Colored Cut Games

Nils Morawietz
Philipps-Universität Marburg, Fachbereich Mathematik und Informatik, Germany
morawietz@informatik.uni-marburg.de

Niels Grüttemeier
Philipps-Universität Marburg, Fachbereich Mathematik und Informatik, Germany
niegru@informatik.uni-marburg.de

Christian Komusiewicz
Philipps-Universität Marburg, Fachbereich Mathematik und Informatik, Germany
komusiewicz@informatik.uni-marburg.de

Frank Sommer
Philipps-Universität Marburg, Fachbereich Mathematik und Informatik, Germany
fsommer@informatik.uni-marburg.de

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 $\tilde{C} \subseteq C$ 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 a family of problems called *colored cut games*. In these games, an attacker and a defender choose colors to delete and to protect, respectively, in an alternating fashion. It is the goal of the attacker to achieve a colored (s,t)-cut and the goal of the defender to prevent this. First, we show that for an unbounded number of alternations, colored cut games are PSPACE-complete. We then show that, even on subcubic graphs, colored cut games with a constant number i of 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 degree at most two. Finally, 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. In the case of $r = 1$, κ_1 is the vertex cover number vc of the input graph and we obtain a kernel with $\mathcal{O}(\text{vc}^2 k^2)$ edges. Moreover, we introduce an algorithm solving the most basic colored cut game, Colored (s,t)-Cut, in $2^{\text{vc}+k} n^{\mathcal{O}(1)}$ time.

2012 ACM Subject Classification Theory of computation → Parameterized complexity and exact algorithms; Theory of computation → Graph algorithms analysis; Theory of computation → Problems, reductions and completeness

Keywords and phrases Labeled Cut, Labeled Path, Network Robustness, Kernelization, PSPACE, Polynomial Hierarchy

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2020.30

Funding *Nils Morawietz*: Partially supported by the Deutsche Forschungsgemeinschaft (DFG), project OPERAH, KO 3669/5-1.
Frank Sommer: Supported by the Deutsche Forschungsgemeinschaft (DFG), project MAGZ, KO 3669/4-1.

Acknowledgements Some of the results of this work are also contained in the first author's Master thesis [21].

© Nils Morawietz, Niels Grüttemeier, Christian Komusiewicz, and Frank Sommer;
licensed under Creative Commons License CC-BY
40th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2020).
Editors: Nitin Saxena and Sunil Simon; Article No. 30; pp. 30:1–30:17
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

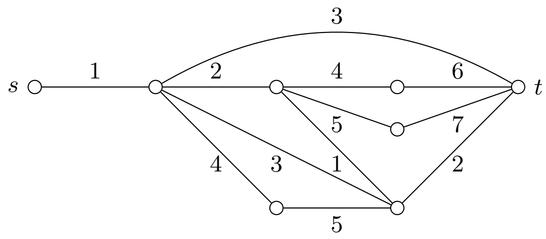

Figure 1 A colored cut game of two rounds on an edge-colored graph with seven colors: In round one, the defender may protect one color and the attacker may attack two colors. In round two, the defender can protect two colors, and the attacker can attack one color. For example, the defender may protect color 1, then the attacker may attack 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.

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 $G = (V, E)$ [12, 17]. 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 [7]. 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 given vertices depends directly on the colors of C that fail. More precisely, given $s \in V$ and $t \in V$, a 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 colored cut, called Colored (s,t)-Cut in the following, has been studied intensively [4, 7, 8, 15, 22, 28, 31]. In contrast to the classic problem on uncolored graphs, Colored (s,t)-Cut is NP-complete [7]. 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 $\tilde{C}$ [7, 15, 29]. Labeled Path is NP-complete in general [29]; when every edge color occurs at most once it is simply Shortest Path and thus solvable in polynomial time. In our scenario, Labeled Path can be seen as motivated 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 is motivated by typical studies in network security where an attacker (sometimes called red team) plays against a defender (sometimes called blue team) [20]. Such scenarios can be modelled using game-theoretic formalizations [14, 19, 25] as we do in this work. In the standard nomenclature [25], we study dynamic games with perfect information where the aim is to complete or to prevent a colored cut.

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, for each turn the attacker and the defender have a fixed budget limiting the number of colors that they may choose. We study different versions of this game, Figure 1 shows an example. 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 $(\text{DA})^*$ Colored (s,t)-Cut Robustness ($(\text{DA})^*$-CCR). The well-known Shannon Switching Game [5, 6] which is polynomial-time solvable is the special case of $(\text{DA})^*$-CCR where every edge color appears at most once and each player may choose one color in every turn.

Our Results. We study the complexity of colored cut games. In Section 3, we show that, in contrast to Shannon Switching Game, $(\text{DA})^*$-CCR is PSPACE-complete, and 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.

