the vertex cover number. In fact, we show polynomial kernels for smaller parameters. To this end, we consider generalizations of vertex cover. For a graph G, we let $\mathrm{lp}(G)$ denote the length of a longest simple path in G. We call a vertex set $S \subseteq V$ an *r-path vertex cover* in G if $\mathrm{lp}(G - S) \leq r$ [5, 7]. Thus, an r-path vertex cover is a vertex set whose deletion results in a graph that has no simple path of length at least $r + 1$. We call the size of a smallest r-path vertex cover of a graph G the *r-path vertex cover number κ_r of G*.

Clearly, the r-path vertex cover number of G is monotonically decreasing with r. Note that the vertex cover number is exactly the 1-path vertex cover. More generally, if every connected component of a graph has order at most r, then $\mathrm{lp}(G) \leq r$. Thus, the r-path vertex cover number of a graph is never larger than the so-called *r-COC number*, the smallest size of a vertex set whose deletion results in a graph where every connected component has order at most r.

To obtain the correctness of the kernelization, we use the following generalization of colored-cut-equivalence for colored (s,t)-cuts of size at most x.

Definition 6.4. *Let x be an integer. Two colored graphs $\mathcal{H} = (G, s, t, C, \ell)$ and $\mathcal{H}' = (G', s', t', C, \ell')$ are* x-colored-cut-equivalent *if for all $\tilde{C} \subseteq C$ of size at most x it holds that $\tilde{C}$ is a colored (s,t)-cut in G if and only if $\tilde{C}$ is a colored (s',t')-cut in G'.*

Since the total attacker budget is an upper bound for the size of the colored (s,t)-cut that the attacker can choose, we obtain the following.

Corollary 6.5. *Two instances $I = (\mathcal{H}, \vec{d}, \vec{a})$ and $I' = (\mathcal{H}', \vec{d}, \vec{a})$ of any colored cut game are equivalent if $\mathcal{H}$ and $\mathcal{H}'$ are x-colored-cut-equivalent for $x \geq k$.*

Note that the total defender budget is irrelevant for this equivalence.

Now, we show that we can compute in polynomial time a k-colored-cut-equivalent graph which has at most $(k + \kappa_r)^{\mathcal{O}(r)}$ edges.

Lemma 6.6. *Let $\mathcal{H} = (G = (V, E), s, t, C, \ell)$ be a colored graph with r-path vertex cover number κ_r and let $k \leq |C|$ be an integer. Then, one can compute in $|\mathcal{H}|^{\mathcal{O}(r)}$ time a k-colored-cut-equivalent graph $\mathcal{H}' = (G' = (V', E'), s', t', C, \ell')$ with at most $\binom{(r+1)\kappa_r+2}{2} \cdot (r + 1)(r + 1)! \cdot k^{r+1}$ edges.*

The idea is the following: First, we approximate an r-path vertex cover Γ containing both s and t and compute for each pair $\{x, y\}$ of vertices of Γ the collection $A_{\{x,y\}}$ of all color sets of (x, y)-paths not containing other vertices of Γ. For each such pair, we compute the Hitting Set-instance $(A_{\{x,y\}}, k)$ and kernelize it to a Hitting Set-instance $(A'_{\{x,y\}}, k)$ with $|A'_{\{x,y\}}| < (r+1)! \cdot k^{r+1}$ by using the Sunflower Lemma [15]. Finally, we construct a colored graph $\mathcal{H}'$ such that Γ is an r-path vertex cover of G' and such that for each pair $\{x, y\}$ of vertices of Γ the collection $A'_{\{x,y\}}$ contains all color sets of (x, y)-paths not containing other vertices of Γ.

We now describe in detail how to construct $\mathcal{H}'$. First, we compute an r-path vertex cover Γ of size at most $\kappa_r(r + 1) + 2$ containing s and t via the following $(r + 1)$-approximation algorithm: Start with an empty set Γ'. While the graph $G - \Gamma'$ contains a path of length at least $r + 1$, add the $r + 1$ vertices of this path to Γ'. Afterwards, we set $\Gamma := \Gamma' \cup \{s, t\}$. By construction, Γ is an r-path vertex cover and it has size at most $\kappa_r(r + 1) + 2$ since every r-path vertex cover contains at least one vertex of each path of length at least $r + 1$.

