Philipps Universität Marburg

Fachbereich Mathematik & Informatik
AG Algorithmik
Prof. Dr. C. Komusiewicz

Wintersemester 2019

Master-Arbeit zum Thema

Computational Complexity of Network Robustness in Edge-Colored Graphs

Autor: Nils Morawietz
Betreuer: Prof. Dr. C. Komusiewicz

Email: morawietz@informatik.uni-marburg.de
Datum der Abgabe: 23. September 2019

Hiermit versichere ich, Nils Morawietz, dass ich die vorliegende Arbeit mit dem Titel “Computational Complexity of Network Robustness in Edge-Colored Graphs” selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.
Die Master-Arbeit wurde in der jetzigen oder einer ähnlichen Form noch bei keiner anderen Hochschule eingereicht und hat noch keinen sonstigen Prüfungszwecken gedient.

Marburg, den 23.09.2019

Danksagung

An dieser Stelle möchte ich allen Menschen meinen Dank aussprechen, durch deren Einfluss ich im Stande war diese Arbeit zu verfassen.

Dies betrifft im Speziellen meinen Betreuer, Christian Komusiewicz, der mir nahezu rund um die Uhr mit Rat und Tat zur Seite stand und selbst an zeitlich eng bemessenen Tagen die Zeit aufgebracht hat, um auf meine Fragen einzugehen. Zudem, möchte ich auch den Doktoranden Frank Sommer und Niels Grüttemeier meinen Dank aussprechen, die mir bei Fragen ebenfalls stets zur Verfügung gestanden haben.

Weiterhin möchte ich meiner Familie und meinen Freunden danken, die mich von jeher unterstützt haben. Dies betrifft vor allem meine Eltern die selbst in schweren Zeiten an mich geglaubt und mich dazu animiert haben, meinen Weg weiter zu gehen.

Abschließend gebührt mein Dank der Adolf-Reichwein-Schule Marburg und ihren Lehrkräften, die meine Freude am Lernen geweckt haben. Insbesondere möchte ich Herrn Jürgen Lederer hervorheben, der mich mit seiner unvergleichbaren Art im Unterricht für die Informatik begeistern konnte.

Abstract

Edge-colored graphs can be used to model various network structures in network security analysis. One of the decision problems which is important in the analysis of edge-colored graphs in this scenario is the COLORED (s,t)-CUT problem, defined as follows. Given an edge-colored graph, two vertices s and t, and an integer k, we ask if there is a size-k set of colors S, such that the removal of all edges colored with a color of S disconnects s from t. Such a set S is called a *colored* (s,t)*-cut*. In this work, we further investigate the parameterized complexity of COLORED (s,t)-CUT for some structural graph parameters. Afterwards, we generalize COLORED (s,t)-CUT to defender-attacker games where a defender and an attacker alternatively choose unchosen colors. The edges colored in the colors chosen by the attacker, are removed from the graph, whereas, the colors chosen by the defender cannot be chosen by the attacker anymore. The attacker wins if he can complete a colored (s,t)-cut. We show that for a constant number of alternations between the agents, these games are complete for complexity classes of different levels of the polynomial-time hierarchy, whereas, the games become PSPACE-complete if the number of alternations is unbounded. We then investigate these games from a parameterized complexity point of view. For example we show that all these games admit polynomial kernel when parameterized by both the number of colors in the instance and the vertex cover number. Finally, we study the classic complexity of these games on restricted instances. For example, we show that on some restricted instances on which COLORED (s,t)-CUT is polynomial-time-solvable, none of the introduced games can be solved in polynomial time, unless P = NP.