In Section 4.1, we study how the structure of the input graph influences the complexity of the games. We show, for example, that all colored cut games are polynomial-time solvable on graphs with degree at most two and hard for different levels of the polynomial hierarchy on bipartite planar subcubic graphs. Finally, in Section 4.2 and Section 4.3 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 is the sum of all budgets 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, k, or $|C|$) even the basic colored cut games Colored (s,t)-Cut and Labeled Path are unlikely to admit a polynomial kernelization [15, 18, 22, 31]; 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 [22]. We are not aware of other studies of kernelization for PSPACE-hard problems. In addition to the kernelization, we develop an algorithm solving Colored (s,t)-Cut in $2^{\text{vc}+k} n^{\mathcal{O}(1)}$ time. One of the main tools in our hardness proofs and algorithms is the notion of colored-cut-equivalence. This notion may be of general interest for the study of colored cuts in graphs. We define colored-cut-equivalence in Section 2, where we give the formal definition of the colored cut games. Due to lack of space, several proofs are deferred to a long version of this article.

2 Basic Definitions and Colored-Cut-Equivalence

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.

A sequence of vertices $P = (v_1, \dots, v_k)$ 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. A subset $V' \subseteq V$ is called a *connected component* of G if V' is a maximal set of vertices such that there is at least one (u,v)-path in G for pairwise distinct $u, v \in V'$.

Parameterized Complexity. For the definition of classical complexity classes such as PSPACE or Σ_2^{P}, we refer to the literature [2]. Parameterized complexity theory aims at a fine-grained analysis of the computational complexity of hard problems [9, 11, 16, 24]. A parameterized problem L is a subset of $\Sigma^* \times \mathbb{N}$, where the first component is the input and the second is the parameter. A parameterized problem is *fixed-parameter tractable (FPT)* if every instance (I,k) can be solved in $f(k) \cdot |I|^{O(1)}$ time where f is a computable function depending only on k; an algorithm with this running time is called *FPT algorithm*. A parameterized problem is in XP if every instance can be solved in $|I|^{g(k)}$ time for some computable function g. The complexity classes W[1] and W[2] are basic classes of presumed parameterized intractability, that is, it is assumed that problems that are hard for W[1] or W[2] have no fixed-parameter algorithm. Hardness for W[1] or W[2] is shown via parameterized reductions. A *parameterized reduction* of a parameterized problem L to a parameterized problem L' is an algorithm that for each instance (I,k) of L computes in $f(k) \cdot |I|^{\mathcal{O}(1)}$ time an equivalent instance (I', k') of L' such that $k' \leq g(k)$ for some computable function g. A parameterized reduction is a *polynomial parameter transformation* if $g(k)$ is a polynomial function.

A main tool to achieve fixed-parameter algorithms is *reduction to a problem kernel* or *problem kernelization*. A problem kernelization for a parameterized problem L is a polynomial-time algorithm that computes for every instance (I,k) an equivalent instance (I',k') such that $|I'| \leq g(k)$ and $k' \leq f(k)$ for computable functions f and g. If g and f are polynomials then, we call it a *polynomial problem kernelization*.

Colored Cut Games. An *edge-colored graph with terminals* (or *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 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 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: Is it true that $\exists D_1 \in \binom{C}{d_1}.\forall A_1 \in \binom{C\setminus D_1}{a_1}.\exists D_2 \in \binom{C\setminus(D_1\cup A_1)}{d_2}.\cdots.\forall A_i \in \binom{C\setminus(\bigcup_{j=1}^{i-1}(D_j\cup A_j)\cup D_i)}{a_i}$: the set $\bigcup_{j=1}^{i} A_j$ is not a colored (s,t)-cut in G?

In (DA)*-CCR we ask if the defender has a *winning strategy.* When the number of turns $i \geq 1$ is a constant and not part of the input, we define the problems $(\mathrm{DA})^i$ Colored (s,t)-Cut Robustness ($(\mathrm{DA})^i$-CCR).

If the attacker starts the game, that is, if $d_1 = 0$, we define the problems $\mathrm{A}(\mathrm{DA})^i$ Colored (s,t)-Cut Robustness ($\mathrm{A}(\mathrm{DA})^i$-CCR) for all constant $i \geq 0$. For all these problems we also define the complement problems in which we ask if there is a winning strategy for the attacker.

(DA)* Colored (s,t)-Cut Vulnerability ((DA)*-CCV)
Input: A colored graph $(G = (V,E), s, t, C, \ell)$, and two 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: Is it true that $\forall D_1 \in \binom{C}{d_1}.\exists A_1 \in \binom{C\setminus D_1}{a_1}.\forall D_2 \in \binom{C\setminus(D_1\cup A_1)}{d_2}.\cdots.\exists A_i \in \binom{C\setminus(\bigcup_{j=1}^{i-1}(D_j\cup A_j)\cup D_i)}{a_i}$: the set $\bigcup_{j=1}^{i} A_j$ is a colored (s,t)-cut in G?

Analogously, if the number of alternations is a constant, then we define the variants $(\mathrm{DA})^i$-CCV and $\mathrm{A}(\mathrm{DA})^i$-CCV. We refer to all problems defined above as *colored cut games.*

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

Colored-Cut-Equivalence. 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.