Since $G - \Gamma$ has no paths of length at least $r + 1$, we know that every path between two vertices of Γ, which does not contain a third vertex of Γ, has at most $r + 1$ edges. We compute for every $\{a, b\} \in \binom{\Gamma}{2}$ the family of all color sets $A_{\{a,b\}}$ of (a, b)-paths in $G_{\{a,b\}} := G - (\Gamma \setminus \{a, b\})$. That is, $A_{\{a,b\}} = \mathcal{C}(\mathcal{H}_{\{a,b\}})$, where $\mathcal{H}_{\{a,b\}} := (G_{\{a,b\}}, a, b, C, \ell)$. Hence, for every color set $\tilde{C} \subseteq C$ it holds that $\tilde{C}$ is a colored (a, b)-cut in $G_{\{a,b\}}$ if and only if $\tilde{C}$ is a hitting set for $A_{\{a,b\}}$. Note that $A_{\{a,b\}}$ contains only color sets of size at most $r + 1$. Next, we reduce each of the sets $A_{\{a,b\}}$ to a size of at most $(r + 1)! \cdot k^{r+1}$ using a well-known reduction rule for $(r + 1)$-Hitting Set. This reduction rule uses the famous Sunflower Lemma [15].

Lemma 6.7. *If $A_{\{a,b\}}$ has size more than $(r + 1)! \cdot k^{r+1}$, then there are $k + 1$ distinct sets $S_1, \ldots, S_{k+1} \in A_{\{a,b\}}$ that can be computed in polynomial time such that $S_j \cap S_{j'} = \bigcap_{1 \leq i \leq k+1} S_i =: \mathcal{S}$ for all distinct $j, j' \in [1, k + 1]$.*

Reduction Rule 6.1. *If $|A_{\{a,b\}}| > (r + 1)! \cdot k^{r+1}$, then compute sets $\mathcal{S}$ and $S_1, \ldots, S_{k+1} \in A_{\{a,b\}}$ with the property of Lemma 6.7.*

- *If $\mathcal{S} = \emptyset$, then remove all sets of $A_{\{a,b\}}$ except $\{S_1, \ldots, S_{k+1}\}$.*

- *Otherwise, remove $S_1, \dots, S_{k+1}$ from $A_{\{a,b\}}$ and add the set $\mathcal{S}$.*

Next, we show that the rule is correct in the following sense.

Proposition 6.8. *Let $\tilde{C} \subseteq C$ be a set of size at most k.*

- *If $\mathcal{S} \neq \emptyset$, then $\tilde{C}$ is a hitting set for $A_{\{a,b\}}$ if and only if $\tilde{C}$ is a hitting set for $\{\mathcal{S}\} \cup (A_{\{a,b\}} \setminus \{S_i \mid 1 \leq i \leq k+1\})$.*
- *If $\mathcal{S} = \emptyset$, then $\tilde{C}$ is a hitting set for $A_{\{a,b\}}$ if and only if $\tilde{C}$ is a hitting set for $\{S_i \mid 1 \leq i \leq k+1\}$ (none of these collections has a hitting set of size at most k).*

Proof. We distinguish the case whether $\mathcal{S} = \emptyset$.

Case 1: $\mathcal{S} \neq \emptyset$. ($\Rightarrow$) Suppose that $\tilde{C}$ is a hitting set for $A_{\{a,b\}}$. It follows that $\tilde{C} \cap \mathcal{S} \neq \emptyset$, as otherwise, $\tilde{C} \cap (S_i \setminus \mathcal{S}) \neq \emptyset$ for all $i \in [1, k+1]$ which contradicts the fact that $|\tilde{C}| \leq k$. Consequently, $\tilde{C}$ is a hitting set for $\{\mathcal{S}\} \cup A_{\{a,b\}} \supseteq \{\mathcal{S}\} \cup (A_{\{a,b\}} \setminus \{S_i \mid 1 \leq i \leq k+1\})$.

($\Leftarrow$) Suppose that $\tilde{C}$ is a hitting set for $\{\mathcal{S}\} \cup (A_{\{a,b\}} \setminus \{S_i \mid 1 \leq i \leq k+1\})$. Hence, $\tilde{C} \cap \mathcal{S} \neq \emptyset$ and, thus, $\tilde{C} \cap S_i \neq \emptyset$ for all $i \in [1, k+1]$. Thus, $\tilde{C}$ is a hitting set for $\{\mathcal{S}\} \cup A_{\{a,b\}} \supseteq A_{\{a,b\}}$.