Zusammenfassung

Kantengefärbte Graphen eignen sich zur Modellierung verschiedenster Netzwerkstrukturen im Rahmen von Netzwerksicherheitsanalysen. Ein wichtiges Entscheidungsproblem im auf kantengefärbter Graphen in diesem Kontext ist COLORED (s,t)-CUT definiert wie folgt. Gegeben sind ein kantengefärbter Graph, zwei Knoten s und t und eine natürliche Zahl k und wir wollen wissen, ob es eine k-elementige Farbmenge gibt, sodass das Entfernen aller Kanten, die in einer Farbe aus S gefärbt sind, s von t trennt. Eine solche Menge S wird *colored* (s,t)*-cut* genannt. In dieser Arbeit führen wir die parametrisierte Analyse von COLORED (s,t)-CUT für strukturelle Graphparameter fort. Anschließend verallgemeinern wir COLORED (s,t)-CUT zu

Verteidiger-Angreifer-Spielen, bei denen ein Verteidiger und ein Angreifer abwechselnd bisher noch ungewählte Farben wählen. Kanten, die in den Farben gefärbt sind, die der Angreifer wählt, werden aus dem Graphen entfernt, wohingegen die Farben, die der Verteidiger wählt, nicht mehr vom Angreifer gewählt werden können. Der Angreifer gewinnt das Spiel, wenn er einen colored (s,t)-cut wählt. Wir zeigen, dass für eine konstante Anzahl an Wechseln zwischen den beiden Agenten, die definierten Spiele für Komplexitätsklassen unterschiedlicher Stufen der Polynomialzeit-Hierarchie vollständig sind. Wenn die Anzahl der Wechsel nicht konstant ist, sind die definerten Spiele hingegen PSPACE-vollständig. Anschließend analysieren wir diese Spiele aus Sicht der parametrisierten Komplexitätstheorie. Zum Beispiel zeigen wir, dass all diese Probleme einen polynomiellen Kern haben, wenn man sie sowohl mit der Anzahl der Farben als auch der Vertex-Cover-Zahl parametrisiert. Abschließend untersuchen wir die klassische Komplexität aller Spiele auf eingeschränkten Instanzen. Zum Beispiel zeigen wir, dass auf einigen eingeschränkten Instanzen auf denen Colored (s,t)-Cut polynomialzeitlösbar ist, keines der Spiele polynomialzeitlösbar ist, es sei denn $\mathrm{P} = \mathrm{NP}$.

Contents

1 Introduction

Due to the ever-growing impact of computer and communication networks, analyzing their robustness becomes inevitable [33, 34]. Edge-colored graphs are a useful tool to model different kinds of network structures for network-security analysis. For example in the context of measuring the robustness of communication, transportation, and computer networks, edge-colored graphs where studied extensively over the last decades [5, 6, 8, 10, 11, 19, 25–27, 29, 38–40]. One of the decision problems which can be used to measure the robustness of such networks is Colored (s, t)-Cut, defined as follows.

> Colored (s, t)-Cut [10]
> **Input**: An undirected graph $G = (V, E)$, two vertices $s, t \in V$, a set of colors C, an edge coloring $\ell : E \to C$, and a positive integer k.
> **Question**: Is there a subset of colors $S \subseteq C$ with $|S| \leq k$ such that s and t are not in the same connected component in $G' = (V, E \setminus E_S)$, where $E_S := \{e \in E \mid \ell(e) \in S\}$?

In other words, we ask if there is a size-k set of colors S such that the removal of all edges colored in a color of S disconnects s from t. Such a set S is called a *colored* (s, t)*-cut*.

As far as we are aware, Colored (s, t)-Cut was first introduced by Jha et. al. in the context of directed attack graphs [26, 35]. In such an attack graph, an attacker starts in a state s of the network and aims to reach a specific state t for example by using different security holes. These security holes are modeled by the colors of the graph and the attacker is able to traverse an edge from u to v in color α by using the security hole α. The state s can be seen as the default state, where the network operates as intended and the state t can be seen as a state where, for example, the attacker gained administrative rights for the network. Hence, to prevent an attacker from reaching t, we have to fix at least one security hole for every (s, t)-path in the attack graph.

Later, Colored (s, t)-Cut was independently rediscovered on undirected graphs by Coudert et. al. [10] and Wang et. al. [39]. In these works, Colored (s, t)-Cut was used to model failures of specific edges in a network. In contrast to the work on attack graphs of Jha et. al. [26,35]. Coudert et. al. [10] and Wang et. al. [39] do not use an abstract graph model but the real communication network itself. In this scenario, one can understand the color of an edge as a condition under which this link of the network fails. For example, all modems of the same brand may be attacked simultaneously by a computer virus spreading the network. To analyze the robustness against multiple simultaneous faults of the communication between two specific computer endpoints s and t in the network, one aims to solve COLORED (s,t)-CUT. Now, a colored (s,t)-cut S can be interpreted a set of attacks S, such that the communication between s and t is impossible if all attacks in S are applied simultaneously. Throughout this work, we follow this interpretation and interpret COLORED (s,t)-CUT as the goal of an attacker aiming to destroy a network.