▶ **Observation 2.1.** *The set of colors $\tilde{C} \subseteq C$ is a colored (s,t)-cut in G if and only if $\tilde{C}$ is a hitting set for $\mathcal{C}(\mathcal{H})$, that is, if $\tilde{C} \cap C' \neq \emptyset$ for all $C' \in \mathcal{C}(\mathcal{H})$.*

Moreover, $\tilde{C}$ is a colored (s,t)-connector in G if and only if there is $C' \in \mathcal{C}(\mathcal{H})$ such that $C' \subseteq \tilde{C}$.

To argue concisely that two instances of some 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 $L_2 \subseteq L_1$.*

Observe 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. Thus, intuitively, only the color sets in $\mathcal{C}(\mathcal{H}) \cap \mathcal{C}(\mathcal{H}')$ are relevant for colored (s,t)-cuts. 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, we only show one direction. Let $\tilde{C}$ be a colored (s,t)-cut in G, then $\tilde{C} \cap L_2 \neq \emptyset$ 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'. ◀

▶ **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.

▶ **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 $\mathcal{H}' = (G',s',t',C,\ell')$ such that G' is bipartite.*

▶ **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 $\mathcal{H}$. 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 three.*

3 Classic Complexity of Colored Cut Games

3.1 Unbounded Number of Alternations

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 Competitive Hitting Set [26].

▶ **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 obviously be solved within polynomial space by a standard 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 [26].

Competitive Hitting Set (CHS)
Input: A universe $\mathcal{U}$ with $|\mathcal{U}| = 2i$ and a collection $\mathcal{F}$ of non-empty subsets of $\mathcal{U}$.
Question: Is it true that $\forall d_1 \in \mathcal{U}. \exists a_1 \in \mathcal{U} \setminus \{d_1\}. \forall d_2 \in \mathcal{U} \setminus \{d_1, a_1\}. \cdots . \exists a_i \in \mathcal{U} \setminus \left(\bigcup_{j=1}^{i-1} \{d_j, a_j\} \cup \{d_i\} \right) : F \cap \{a_j \mid 1 \leq j \leq i\} \neq \emptyset$ 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 there is no unselected element of the universe remaining and the second player wins if he intersects every subset $F \in \mathcal{F}$ with the elements he chose. Otherwise, the first player wins. We ask if the second player has a winning strategy.

Given an instance $I = (\mathcal{U}, \mathcal{F})$ of Competitive Hitting Set, we describe how to construct an equivalent instance $I' = (G = (V, E), s, t, C, \ell)$ of $(\text{DA})^*$-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 $F \in \mathcal{F}$ 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 $(\text{DA})^*$-CCV instance I' is also a winning strategy for the second player in the Competitive Hitting Set instance I and vice versa. Therefore, I is a yes-instance of Competitive Hitting Set if and only if I' is a yes-instance of $(\text{DA})^*$-CCV. Since the class of PSPACE-complete problems is closed under complement, $(\text{DA})^*$-CCR where the budget in every turn is one is also PSPACE-complete. ◀

3.2 Bounded Number of Alternations

Next, we analyze the complexity of $(\text{DA})^i$-CCR and $\text{A}(\text{DA})^i$-CCR. To this end, recall that $(\text{DA})^i$-CCR asks if the defender has a winning strategy when the defender starts and both agents have exactly i turns for some constant i.

▶ **Lemma 3.2.** *For all $i \geq 1$, $(DA)^i$-CCV is Π^{P}_{2i}-hard and $(DA)^i$-CCR is Σ^{P}_{2i}-hard even on planar graphs.*

To prove Lemma 3.2, we reduce QSAT_{2i} 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 $\phi \in \Phi$.

Proof sketch. We reduce QSAT_{2i}, which is Π^{P}_{2i}-hard [2], to $(\text{DA})^i$-CCV.

> QSAT_{2i}
> **Input:** A set Φ of clauses in 3-CNF over the set of 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.

Given an instance $I' = (Z, \Phi, X_1, Y_1, \ldots, X_i, Y_i)$ of QSAT_{2i}, we construct an instance $I = (\mathcal{H}, \vec{d}, \vec{a})$ of $(\text{DA})^i$-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 $1 \leq j \leq i$ 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, E_d and V_a, E_a are the variable gadgets for the defender and attacker, respectively, and V_Φ, E_Φ is the gadget