Case 2: $\mathcal{S} = \emptyset$. Since both collections $A_{\{a,b\}}$ and $\{S_i \mid 1 \leq i \leq k+1\}$ contain $k+1$ pairwise disjoint sets each, none of these collections has a hitting set of size at most k. □

Let $A'_{\{a,b\}}$ be the set obtained after exhaustively applying Reduction Rule 6.1 to $A_{\{a,b\}}$. By the definition of Reduction Rule 6.1, $A'_{\{a,b\}}$ has size at most $(r+1)! \cdot k^{r+1}$. Moreover, by the definition of $A_{\{a,b\}}$ and Proposition 6.8, we obtain that every color set $\tilde{C} \subseteq C$ of size at most k is a colored (a,b)-cut in $G_{\{a,b\}}$ if and only if $\tilde{C}$ is a hitting set for $A'_{\{a,b\}}$.

Finally, we define the colored graph $\mathcal{H}'$. We start with a graph G' containing only the vertices of Γ and set $s' = s$ and $t' = t$. Next, for every set $\{a,b\} \in \binom{\Gamma}{2}$ and every color set $L \in A'_{\{a,b\}}$, we add an (a,b)-path P_L with $\max(1, |L|-1)$ new internal vertices to G' and color the edges of P in such a way that $\ell'(P'_L) := L$, where $P'_L := a \cdot P_L \cdot b$. This finishes the definition of $\mathcal{H}'$. We may now show the correctness and running time of the data reduction and the size bound of the resulting graph $\mathcal{H}'$.

Proof of Lemma 6.6. Note that $\mathcal{C}(\mathcal{H}'_{\{a,b\}}) = A'_{\{a,b\}}$, where $G'_{\{a,b\}} := G' - (\Gamma \setminus \{a,b\})$ and $\mathcal{H}'_{\{a,b\}} := (G'_{\{a,b\}}, a, b, C, \ell')$. Hence, every color set $\tilde{C} \subseteq C$ of size at most k is a colored (a,b)-cut in $G'_{\{a,b\}}$ if and only if $\tilde{C}$ is a hitting set for $A'_{\{a,b\}}$. By the above, this is the case if and only if $\tilde{C}$ is a colored (a,b)-cut in $G_{\{a,b\}}$. Consequently, $\mathcal{H}_{\{a,b\}}$ and $\mathcal{H}'_{\{a,b\}}$ are k-colored-cut-equivalent.

Now, we use this fact to prove that $\mathcal{H}$ and $\mathcal{H}'$ are k-colored-cut-equivalent. Let $\tilde{C}$ be a colored (s,t)-cut of size at most k in G. We show that $\tilde{C}$ is a colored (s,t)-cut in G'. Assume towards a contradiction, that this is not the case.

Then, there is an (s,t)-path $P' = (u_1, \ldots, u_q)$ in G' with $u_1 = s$ and $u_q = t$ such that $\ell'(P') \cap \tilde{C} = \emptyset$. Let $u_{i_1}, \ldots, u_{i_z}$ be the vertices of Γ in P' in the ordering of the traversal of the path. Recall that $s \in \Gamma$ and $t \in \Gamma$, which implies that $u_{i_1} = u_1$ and $u_{i_z} = u_q$. Now, let $P'_j := (u_{i_j}, u_{i_j+1}, \ldots, u_{i_{(j+1)}-1}, u_{i_{(j+1)}})$ for all $j \in [1, z-1]$. Due to the fact that $\ell'(P') \cap \tilde{C} = \emptyset$, it follows that $\ell'(P'_j) \cap \tilde{C} = \emptyset$ for all $j \in [1, z-1]$. Thus, for each $j \in [1, z-1]$ it holds that $\tilde{C}$ is not a colored $(u_{i_j}, u_{i_{j+1}})$-cut in $G'_{\{u_{i_j}, u_{i_{j+1}}\}}$. Moreover, since for each $j \in [1, z-1]$, $\mathcal{H}_{\{u_{i_j}, u_{i_{j+1}}\}}$ and $\mathcal{H}'_{\{u_{i_j}, u_{i_{j+1}}\}}$ are k-colored-cut-equivalent, it follows that there is a $(u_{i_j}, u_{i_{j+1}})$-path P_j in $G_{\{u_{i_j}, u_{i_{j+1}}\}}$ such that $\ell(P_j) \cap \tilde{C} = \emptyset$. By connecting all paths $P_1 \multimap \cdots \multimap P_{z-1}$, we get an (s,t)-path P in G with $\ell(P) \cap \tilde{C} = \bigcup_{j=1}^{z-1} (\ell(P_j) \cap \tilde{C}) = \emptyset$. This contradicts the assumption that $\tilde{C}$ is a colored (s,t)-cut in G. The converse can be shown analogously.