Another important problem for network security analysis, which is closely related to COLORED (s,t)-CUT, is the COLORED PATH problem.

> COLORED PATH [40]
> **Input**: An undirected graph $G = (V, E)$, two vertices $s, t \in V$, a set of colors C, an edge coloring $\ell : E \to C$, and a positive integer k.
> **Question**: Is there a subset of colors $S \subseteq C$ with $|S| \leq k$ such that there is an (s,t)-path P in G with $\ell(E(P)) \subseteq S$?

In other words, we ask if there is a size-k set of colors S such that s and t are connected in the graph consisting of the edges colored in S and their endpoints. Hence, COLORED PATH can be seen as the perspective of a network security analyst who tries to secure communication in his network against multiple simultaneous failures or attacks.

1.1 Related Work

Both COLORED (s,t)-CUT and COLORED PATH are NP-complete [10, 40]. Thus, it is unlikely that these problems can be solved in polynomial time. Hence, COLORED (s,t)-CUT and COLORED PATH where investigated extensively from a classic complexity theory and from a parameterized complexity theory point of view to obtain efficient algorithm for restricted instances [10, 11, 19, 26, 27, 35, 38–40, 42].

The NP-hardness of COLORED (s,t)-CUT was independently shown via a polynomial-time reduction from the HITTING SET problem [10, 26, 35]. This reduction will be recalled in Section 2.8, since many other hardness results for COLORED (s,t)-CUT can also be obtained from this reduction. Furthermore, it was shown that COLORED (s,t)-CUT is NP-complete even on bipartite planar graphs with a vertex cover number of two, that is, on graphs where every edge is incident with either s or t, and on complete graphs [38,39]. Hence, most graph structures alone do not give helpful insight on how to solve COLORED (s,t)-CUT efficiently. A notable exception is that COLORED (s,t)-CUT can be solved in polynomial time on graphs with a constant degree Δ [10].

When analyzing the structure of the colors of the instance, one can obtain more positive results. COLORED (s,t)-CUT can be solved in polynomial time if the span of each color is at most one, that is, if for every color $\alpha \in C$, the subgraph consisting of the edges colored in α and their endpoint, is connected [10]. Clearly, this is a generalization of uncolored graphs, on which COLORED (s,t)-CUT corresponds to the polynomial-time-solvable MIN (s,t)-CUT problem [17, 22]. In contrast, COLORED (s,t)-CUT is already NP-hard if every color is only assigned to two edges each [10] and can be solved in polynomial time if every color appears on at most two (s,t)-paths [38].

A trivialFPT-algorithm which runs in time $2^{|C|}n^{\mathcal{O}(1)}$ for COLORED (s,t)-CUT can be obtained by checking if S is a colored (s,t)-cut, for every subset of colors $S \subseteq C$ [10]. This algorithm is called an FPT-*algorithms* for the parameter $|C|$ because the exponential part of the running time depends only on $|C|$. This algorithm was improved to even smaller parameters like the number of colors c_{span} with span at least two [11, 30, 38] or the number of colors c_{path} which appear on at least three (s,t)-paths [30].

Moreover, COLORED (s,t)-CUT admits an FPT-algorithm when parameterized by the number p of (s,t)-paths of the instance [27] or both the size of the solution k and the length of the longest (s,t)-path [42]. For all those parameterizations, COLORED (s,t)-CUT does not admit polynomial kernels [27, 38, 42]. In other words, one is not able to find polynomial-time data reduction rules whose application yields an equivalent instance which has a size that is upper-bounded by a polyonmial of the previous parameters. Furthermore, COLORED (s,t)-CUT is W[2]-hard when parameterized by the size of the solution k even on graphs with pathwidth three [19]. Thus, COLORED (s,t)-CUTparameterized by k presumably admits no FPT-algorithm.