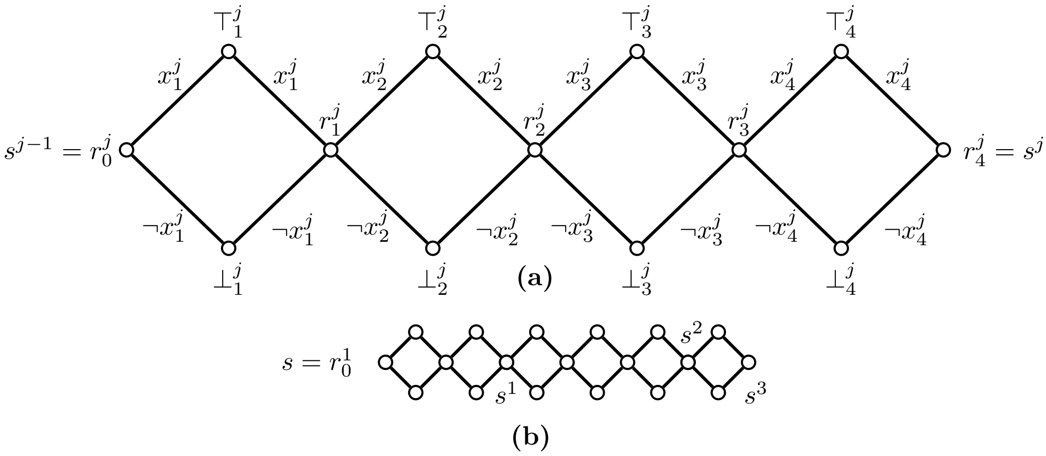

Figure 2 (a) The gadget for the defender for the variables of X_j with $|X_j| = 4$. (b) The graph $G_D = (V_D, E_D)$ where $|X_1| = 2$, $|X_2| = 3$, and $|X_3| = 1$.

for the evaluation of the clauses. First, we introduce the variable gadget for the defender, shown in Figure 2:

- $V_d := \{r_0^j \mid 1 \leq j \leq i\} \cup \{r_k^j, \top_k^j, \bot_k^j \mid 1 \leq j \leq i, 1 \leq k \leq |X_j|\}$
- $E_d := \Big\{\{r_{k-1}^j, \top_k^j\}, \{r_{k-1}^j, \bot_k^j\}, \{\top_k^j, r_k^j\}, \{\bot_k^j, r_k^j\} \mid 1 \leq j \leq i, 1 \leq k \leq |X_j|\Big\}$,
- $\ell(\{r_{k-1}^j, \top_k^j\}) := \ell(\{\top_k^j, r_k^j\}) := x_k^j$,
- $\ell(\{r_{k-1}^j, \bot_k^j\}) := \ell(\{\bot_k^j, r_k^j\}) := \neg x_k^j$,

where $r_{|X_j|}^j = r_0^{j+1}$ for all $1 \leq j < i$. In the following, let $s := s^0 := r_0^1$ and $s^j := r_{|X_j|}^j$ 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 in his jth turn the defender has to choose 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:

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

The idea is that either the color set chosen by the attacker in his jth turn is an assignment of the variables of $X_j \cup Y_j$, or 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 clause gadget. To model each clause $\phi \in \Phi$, we add an (s^i, t)-path P with $\ell(P) = \phi$. Formally, the gadget is defined as follows. We fix an ordering on every clause $\phi_j \in \Phi$ and denote with $\phi_j(y)$ the yth literal of ϕ_j and add

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

The final graph can be seen in Figure 3. We set $d_j := |X_j|$ and $a_j := |X_j| + |Y_j|$ for all $j \in [1, i]$. This completes the construction.

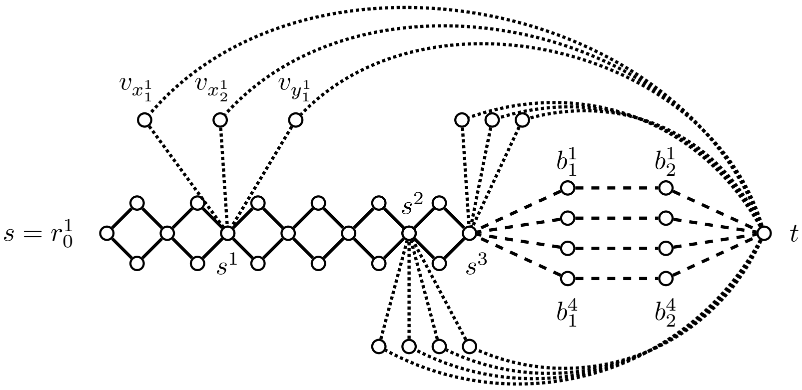

Figure 3 The construction for an instance with $|\Phi| = 4$, $|X_1| = |Y_3| = 2$, $|Y_1| = |Y_2| = |X_3| = 1$, and $|X_2| = 3$. Solid edges belong to E_d, dotted edges belong to E_a, dashed edges belong to E_Φ. The clause gadget is connected with s^3 and t.

Before we show the equivalence between I and I', we make some observations about winning strategies. The following establishes the link between sensible choices of color sets and partial assignments for variables in Z: Let $j \in [1, i]$ and let $D_j \subseteq C$ be the set of colors the defender chooses in his jth turn. We call D_j *nice* if D_j is an assignment for X_j. Analogously, let $A_j \subseteq C$ be the set of colors the attacker chooses in his jth turn. We call A_j *nice* if A_j is an assignment for $X_j \cup Y_j$.

▷ Claim 3.3. For both players, it is never part of a winning strategy to be the first to choose a set of colors which is not nice.

Hence, we can assume that both players will only choose nice sets of colors.