Next, we show the running time of the construction. Since paths of length at least $r+1$ can be computed in $2^{\mathcal{O}(r)} \cdot |V|^{\mathcal{O}(1)}$ time [1], we can compute the set Γ in the same running time. Moreover, since no (a,b)-path in $G_{\{a,b\}}$ has length more than $r+2$, we can compute all the sets $A_{\{a,b\}}$ in $\mathcal{O}(\binom{|\Gamma|}{2} |V|^{r+\mathcal{O}(1)})$ time. Since each application of Reduction Rule 6.1 takes only polynomial time and reduces the size of $A_{\{a,b\}}$ by at least one, all the sets $A'_{\{a,b\}}$ can be computed in $\mathcal{O}(\binom{|\Gamma|}{2} |V|^{r+\mathcal{O}(1)})$ time as well. Thus, the construction takes $\mathcal{O}(\binom{|\Gamma|}{2} \cdot 2^{\mathcal{O}(r)} \cdot |V|^{r+\mathcal{O}(1)})$ time.

Finally, we show the size of the kernel. By construction, G' contains for every $\{a,b\} \in \binom{\Gamma}{2}$ at most $|A'_{\{a,b\}}| \leq (r+1)! \cdot k^{r+1}$ paths with at most $r+1$ edges each. Consequently, G' contains at most $\binom{|\Gamma|}{2} \cdot (r+1)(r+1)! \cdot k^{r+1}$ edges. Since $|\Gamma|$ has size at most $(r+1)\kappa_r + 2$, we obtain the stated kernel size. □

Corollary 6.5 and Lemma 6.6 lead to the following kernelization.

Theorem 6.9. *For each constant $r \geq 1$, every colored cut game admits a polynomial kernel with at most $\binom{(r+1)\kappa_r+2}{2} \cdot (r+1)(r+1)! \cdot k^{r+1}$ edges when parameterized by the r-path vertex cover number κ_r of G and the total attacker budget k.*

Proof. The bound on the number of edges follows due to Lemma 6.6. Thus, to obtain an instance of bounded size, we also have to show that we can bound the budget of the defender and the total number of colors by $(\kappa_r + k)^{\mathcal{O}(1)}$. Note that there are at most $|E'|$ colors that are assigned to at least one edge each. Let C^* be the set of exactly these colors. Since there is no edge in E' assigned with any color of $C \setminus C^*$, no winning strategy for the defender or the attacker has to contain any of the colors of $C \setminus C^*$. Hence, replace the set of colors by C^* which is as most as large as $|E'|$ and, thus, we only have to bound the budgets of the defender. If $b(I) \leq |C^*|$, then we are done. Otherwise, if $b(I) > |C^*|$, then there is a smallest index j^* such that the sum of the first j^* budgets of attacker and defender are larger than $|C^*|$. We set $d'_j := d_j$ and $a'_j := a_j$ for all $j \in [1, j^*-1], d'_r := a'_r := 0$ for all $r \in [j^*-1, i], d'_{j^*} :=$

$\min(d_{j^*}, |C^*| - \sum_{r=1}^{j^*-1}(d_r + a_r))$, and $a'_{j^*} := \max(0, |C^*| - \sum_{r=1}^{j^*-1}(d_r + a_r) - d_{j^*})$. That is, the budgets remain the same as long as the sum of the budgets so far (including the current one) is not larger than $|C^*|$. The next budget is set to the difference of $|C^*|$ and the sum of all previous budgets (including the current one) and all remaining budgets are set to zero. □

For the special case of $r = 1$, we obtain the following.

Corollary 6.10. *Every colored cut game admits a polynomial kernel with at most* $\binom{2\text{vc}+2}{2} \cdot 4k^2$ *edges when parameterized by the vertex cover number* vc *of* G *and the total attacker budget* k.

A further parameter to consider in this context is the treedepth of G [26]: The treedepth $\text{td}(G)$ of a graph is at least $\log(\text{lp}(G))$ [26]. Thus, Theorem 6.9 also implies the following result for modulators to graphs with treedepth at most r. By λ_r we denote the size of a smallest treedepth r-modulator.