1.2 Our Results

In Section 2, we give the basic notation and definitions we use throughout this work as well as the standard reduction from HITTING SET to COLORED (s,t)-CUT.

In Section 3, we further analyze the parameterized complexity of Colored (s,t)-Cut parameterized by structural graph parameters. We show W[2]-hardness when parameterized by both the maximum degree Δ and the edge deletion distance to a maximum degree of three ξ_3. Thus, Colored (s,t)-Cut parameterized by both Δ and ξ_3 presumably admits no FPT-algorithm. In contrast, we show that parameterization by either the edge deletion distance to a maximum degree of two ξ_2 or the feedback edge set number fes lead to FPT-algorithms.

In Section 4.1, we introduce competitive defender-attacker games by generalizing Colored (s,t)-Cut. These games have the following rules: a defender and an attacker alternating choose sets of unchosen colors each turn. Both players are assigned a budget each turn and the size of the set of colors they choose must equal this budget. The attacker wins if the union of all sets of colors he chose is a colored (s,t)-cut, whereas, the defender wins if he can prevent this by choosing sets of colors since the attacker may not choose these colors in subsequent turns. These games can be analyzed to find strategies to protect important layers in multi-layer networks before an attacker is able to destroy them. We assume that the game where every agent has exactly one turn and where the defender starts, might be particularly interesting in practice.

We show that for a constant number of alternations between the agents, these games are complete for complexity classes of different levels of the polynomial-time hierarchy. Thus, the more alternations between the agents, the more complex it becomes to determine which player has a winning startegy. For a non-constant number of alternations between the agents, we introduce in Section 4.2 the games Colored (s,t)-Cut Vulnerability Game and Colored (s,t)-Cut Robustness Game and show that both problems are PSPACE-complete even if every agent chooses only one color in each turn. In this restricted case, these games can be seen as Shannon Switching Games where one player tries to destroy all (s,t)-paths, whereas, the other player tries to secure at least one (s,t)-path [18]. Throughout this work, we will refer to the collection of all these introduced games as *colored cut games*.

In Section 5, we analyze the parameterized complexity of the colored cut games in the hope to find efficient algorithms for them if specific parameters are small. First, we show that parameterization by the most natural parameters related to the budgets of the agents do not lead to FPT-algorithms. Second, we prove polynomial kernels for all these games when parameterized by both the number of colors $|C|$ and the vertex cover number of the input graph. Moreover, we generalize this kernelization algorithm by replacing the vertex cover number with an even smaller parameter.

In Section 6, we analyze the computational complexity of the colored cut games on restricted instances to understand what makes the games difficult. First, we show hardness for all these games on graphs with degree constraints such as complete and subcubic graph. Second, we show that none of the games can be solved in polynomial time on uncolored graphs. In contrast, Colored (s,t)-Cut can be solved in polynomial time on uncolored graphs [17, 22].

2 Preliminaries

In this section we introduce the basic notation and definitions we use throughout this work. First, we give basic graph and set notation. Afterwards, we give an overview on classic complexity theory and parameterized complexity theory. Finally, we recall the standard reduction from Hitting Set to Colored (s,t)-Cut.

2.1 Set and Graph Notation

For a finite set X, we denote with $\mathbb{P}(X) := \{A \mid A \subseteq X\}$ the *power set* of X, that is, the set of all subsets of X. Let $0 \leq k \leq |X|$, we define $\binom{X}{k} := \{S \subseteq X \mid |S| = k\}$. We say that $(Y_1, \dots, Y_r)$ is a *partition* of X if $Y_j \cap Y_k = \emptyset$ for all $1 \leq j < k \leq r$, and $\bigcup_{j=1}^{r} Y_j = X$. Moreover, we generalize a function $\ell : A \to B$ to sets with $\ell(A') := \{\ell(a) \mid a \in A'\}$ for all $A' \subseteq A$. Furthermore, we define the *inverse function for sets* $\ell^{-1} : \mathbb{P}(B) \to \mathbb{P}(A)$ with $\ell^{-1}(B') := \{a \in A \mid \ell(a) \in B'\}$ for every set $B' \subseteq B$.