▷ Claim 3.4. 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 $\tilde{A} \models \Phi$.

Using these claims, we show that the QSAT_{2i} instance is a yes-instance if and only if the constructed $(\text{DA})^i$-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, $\mathbb{P}$ denotes the powerset. The functions $f_1, \ldots, f_i$ 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 $(\text{DA})^i$-CCV instance iteratively. Let D_1 be the color set chosen by the defender in his first turn. If D_1 is not nice then, due to Claim 3.3, 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.

After this initial choice, the winning strategy for the attacker works as follows. 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 color set chosen by the defender in his jth turn. If D_j is not nice then, due to Claim 3.3, 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. Hence, we can assume that D_j is nice and A_j is nice and defined as described for all $j \in [1, i]$.

It remains to show that $\tilde{A}_i := \bigcup_{j=1}^{i} A_j$ is a colored (s,t)-cut in G. 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 due to Claim 3.4. Hence, the attacker has a winning strategy.

($\Leftarrow$) The proof of this direction is deferred to the the long version of this article.

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. ◀

Lemma 3.2 is the main step to prove the following.

▶ **Theorem 3.5.** *For all $i \geq 0$, $A(DA)^i$-CCR is Π_{2i+1}^{P}-complete and for all $i \geq 1$, $(DA)^i$-CCR is Σ_{2i}^{P}-complete even on planar graphs.*

4 Restricted Instances and Parameterizations

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 obtain a complexity dichotomy with regard to the maximum degree and strengthen our hardness results from Section 3.1 to restricted graph classes. Second, we analyze a restricted class 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. Finally, we investigate the parameterized complexity and describe how to obtain polynomial kernel for all colored cut games by combining the budget with structural graph parameters.

4.1 Restricted Instances

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 two but cannot be solved in polynomial time on graphs with maximum degree at least three, unless $\mathrm{P} = \mathrm{NP}$.

By Theorem 3.5, $(\mathrm{DA})^i$-CCV and $\mathrm{A}(\mathrm{DA})^i$-CCV are hard even on planar graphs. Given a planar graph, we can replace it with a bipartite planar colored-cut-equivalent graph in polynomial time due to Lemma 2.4. By Corollary 2.3, this gives an equivalent instance.

▶ **Corollary 4.1.** *For all $i \geq 1$, $(DA)^i$-CCV is Π_{2i}^{P}-complete and for all $i \geq 0$, $A(DA)^i$-CCV is $\Sigma_{2i+1}^{\mathrm{P}}$-complete even on bipartite planar graphs.*

▶ **Theorem 4.2.** *Let $i \geq 1$. The problems $(DA)^i$-CCR, $A(DA)^i$-CCR, and $(DA)^*$-CCR can be solved in polynomial time on graphs with a maximum degree of at most two. On bipartite planar graphs with a maximum degree of at least three, $(DA)^i$-CCR and $A(DA)^i$-CCR are Σ_{2i}^{P}-hard and $(DA)^*$-CCR is* PSPACE*-hard.*

Second, we analyze the complexity of $(\mathrm{DA})^1$-CCR on instances where every color appears in at most two (s,t)-paths. In this case, Colored (s,t)-Cut can be solved in polynomial time [7, 17, 27]. In contrast, we will show that $(\mathrm{DA})^1$-CCR is NP-complete. Hence, for any $i \geq 1$, $(\mathrm{DA})^i$-CCR and $\mathrm{A}(\mathrm{DA})^i$-CCR cannot be solved in polynomial time on these restricted colored graphs, unless $\mathrm{P} = \mathrm{NP}$. We show NP-completeness via reduction from Matching Interdiction which is NP-hard [30].

▶ **Theorem 4.3.** *$(DA)^1$-CCR is* NP*-complete and* W[1]*-hard when parameterized by d_1 even if every color appears in at most two (s,t)-paths.*

4.2 Parameterization by the Full Budget and the Number of Colors

In this section we analyze the parameterized complexity of the colored cut games. Next, we investigate budget-related parameters. 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 $k = b(I)$ [7]. We extend this hardness result to all colored cut games. Moreover, we show that all colored cut games are fixed-parameter tractable and do not admit polynomial kernels when parameterized $|C|$.

▶ **Proposition 4.4.** *$(DA)^i$-CCR, $i \geq 1$, $A(DA)^i$-CCR, $i \geq 0$, and $(DA)^*$-CCR parameterized by $b(I)$ are* coW[2]*-hard and can be solved in $\mathcal{O}(|C|^{b(I)}(n+m))$ time.*

By definition, $b(I) \leq |C|$. Hence, the described algorithm of Proposition 4.4 with a running time of $\mathcal{O}(|C|^{b(I)}(n+m))$ also implies an FPT-algorithm when parameterized by $|C|$.

▶ **Corollary 4.5.** *$(DA)^i$-CCR, $A(DA)^{i-1}$-CCR, $i \geq 1$, and $(DA)^*$-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.