Corollary 6.11. *For any constant* $r \geq 1$, *every colored cut game admits a polynomial kernel when parameterized by the size* λ_r *of a smallest treedepth* r*-modulator and the total attacker budget* k.

The size of the kernel is $\lambda_r^2 k^{\mathcal{O}(2^r)}$ and thus the guarantee is not of practical interest even for rather moderate values of k and r. However, both kernelization results are optimal in the following two ways: First, Colored (s,t)-Cut does not admit a kernel with respect to k even on graphs with treewidth two [17]. Hence, we may not replace r-path vertex cover or treedepth-r modulators by treewidth-r modulators. Moreover, the so-called standard reduction from d-Hitting Set to Colored (s,t)-Cut [25] gives graphs in which s and t are connected only via vertex-disjoint paths of length at most $d + 1$. Hence, $\text{lp}(G - \{s,t\}) \leq d - 1$ and, thus, $\kappa_{d-1} \leq 2$. Moreover, the size k of the sought colored (s,t)-cut is exactly the size of the sought hitting set of the Hitting Set instance. Thus, since d-Hitting Set does not admit a compression of bitsize $k^{d-\epsilon}$ unless NP ⊆ coNP/poly [12], Colored (s,t)-Cut does not admit a kernel of size $k^{d-\epsilon}$ even if it has an $(d-1)$-path vertex cover of size two. Since in the simple graphs produced by the reduction, we have $\text{td}(G) \in \Theta(\log \text{lp}(G))$, we can also not improve on the exponential dependence on r in the exponent of the kernel bound for treedepth.

It would also be possible to generalize the vertex cover number to the vertex deletion distance to a maximum degree of r for any $r \in \mathbb{N}$. Note that Colored (s,t)-Cut is NP-hard even when G has only two vertices of degree at least 3 [9]. Hence, Colored (s,t)-Cut parameterized by both $|C|$ and the vertex deletion distance to a maximum degree of r, for $r \geq 2$ admits a polynomial kernel if Colored (s,t)-Cut parameterized by $|C|$ alone admits a polynomial kernel. Such a kernel does not exists, unless NP ⊆ coNP/poly [17, 30].

6.3. Direct FPT-algorithms for Colored (s,t)-Cut

The kernelization algorithms of Theorem 6.9 also implies that all colored cut games admit FPT-algorithms when parameterized by $\kappa_r + k$ by simply brute-forcing on the kernel. In the following, we describe FPT-algorithms for Colored (s,t)-Cut and DA-CCV when parameterized by $\kappa_r + k$ with a better running time than a simple brute-force on the kernel. Recall that Colored (s,t)-Cut asks if there is a colored (s,t)-cut of size at most k in a given colored graph.

Theorem 6.12. *For any constant $r \geq 1$,* Colored (s,t)-Cut *can be solved in $(2^{\kappa_r}(r+1)^k + (r+1)^{\kappa_r}) \cdot n^{\mathcal{O}(r)}$ time, where κ_r denotes the r-path vertex cover number of G and the size of the sought colored (s,t)-cut is denoted by k.*

Proof. First, we compute an r-path vertex cover Γ' of size κ_r in $(r+1)^{\kappa_r} n^{\mathcal{O}(r)}$ time using a search tree algorithm that checks whether a graph contains a simple path of length $r+1$ and branches on the possibilities to destroy this path via vertex deletion. Afterwards, we check for each of the 2^{κ_r} many partitions (S,T) of $\Gamma := \Gamma' \cup \{s,t\}$ with $s \in S$ and $t \in T$, if there is a color set $\tilde{C} \subseteq C$ of size at most k such that there is no connected component containing both a vertex of S and a vertex of T after removing all the edges colored in $\tilde{C}$. To this end, we first compute for every pair of vertices $x \in S$ and $y \in T$ the collection $A_{\{x,y\}}$ of all color sets of (x,y)-paths in $G_{\{x,y\}} := G - (\Gamma \setminus \{x,y\})$. This can be done in $n^{\mathcal{O}(r)}$ time since $G_{\{x,y\}}$ does not contain any (x,y)-path of length more than $r+2$. To check if there is a color set $\tilde{C} \subseteq C$ of size at most k with the intended property, we only have to check if $\tilde{C} \cap L \neq \emptyset$ for all pairs of vertices $x \in S$ and $y \in T$ and all $L \in A_{\{x,y\}}$. This is equivalent to ask, if there is a hitting set of size at most k for $\bigcup_{(x,y)\in S\times T} A_{\{x,y\}}$. This can be determined in $(r+1)^k n^{\mathcal{O}(1)}$ time due to the fact that every $A_{\{x,y\}}$ contains only color sets of size at most $r+1$ and $(r+1)$-Hitting Set can be solved in $(r+1)^k n^{\mathcal{O}(1)}$ time. □

For $r = 1$, we obtain the following.