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 $m := |E|$. Let $V' \subseteq V$, then we denote with $G[V'] := (V', E \cap \binom{V'}{2})$ the *induced subgraph* of V' and with $G - V' := G[V \setminus V']$ the induced subgraph of G obtained by deleting the vertices of V' and their incident edges. Analogously, we denote for some $E' \subseteq E$, the graph obtained by deleting the edges in E' with $G - E' := (V, E \setminus E')$. For a vertex $v \in V$ we denote with $N_G(v) := \{w \in V \mid \{v, w\} \in E\}$ the *neighborhood* of v in G. Furthermore, we denote with $\deg_G(v) := |N_G(v)|$ the *degree* of v in G, that is, the size of the neighborhood of v. If G is clear from the context, we may also write $N(v)$ and $\deg(v)$ instead. The *degree* Δ of G is defined as the maximum degree of all vertices of G, that is, $\Delta(G) := \max\{\deg_G(v) \mid v \in V\}$.

2.2 Paths

In a graph $G = (V, E)$, we call a sequence of vertices $P = (v_1, \dots, v_k) \in V^k, k \geq 1$, a *path* of length k in G if $\{v_i, v_{i+1}\} \in E$ for all $1 \leq i < k$. If $v_i \neq v_j$ for all $1 \leq i < j \leq k$, then we call P a *vertex-simple* path. If not mentioned otherwise, we only consider vertex-simple paths. Furthermore, we say that P is a (v_1, v_k)*-path*. Moreover, we denote with $V(P) := \{v_i \mid 1 \leq i \leq k\}$ the vertices of P and with $E(P) := \{\{v_i, v_{i+1}\} \mid 1 \leq i < k\}$ the edges of P.

Given two paths $P_1 = (v_1, \ldots, v_k)$ and $P_2 = (w_1, \ldots, w_r)$ in G, we define the *concatenation* $P_1 \cdot P_2 := (v_1, \ldots, v_k, w_1, \ldots, w_r)$. Note that $P_1 \cdot P_2$ is a path in G if $\{v_k, w_1\} \in E$. Furthermore, if $v_k = w_1$ we define the *merge* of P_1 and P_2 as $P_1 \multimap P_2 := (v_1, \ldots, v_k = w_1, \ldots, w_r)$. For a path $P = (v_1, \ldots, v_r)$ we denote with $\overleftarrow{P} := (v_r, \ldots, v_1)$ the *reverse path* of P.

Proposition 2.1. *Given a non-vertex-simple (a, b)-path P in G, then there is a vertex-simple (a, b)-path $\tilde{P}$ in G with $E(\tilde{P}) \subseteq E(P)$.*

This result is a well known fact, but since we may refer to it multiple times throughout this work, we give the proof for the sake of completeness.

Proof. Let $P = (v_1, \ldots, v_r)$ be a non-vertex-simple (a, b)-path in G. Then there is some v_x such that $v_x = v_y$ for $1 \leq x < y \leq r$. By definition, $P' = (v_1, \ldots, v_x) \multimap (v_y, \ldots, v_r)$ is an (a, b)-path in G and $E(P') \subseteq E(P)$. This can be done at most r times since the length of P' is less than the length of P. Hence, we get an (a, b)-path $\tilde{P}$ in at most r iterations such that $E(\tilde{P}) \subseteq E(P)$ and $\tilde{P}$ is vertex-simple. □