4.3 Polynomial Kernels by Combining Budget with Structural Graph Parameters

Finally, we investigate colored cut games from the viewpoint of kernelization. By the above, natural parameterizations by $b(I)$ or even $|C|$ 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 [28]. Hence, for most structural graph parameters there is little hope to obtain polynomial kernels. We will show that, however, 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 covers.

▶ **Definition 4.6.** *For a graph G, we let $\text{lp}(G)$ denote the length of a longest path in G. We call a vertex set $S \subseteq V$ an* r-lp-modulator *in G if $\text{lp}(G - S) \leq r$. The size of a smallest r-lp-modulator of a graph G is the* r-lp-deletion number κ_r of G.

Thus, an r-lp-modulator is a vertex set whose deletion results in a graph that has no simple paths of length at least $r+1$. Clearly, the r-lp-deletion number of G is monotonically decreasing with r. Note that the vertex cover number is exactly the 1-lp-deletion number. More generally, if every connected component of a graph has order at most r, then $\text{lp}(G) \leq r$. Thus, the r-lp-deletion 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 show the correctness of the kernelization, we need to argue that an attacker can achieve a colored cut in the kernel if and only if he can achieve it in the input instance. Thus, we only need to consider colored cuts of bounded size in the correctness proof. Motivated by this, we generalize the notion of colored-cut-equivalence as follows.

▶ **Definition 4.7.** *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 the attacker can choose, we obtain the following.

▶ **Corollary 4.8.** *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 k-colored-cut-equivalent where $k = \sum_{x=1}^{i} a_x$.*

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

▶ **Lemma 4.9.** *Let $\mathcal{H} = (G = (V,E), s, t, C, \ell)$ be a colored graph with r-lp-deletion number κ_r and let $k \le |C|$ 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 $\binom{(r+1)\kappa_r+2}{2} \cdot (r+1)(r+1)!k^{r+1}$ edges.*

The idea of the algorithm is the following: First, we approximate an r-lp-modulator 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 $|A'_{\{x,y\}}| < (r+1)!k^{r+1}$ by using the Sunflower Lemma [13]. Finally, we construct a colored graph $\mathcal{H}'$ such that Γ is an r-lp-modulator of G' and such that for each pair $\{x,y\}$ of vertices of Γ, the collection of all color sets of (x,y)-paths not containing other vertices of Γ is precisely $A'_{\{x,y\}}$. This can be done with $|A'_{\{x,y\}}|$ paths for each $A'_{\{x,y\}}$. Hence, the resulting graph has bounded size.

We now describe in detail how to construct $\mathcal{H}'$. First, we compute an r-lp-modulator 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-lp-modulator and it has size at most $\kappa_r(r+1)+2$ since every r-lp-modulator 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 [13].

▶ **Lemma 4.10.** *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 $S_j \cap S_{j'} = \bigcap_{1 \le i \le k+1} S_i =: \mathcal{S}$ for all distinct $j, j' \in [1, k+1]$.*

▶ **Rule 4.1.** *If $|A_{\{a,b\}}| > (r+1)! \cdot k^{r+1}$, then compute sets $S_1, \ldots, S_{k+1} \in A_{\{a,b\}}$ and $\mathcal{S}$ with the property of Lemma 4.10.*

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

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

▶ **Proposition 4.11.** *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 $\{\mathcal{S}\} \cup (A_{\{a,b\}} \setminus \{S_i \mid 1 \le i \le k+1\})$.*
- *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 $\{S_i \mid 1 \le i \le k+1\}$.*

Let $A'_{\{a,b\}}$ be the set obtained after exhaustively applying Rule 4.1 to $A_{\{a,b\}}$. By the definition of Rule 4.1, $A'_{\{a,b\}}$ has size at most $(r+1)! \cdot k^{r+1}$. Moreover, by the definition of $A_{\{a,b\}}$ and Proposition 4.11, 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 the running time of the data reduction and the size bound of the resulting graph $\mathcal{H}'$.

Proof of Lemma 4.9. 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, 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 $A'_{\{a,b\}}$. 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 $P'_j := (u_{i_j}, u_{i_j+1}, \ldots, u_{i_{(j+1)}-1}, u_{i_{(j+1)}})$ denote the subpath of P' connecting u_{i_j} and $u_{i_{j+1}}$ 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 an $(u_{i_j}, u_{i_{j+1}})$-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 opposite direction 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 $\mathcal{O}(\binom{|\Gamma|}{2} \cdot |V|^{r+\mathcal{O}(1)})$ time. Since each application of Rule 4.1 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 $\mathcal{O}(\binom{|\Gamma|}{2} \cdot |V|^{r+\mathcal{O}(1)})$ time as well. Thus, the complete 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 $|A'_{\{a,b\}}| \leq (r+1)!k^{r+1}$ paths with at most $r+1$ edges each. Consequently, G' contains at most $\binom{|\Gamma|}{2} \cdot (r+1)(r+1)!k^{r+1}$ edges. Since $|\Gamma|$ has size at most $(r+1)\kappa_r + 2$, we obtain the stated kernel size. ◀

Corollary 4.8 and Lemma 4.9 lead to the following kernelization.