Corollary 6.13. Colored (s,t)-Cut *can be solved in $2^{\mathrm{vc}+k} n^{\mathcal{O}(1)}$ time, where the vertex cover number of G is denoted by* vc *and the size of the sought colored (s,t)-cut is denoted by k.*

We extend our fixed-parameter tractability result from Colored (s,t)-Cut to $(\mathrm{DA})^1$-CCV.

Theorem 6.14. *For any constant $r \geq 1$ the problem* $(\mathrm{DA})^1$-CCV *can be solved in $((2k)^{\kappa_r}(r+1)^k + (r+1)^{\kappa_r}) \cdot n^{\mathcal{O}(r)}$ time, where κ_r denotes the r-path vertex cover number of G and k denotes the budget of the attacker.*

Proof. Let $I := (G, s, t, C, \ell, d_1, a_1 = k)$ be an instance of $(\mathrm{DA})^1$-CCV. First, we compute an r-path vertex cover Γ of size κ_r in $(r+1)^{\kappa_r} n^{\mathcal{O}(r)}$ time using a search tree algorithm. Note that the length of a shortest (s,t)-path P^* in G is at most $(r+1)(\kappa_r+1)+1$: there is no path of length more than r in $G-(\Gamma' \cup \{s,t\})$ and, thus, P^* contains at most r vertices between each pair of consecutive vertices of $\Gamma' \cup \{s,t\}$. Hence, if $d_1 > (r+1)(\kappa_r+1)$, then the defender has a winning strategy by choosing the color set $\ell(P^*)$. Thus, we assume in the following that $d_1 \leq (r+1)(\kappa_r+1)$.

We describe a branching algorithm to determine if there is a set of colors D of size at most d such that every disjoint colored (s,t)-cut in G has size at least $k+1$. We start with an empty set D.

If D has size at least d_1+1, then discard the current branch. Otherwise, check if there is a colored (s,t)-cut $\tilde{C}$ of size at most k in G with $\tilde{C} \cap D = \emptyset$. This can be done in $2^{\kappa_r}(r+1)^k n^{\mathcal{O}(r)}$ time by using a modified version of the algorithm described in the proof of Theorem 6.12: instead of checking for every partition (S,T) of $\Gamma := \Gamma' \cup \{s,t\}$ if there is a hitting set of size at most k for $\bigcup_{(x,y)\in S\times T} A_{\{x,y\}}$, we check if there is a hitting set of size at most k for $\bigcup_{(x,y)\in S\times T}\{L \setminus D \mid L \in A_{\{x,y\}}\}$.

If G has no colored (s,t)-cut $\tilde{C}$ of size at most k with $\tilde{C} \cap D = \emptyset$, then D is a winning strategy for the defender. Otherwise, a winning strategy for the defender has to contain at least one of the elements of $\tilde{C}$. Hence, we branch in all $|\tilde{C}| \leq k$ different cases of adding one color of $\tilde{C}$ to D.

The branching tree has size $\mathcal{O}(k^{d_1}) \subseteq \mathcal{O}(k^{(r+1)(\kappa_r+1)})$ and, thus, the whole algorithm runs in $((2k)^{\kappa_r}(r+1)^k + (r+1)^{\kappa_r}) \cdot n^{\mathcal{O}(r)}$ time. □

7. Conclusion

We have studied the complexity of a variety of games that deal with preventing or establishing a colored cut in edge-colored graphs. The main results are summarized in Table 1. It has been observed several times, that there is a close connection between colored cuts and the Hitting Set problem. In fact, the PSPACE-hardness proof for the most general game presented in this work, is based on a simple reduction from Maker-Breaker which is essentially a competitive version of Hitting Set. Ideally, we would have liked to also use such a simple reduction for the games with a constant number of rounds. However, we do not know whether the corresponding Hitting Set games are hard. In particular, it seems open whether the following problem is Π_2^{P}-hard.

> ∀∃ Hitting Set
> **Input:** A collection $\mathcal{F}$ of subsets of a universe $\mathcal{U}$ and two integers k_1 and k_2.
> **Question:** $\forall D \in \binom{\mathcal{U}}{k_1}.\exists A \in \binom{\mathcal{U}\setminus D}{k_2}$ such that $A \cap F \neq \emptyset$ for all $F \in \mathcal{F}$?