2.3 Graph Properties and Colored Graphs

For a graph $G = (V, E)$, a subset $V' \subseteq V$ is called an *independent set* if $G[V']$ is an edgeless graph, that is, $\{u, v\} \notin E$ for all $u, v \in V'$. A subset $V' \subseteq V$ is called a *connected component* if $V' \neq \emptyset$ is a maximal set of vertices such that there is at least one (u, v)-path in G for pairwise distinct $u, v \in V'$. Next, we will define some graph classes that will be used throughout this work. A graph $G = (V, E)$ is *connected*, if G has exactly one connected component; in other words, for every $u, v \in V$ there is at least one (u, v)-path in G. Note that in a connected graph, it holds that $m \geq n - 1$. A graph $G = (V, E)$ is a *tree*, if G is connected and contains exactly one (u, v)-path for every $u, v \in V$. Note that in a tree, it holds that $m = n - 1$. A graph $G = (V, E)$ is a *forest*, if G is a disjoint union of trees, that is, G contains at most one (u, v)-path for every $u, v \in V$. A graph $G = (V, E)$ is *bipartite*, if there is a partition (V_1, V_2) of V such that $G_1 := G[V_1]$ and $G_2 := G[V_2]$ are both edgeless graphs. A graph $G = (V, E)$ is *planar*, if there can be drawn in the Euclidean plane such that no two edges intersect each other. A graph $G = (V, E)$ is *complete*, if $E = \binom{V}{2}$, in other words, for all pairwise distinct $u, v \in V$ there is an edge $\{u, v\} \in E$. A graph $G = (V, E)$ is *cubic*, if $\deg(v) = 3$ for all $v \in V$ and *subcubic*, if $\deg(v) \leq 3$ for all $v \in V$.

An *edge colored graph with terminals* or short *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.

2.4 Colored Cuts

For a graph $G = (V, E)$ and two vertices $s \in V$ and $t \in V$, we call $E' \subseteq E$ 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. We say that $\tilde{C} \subseteq C$ is a *colored* (s,t)*-cut* in G if for every (s,t)-path P in G, it holds that $\ell(E(P)) \cap \tilde{C} \neq \emptyset$, in other words, $\ell^{-1}(\tilde{C})$ is an uncolored (s,t)-cut 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 $\ell(E(P)) \subseteq \tilde{C}$. In the following we denote for a colored graph $\mathcal{H} = (G, s, t, C, \ell)$ with $\mathcal{C}(\mathcal{H}) := \{\ell(E(P)) \mid P \text{ is an } (s,t)\text{-path in } G\}$ the set of colors of vertex-simple (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 $C' \in \mathcal{C}(\mathcal{H})$. 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 $C' \subseteq \tilde{C}$. Furthermore, if a colored graph $\mathcal{H}$ is part of a tuple I, we also use the notation $\mathcal{C}(I) := \mathcal{C}(\mathcal{H})$.

Definition 2.2. We call two colored graphs $\mathcal{H} = (G, s, t, C, \ell)$ and $\mathcal{H}' = (G', s', t', C, \ell')$ *colored-cut-equivalent* if for all $L_1 \in \mathcal{C}(\mathcal{H}) \cup \mathcal{C}(\mathcal{H}')$ there exists $L_2 \in \mathcal{C}(\mathcal{H}) \cap \mathcal{C}(\mathcal{H}')$ such that $L_2 \subseteq L_1$.

In other words, $\mathcal{H}$ 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 $\ell(E(P)) \subseteq \ell'(E(P'))$ 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.

Proposition 2.3. *Given two colored-cut-equivalent graphs* $\mathcal{H} = (G, s, t, C, \ell)$ *and* $\mathcal{H}' = (G', s', t', C', \ell')$ *and a set* $\tilde{C} \subseteq C$*, then* $\tilde{C}$ *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 $\tilde{C} \cap L_{\mathcal{H}} \neq \emptyset$ for all $L_{\mathcal{H}} \in \mathcal{C}(\mathcal{H}) \cap \mathcal{C}(\mathcal{H}')$. We show $\tilde{C} \cap L_{\mathcal{H}'} \neq \emptyset$ for all $L_{\mathcal{H}'} \in \mathcal{C}(\mathcal{H}')$. Let $L_{\mathcal{H}'} \in \mathcal{C}(\mathcal{H}')$, then there is some $L_{\mathcal{H}} \in \mathcal{C}(\mathcal{H}) \cap \mathcal{C}(\mathcal{H}')$ with $L_{\mathcal{H}} \subseteq L_{\mathcal{H}'}$ since $\mathcal{H}$ and $\mathcal{H}'$ are colored-cut-equivalent. Hence, $L_{\mathcal{H}'} \cap \tilde{C} \supseteq L_{\mathcal{H}} \cap \tilde{C} \neq \emptyset$ and therefore $\tilde{C}$ is a colored (s',t')-cut in G'. □