▶ **Theorem 4.12.** *For each constant $r \geq 1$, every colored cut game admits a polynomial kernel with at most $\binom{(r+1)\kappa_r+2}{2} \cdot (r+1)(r+1)!k^{r+1}$ edges when parameterized by the r-lp-deletion number κ_r of G and the total attacker budget k.*

▶ **Corollary 4.13.** *Every colored cut game admits a polynomial kernel with at most* $\binom{2\mathrm{vc}+2}{2}\cdot 4k^2$ *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 [23]: The treedepth $\mathrm{td}(G)$ of a graph is at least $\log(\mathrm{lp}(G))$ [23]. Thus, Theorem 4.12 also implies the following result for modulators to graphs with treedepth at most r. Herein λ_r denotes the size of a smallest treedepth r-modulator.

▶ **Corollary 4.14.** *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 the treedepth bound r. However, these 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 [15]. Hence, we may not replace r-lp-modulators or treedepth-r modulators by treewidth-r modulators. Moreover, the standard reduction from $(r+1)$-Hitting Set to Colored (s,t)-Cut gives graphs in which s and t are connected only via vertex disjoint paths of length at most $r+2$. Hence, $\mathrm{lp}(G - \{s,t\}) \leq r$ and, thus, $\kappa_r \leq 2$. Moreover, k is exactly the budget of the Hitting Set instance. Thus, since $(r+1)$-Hitting Set does not admit a compression of bitsize $k^{r+1-\epsilon}$ unless $\mathrm{NP} \subseteq \mathrm{coNP/poly}$ [10], Colored (s,t)-Cut does not admit a kernel of size $k^{r+1-\epsilon}$ even if it has a r-lp-deletion number of size two. Since in these simple graphs produced by the reduction, we have $\mathrm{td}(G) \in \Theta(\log \mathrm{lp}(G))$, we can also not improve on the doubly exponential dependence on r in the exponent of the kernelization for treedepth.

Based on these kernel results, it also follows that all colored cut games admit FPT-algorithms when parameterized by $\kappa_r + k$. 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.

▶ **Theorem 4.15.** *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-lp-*deletion number of* G *and* k *denotes the budget of the attacker.*

Proof. First, we compute an r-lp-modulator 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 the question, if there is a hitting set of size at most k for $\bigcup_{(x,y)\in S\times T} A_{\{x,y\}}$, which can be determined in $(r+1)^k n^{\mathcal{O}(1)}$ time due to the fact that every set $A_{\{x,y\}}$ 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. ◀

▶ **Corollary 4.16.** Colored (s,t)-Cut *can be solved in* $2^{\mathrm{vc}+k} n^{\mathcal{O}(1)}$ *time, where* vc *denotes the vertex cover number of* G *and* k *denotes the budget of the attacker.*

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 cases.

graph classes	COLORED (s,t)-CUT	$(\mathrm{DA})^i$-CCR	$\mathrm{A(DA)}^i$-CCR	$(\mathrm{DA})^*$-CCR
general	NP-c [7, 15]	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 [28]	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
every color in ≤ 2 (s,t)-paths	$\in$ P [27]	NP-h NP-c if $i = 1$	NP-h	NP-h

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

▶ **Theorem 4.17.** *For any constant $r \geq 1$, $(DA)^1$-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-lp-deletion number of G and k denotes the budget of the attacker.*

Let us remark that it would also be natural to attempt to generalize the vertex cover number to the vertex deletion distance to a maximum degree of r for any $r \in \mathbb{N}$. Note, however, that the standard reduction from HITTING SET to COLORED (s, t)-CUT [7] already implies that COLORED (s, t)-CUT has no kernel of size $|C|^{\mathcal{O}(1)}$ even when G has only two vertices of degree at least three, unless NP $\subseteq$ coNP/poly. Hence, for any $r \geq 2$ it is unlikely that we can obtain polynomial kernels for $|C|$ plus the vertex deletion distance to a maximum degree of r.

5 Conclusion

We have studied the complexity of a variety of games that deal with preventing or establishing a colored cut in edge-colored graphs (see Table 1 for an overview of the classic complexity results). In the negative and the positive results of this work we exploited the close connection between colored cut games and the HITTING SET problem. For example, the PSPACE-hardness proof for the most general game presented in this work, is based on a simple reduction from COMPETITIVE 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.

$\forall\exists$ 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. It would also be interesting to explore further how efficiently we can reduce from colored cut games to HITTING SET. In other words, how long does it take to construct $\mathcal{C}(\mathcal{H})$, the collection of color sets of (s, t)-paths, for a given colored graph $\mathcal{H}$? In particular, can we compute the set $\mathcal{C}(\mathcal{H})$ in $|\mathcal{C}(\mathcal{H})| \cdot |\mathcal{H}|^{\mathcal{O}(1)}$ time?