This problem asks for a winning strategy for the attacker who wants to complete a hitting set in the case that the defender starts. If this problem is Π_2^{P}-hard, then we can infer the Π_2^{P}-hardness of $(\mathrm{DA})^1$-CCV directly from it. Otherwise, the hardness of $(\mathrm{DA})^1$-CCV would be rooted in the fact that we can create an exponential number of paths in our hardness construction.

There are several interesting directions for future research. Since most of the colored cut games are hard for higher classes of the polynomial hierarchy, it is unlikely that we can reduce them in polynomial time to ILP or SAT. This also limits the use of ILP or SAT solvers for these problems. To circumvent this obstacle, one might study whether there are *parameterized* reductions from

Table 1: Classic Complexity of Colored (s,t)-Cut, $(\mathrm{DA})^i$-CCR, $\mathrm{A(DA)}^i$-CCR, and $(\mathrm{DA})^*$-CCR in general and in some restricted graph classes.

	Colored (s,t)-Cut	$(\mathrm{DA})^i$-CCR	$\mathrm{A(DA)}^i$-CCR	$(\mathrm{DA})^*$-CCR
general	NP-c [9, 17]	Σ_{2i}^{P}-c	Π_{2i+1}^{P}-c	PSPACE-c
subcubic	$\in$ P	Σ_{2i}^{P}-c	Σ_{2i}^{P}-h	PSPACE-c
bipartite planar	NP-c [31]	Σ_{2i}^{P}-c	Π_{2i+1}^{P}-c	PSPACE-c
bipartite planar subcubic	$\in$ P	Σ_{2i}^{P}-c	Σ_{2i}^{P}-h	PSPACE-c
uncolored	$\in$ P [19, 14]	NP-h [20] NP-c if $i=1$	NP-h	NP-h
every color in ≤ 2 (s,t)-paths	$\in$ P [30]	NP-h NP-c if $i=1$	NP-h	NP-h

colored cut games to some problem in NP [21]. Moreover, for the uncolored versions of the colored cut games, only NP-hardness is known. It could be interesting to examine if for increasing number of alternations between defender and attacker, these problems are also hard for increasing levels of the polynomial hierarchy. Finally, one could investigate further parameterizations for the colored cut games. One candidate parameter could be $\mathcal{C}(\mathcal{H})$, the number of different color sets in the input graph. Here, an important first step would be to determine whether the problem of computing $\mathcal{C}(\mathcal{H})$ is fixed-parameter tractable in $|\mathcal{C}(\mathcal{H})|$.

References

[1] Alon, N., Yuster, R., Zwick, U., 1995. Color-coding. Journal of the ACM 42, 844–856.

[2] Arora, S., Barak, B., 2009. Computational Complexity - A Modern Approach. Cambridge University Press.

[3] Ben-Ari, M., 2012. Mathematical Logic for Computer Science, 3rd Edition. Springer.

[4] Bordini, A., Protti, F., da Silva, T.G., de Sousa Filho, G.F., 2019. New algorithms for the minimum coloring cut problem. International Transactions in Operational Research 26, 1868–1883.

[5] Bresar, B., Kardos, F., Katrenic, J., Semanisin, G., 2011. Minimum k-path vertex cover. Discrete Applied Mathematics 159, 1189–1195.

[6] Bruno, J., Weinberg, L., 1970. A constructive graph-theoretic solution of the Shannon switching game. IEEE Transactions on Circuit Theory 17, 74–81.

[7] Cervený, R., Suchý, O., 2019. Faster FPT algorithm for 5-path vertex cover, in: Proceedings of the 44th International Symposium on Mathematical Foundations of Computer Science (MFCS '19), Schloss Dagstuhl - Leibniz-Zentrum für Informatik. pp. 32:1–32:13.

[8] Chase, S.M., 1972. An implemented graph algorithm for winning Shannon switching games. Communications of the ACM 15, 253–256.

[9] Coudert, D., Datta, P., Perennes, S., Rivano, H., Voge, M., 2007. Shared risk resource group complexity and approximability issues. Parallel Processing Letters 17, 169–184.

[10] Coudert, D., Pérennes, S., Rivano, H., Voge, M., 2016. Combinatorial optimization in networks with shared risk link groups. Discrete Mathematics & Theoretical Computer Science 18.