2.5 Clauses and Satisfiability

For a set of *variables* Z, we define the set of *literals* $\mathcal{L}(Z) := Z \cup \{\neg z \mid z \in Z\}$. A set 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. For a set of variables Z, a *(CNF-)clause* $\phi \subseteq \mathcal{L}(Z)$ is *satisfied* by an assignment $\tilde{Z}$ of Z if $\phi \cap \tilde{Z} \neq \emptyset$, and we write $\tilde{Z} \models \phi$. Analogously, a set $\Phi \subseteq \mathbb{P}(\mathcal{L}(Z))$ of (CNF-)clauses is satisfied by $\tilde{Z}$ if $\tilde{Z} \models \phi$ for all $\phi \in \Phi$, and we write $\tilde{Z} \models \Phi$. We say that a set Φ of clauses is in 3-CNF, if $\Phi \subseteq \binom{\mathcal{L}(Z)}{3}$, that is, every clause $\phi \in \Phi$ contained exactly three literals.

2.6 Complexity Theory

For a general overview about computational complexity theory we refer to [2, 23, 32].

We use the *big-O* notation to give worst-case analysis for running time and space efficiency for the algorithms we describe in this work. Let $f, g : \mathbb{N} \to \mathbb{R}$ be functions. We say that g is an *asymptotic upper bound* for f and write $f \in \mathcal{O}(g)$ if there exists $c > 0$ and $n_0 \in \mathbb{N}$ such that for all $n \geq n_0$ it holds that $f(n) \leq c \cdot g(n)$. If $f(n) \in \mathcal{O}(n^c)$ for some constant c, we say that f is *polynomial.*

A *decision problem* is a language $L \subseteq \{0,1\}^*$. We say that an *instance* $I \in \{0,1\}^*$ is a *yes-instance* of L if $I \in L$ and a *no-instance* of L otherwise. A *polynomial-time (many-one) reduction* from a decision problem A to a decision problem B is an algorithm running in polynomial time and transforming an instance I_A of A into an instance I_B of B such that I_A and I_B are equivalent. We write $A \leq_{\mathrm{P}} B$ when there is a polynomial-time reduction from A to B. A *complexity class* $\mathcal{L}$ is a set of decision problems. For the complexity classes we will analyze in this work, we say that a problem A is $\mathcal{L}$*-hard* if for every $L \in \mathcal{L}$ there is a polynomial-time reduction from L to A. Since the relation $\leq_{\mathrm{P}}$ is transitive, it is sufficient to show that $L \leq_{\mathrm{P}} A$ for some $\mathcal{L}$-hard problem L. If $A \in \mathcal{L}$ and $\mathcal{L}$-hard, then we say that A is $\mathcal{L}$*-complete.*

For a decision problem L a *verifier* V is an algorithm such that for all $w \in \{0,1\}^*$ it holds that $w \in L$ if and only if there is a *certificate* $c \in \{0,1\}^{p(|w|)}$ for some polynomial function p such that V accepts the input (w, c). Using the notion of verifier, one may define the hierarchy of complexity classes Σ_i^{P} for all $i \geq 0$, inductively. The class $\Sigma_0^{\mathrm{P}} := \mathrm{P}$ is the set of all problems that can be solved in polynomial time and for all $i \geq 0$ the class $\Sigma_{i+1}^{\mathrm{P}}$ contains exactly the problems that have a verifier V which runs in polynomial time and are allowed to use an *oracle* for a Σ_i^{P}-complete problem. That is, V can use an algorithm to solve a Σ_i^{P}-complete problem and runs in polynomial time where the running time is assumed to be $\mathcal{O}(1)$ of an oracle query. Similarly, the complexity classes $\Pi_i^{\mathrm{P}} := \{\{0,1\}^* \setminus L \mid L \in \Sigma_i^{\mathrm{P}}\}$ for all $i \geq 0$ is the set of all problems $\overline{L}$ such that the complement problem L is contained in Σ_i^{P}. Note that it is also possible to show that a problem L is in $\Sigma_{i+1}^{\mathrm{P}}$ by giving a verifier that uses an oracle for a Π_i^{P}-complete problem. By definition, $\Sigma_i^{\mathrm{P}} \subseteq \Sigma_{i+1}^{\mathrm{P}}, \Pi_i^{\mathrm{P}} \subseteq \Sigma_{i+1}^{\mathrm{P}}, \Sigma_i^{\mathrm{P}} \subseteq \Pi_{i+1}^{\mathrm{P}}$, and $\Pi_i^{\mathrm{P}} \subseteq \Pi_{i+1}^{\mathrm{P}}$ for all $i \geq 0$ and it is widely assumed that these are proper inclusions [2]. Canonical $\Sigma_{2i+1}^{\mathrm{P}}$-complete and Π_{2i}^{P}-complete problems are QSAT_{2i+1}-3-CNF and QSAT_{2i}-3-CNF, respectively [36].