References

1. Noga Alon, Raphael Yuster, and Uri Zwick. Color-coding. *Journal of the ACM*, 42(4):844–856, 1995.
2. Sanjeev Arora and Boaz Barak. *Computational Complexity - A Modern Approach*. Cambridge University Press, 2009.
3. Mordechai Ben-Ari. *Mathematical Logic for Computer Science, 3rd Edition*. Springer, 2012.
4. Augusto Bordini, Fábio Protti, Thiago Gouveia da Silva, and Gilberto Farias de Sousa Filho. New algorithms for the minimum coloring cut problem. *International Transactions in Operational Research*, 26(5):1868–1883, 2019.
5. John Bruno and Louis Weinberg. A constructive graph-theoretic solution of the Shannon switching game. *IEEE Transactions on Circuit Theory*, 17(1):74–81, 1970.
6. Stephen M. Chase. An implemented graph algorithm for winning Shannon switching games. *Communications of the ACM*, 15(4):253–256, 1972.
7. David Coudert, Pallab Datta, Stephane Perennes, Hervé Rivano, and Marie-Emilie Voge. Shared risk resource group complexity and approximability issues. *Parallel Processing Letters*, 17(2):169–184, 2007.
8. David Coudert, Stéphane Pérennes, Hervé Rivano, and Marie-Emilie Voge. Combinatorial optimization in networks with shared risk link groups. *Discrete Mathematics & Theoretical Computer Science*, 18(3), 2016.
9. Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015.
10. Holger Dell and Dieter van Melkebeek. Satisfiability allows no nontrivial sparsification unless the polynomial-time hierarchy collapses. *Journal of the ACM*, 61(4):23:1–23:27, 2014.
11. Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.
12. Jack Edmonds and Richard M. Karp. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the ACM*, 19(2):248–264, 1972.
13. Paul Erdös and Richard Rado. Intersection theorems for systems of sets. *Journal of the London Mathematical Society*, 1(1):85–90, 1960.
14. Seyed Rasoul Etesami and Tamer Basar. Dynamic games in cyber-physical security: An overview. *Dynamic Games and Applications*, 9(4):884–913, 2019.
15. Michael R. Fellows, Jiong Guo, and Iyad A. Kanj. The parameterized complexity of some minimum label problems. *Journal of Computer and System Sciences*, 76(8):727–740, 2010.
16. Jörg Flum and Martin Grohe. *Parameterized Complexity Theory*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2006.
17. L. R. Ford and D. R. Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 8:399–404, 1956.
18. Sulamita Klein, Luerbio Faria, Ignasi Sau, Rubens Sucupira, and Uéverton Souza. On colored edge cuts in graphs. In *Proceedings of the 1st Encontro de Teoria da Computaçao (ETC '16)*, Sociedade Brasileira de Computaçao, pages 780–783. CSBC, 2016.
19. Xiannuan Liang and Yang Xiao. Game theory for network security. *IEEE Communications Surveys & Tutorials*, 15(1):472–486, 2013.
20. Jelena Mirkovic, Peter Reiher, Christos Papadopoulos, Alefiya Hussain, Marla Shepard, Michael Berg, and Robert Jung. Testing a collaborative ddos defense in a red team/blue team exercise. *IEEE Transactions on Computers*, 57(8):1098–1112, 2008.
21. Nils Morawietz. Computational complexity of network robustness in edge-colored graphs. Master's thesis, Philipps-Universität Marburg, 2019.
22. Nils Morawietz, Niels Grüttemeier, Christian Komusiewicz, and Frank Sommer. 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)*, volume 12011 of *LNCS*, pages 248–259. Springer, 2020.

23 Jaroslav Nesetril and Patrice Ossona de Mendez. Tree-depth, subgraph coloring and homomorphism bounds. *European Journal of Combinatorics*, 27(6):1022–1041, 2006.
24 Rolf Niedermeier. *Invitation to Fixed-Parameter Algorithms*. Oxford University Press, 2006.
25 Sankardas Roy, Charles Ellis, Sajjan G. Shiva, Dipankar Dasgupta, Vivek Shandilya, and Qishi Wu. A survey of game theory as applied to network security. In *Proceedings of the 43rd Hawaii International International Conference on Systems Science (HICSS '10)*, pages 1–10. IEEE Computer Society, 2010.
26 Thomas J. Schaefer. On the complexity of some two-person perfect-information games. *Journal of Computer and System Sciences*, 16(2):185–225, 1978.
27 Rubens André Sucupira. *Problemas de cortes de arestas maximos e mínimos em grafos*. PhD thesis, Universidade Federal do Rio de Janeiro, 2017.
28 Yongge Wang and Yvo Desmedt. Edge-colored graphs with applications to homogeneous faults. *Information Processing Letters*, 111(13):634–641, 2011.
29 Shengli Yuan, Saket Varma, and Jason P. Jue. 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)*, pages 2658–2669, 2005.
30 Rico Zenklusen. Matching interdiction. *Discrete Applied Mathematics*, 158(15):1676–1690, 2010.
31 Peng Zhang and Bin Fu. The label cut problem with respect to path length and label frequency. *Theoretical Computer Science*, 648:72–83, 2016.