[11] Cygan, M., Fomin, F.V., Kowalik, L., Lokshtanov, D., Marx, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S., 2015. Parameterized Algorithms. Springer.

[12] Dell, H., van Melkebeek, D., 2014. Satisfiability allows no nontrivial sparsification unless the polynomial-time hierarchy collapses. Journal of the ACM 61, 23:1–23:27.

[13] Downey, R.G., Fellows, M.R., 2013. Fundamentals of Parameterized Complexity. Texts in Computer Science, Springer.

[14] Edmonds, J., Karp, R.M., 1972. Theoretical improvements in algorithmic efficiency for network flow problems. Journal of the ACM 19, 248–264.

[15] Erdös, P., Rado, R., 1960. Intersection theorems for systems of sets. Journal of the London Mathematical Society 1, 85–90.

[16] Even, S., Tarjan, R.E., 1976. A combinatorial problem which is complete in polynomial space. Journal of the ACM 23, 710–719.

[17] Fellows, M.R., Guo, J., Kanj, I.A., 2010. The parameterized complexity of some minimum label problems. Journal of Computer and System Sciences 76, 727–740.

[18] Flum, J., Grohe, M., 2006. Parameterized Complexity Theory. Texts in Theoretical Computer Science. An EATCS Series, Springer.

[19] Ford, L.R., Fulkerson, D.R., 1956. Maximal flow through a network. Canadian Journal of Mathematics 8, 399–404.

[20] Grüttemeier, N., Komusiewicz, C., Morawietz, N., Sommer, F., 2021. Preventing small (s, t)-cuts by protecting edges, in: Proceedings of the 47th International Workshop on Graph-Theoretic Concepts in Computer Science (WG '21), Springer. pp. 143–155.

[21] de Haan, R., Szeider, S., 2019. A compendium of parameterized problems at higher levels of the polynomial hierarchy. Algorithms 12, 188.

[22] Klein, S., Faria, L., Sau, I., Sucupira, R., Souza, U., 2016. On colored edge cuts in graphs, in: Proceedings of the 1st Encontro de Teoria da Computaçao (ETC '16), CSBC. pp. 780–783.

[23] Mirkovic, J., Reiher, P., Papadopoulos, C., Hussain, A., Shepard, M., Berg, M., Jung, R., 2008. Testing a collaborative DDoS defense in a red team/blue team exercise. IEEE Transactions on Computers 57, 1098–1112.

[24] Morawietz, N., 2019. Computational Complexity of Network Robustness in Edge-Colored Graphs. Master's thesis. Philipps-Universität Marburg.

[25] Morawietz, N., Grüttemeier, N., Komusiewicz, C., Sommer, F., 2020. Refined parameterizations for computing colored cuts in edge-colored graphs, in: Proceedings of the 46th International Conference on Current Trends in Theory and Practice of Informatics (SOFSEM '20), Springer. pp. 248–259.

[26] Nesetril, J., de Mendez, P.O., 2006. Tree-depth, subgraph coloring and homomorphism bounds. European Journal of Combinatorics 27, 1022–1041.

[27] Niedermeier, R., 2006. Invitation to Fixed-Parameter Algorithms. Oxford University Press.

[28] Rahman, M.L., Watson, T., 2021. 6-Uniform Maker-Breaker game is PSPACE-complete, in: Proceedings of the 38th International Symposium on Theoretical Aspects of Computer Science (STACS '21), Schloss Dagstuhl - Leibniz-Zentrum für Informatik. pp. 57:1–57:15.

[29] Schaefer, T.J., 1978. On the complexity of some two-person perfect-information games. Journal of Computer and System Sciences 16, 185–225.

[30] Sucupira, R.A., 2017. Problemas de cortes de arestas maximos e mínimos em grafos. Ph.D. thesis. Universidade Federal do Rio de Janeiro.

[31] Wang, Y., Desmedt, Y., 2011. Edge-colored graphs with applications to homogeneous faults. Information Processing Letters 111, 634–641.

[32] Yuan, S., Varma, S., Jue, J.P., 2005. Minimum-color path problems for reliability in mesh networks, in: Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '05), pp. 2658–2669.

[33] Zenklusen, R., 2010. Matching interdiction. Discrete Applied Mathematics 158, 1676–1690.

[34] Zhang, P., Fu, B., 2016. The label cut problem with respect to path length and label frequency. Theoretical Computer Science 648, 72–83.