> QSAT_{2i+1}-3-CNF
> **Input**: A set Φ of clauses in 3-CNF over the variables Z and a partition $(Y_1, X_2, \ldots, X_{i+1}, Y_{i+1})$ of Z.
> **Question**: Is it true that $\exists \tilde{Y}_1 \subseteq Y_1 . \forall \tilde{X}_2 \subseteq X_2 . \cdots . \forall \tilde{X}_{i+1} \subseteq X_{i+1} . \exists \tilde{Y}_{i+1} \subseteq Y_{i+1} : \tau_Z(\tilde{Y}_1 \cup \tilde{X}_2 \cup \cdots \cup \tilde{X}_{i+1} \cup \tilde{Y}_{i+1}) \models \Phi$?

> QSAT_{2i}-3-CNF
> **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 $\forall \tilde{X}_1 \subseteq X_1 . \exists \tilde{Y}_1 \subseteq Y_1 . \cdots . \forall \tilde{X}_i \subseteq X_i . \exists \tilde{Y}_i \subseteq Y_i : \tau_Z(\tilde{X}_1 \cup \tilde{Y}_1 \cup \cdots \cup \tilde{X}_i \cup \tilde{Y}_i) \models \Phi$?

These classes build the *polynomial-time hierarchy* [36]. We will use the more common names NP for Σ_1^{P} and coNP for Π_1^{P}.

The complexity class PSPACE contains exactly the decision problems that can be solved with polynomial space. For all $i \geq 0$, it holds that $\Sigma_i^{\mathrm{P}} \subseteq$ PSPACE and $\Pi_i^{\mathrm{P}} \subseteq$ PSPACE and, again, it is widely assumed that the containment is proper for all $i \geq 0$ [2]. A canonical PSPACE-complete problem is QSAT-3-CNF [36, 37].

> QSAT-3-CNF
> **Input**: A set Φ of clauses in 3-CNF over the variables $Z = \{x_j, y_j \mid 1 \leq j \leq i\}$ for some $i \in \mathbb{N}$.

Question: Is it true that $\forall \tilde{x}_1 \in \{x_1, \neg x_1\}.\exists \tilde{y}_1 \in \{y_1, \neg y_1\}. \cdots .\forall \tilde{x}_i \in \{x_i, \neg x_i\}.\exists \tilde{y}_i \in \{y_i, \neg y_i\} : \{\tilde{x}_j, \tilde{y}_j \mid 1 \leq j \leq i\} \models \Phi$?

2.7 Parameterized Complexity Theory

For more information about parameterized complexity theory we refer to [12, 15, 20, 31].

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 we also call the reduction a *polynomial parameter transformation*.

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. Moreover, we call h the *size of the kernel*. A parameterized problem L admits a kernel if and only if L admits an FPT-algorithm [31]. Clearly, one might be interested in finding kernels of small size for a given $L \in$ FPT. But for some parameterized problems one can show that it is not possible to find a kernel of polynomial size, unless NP $\subseteq$ coNP/poly [12], which is widely assumed to be false. If $L_1 \in$ FPT does not admit a polynomial kernel, unless NP $\subseteq$ coNP/poly, the unparameterized version of L_1 is NP-hard, and there is a polynomial parameter transformation to another parameterized problem L_2 where the unparameterized version of L_2 is contained in NP, then L_2 does also not admit a polynomial kernel, unless NP $\subseteq$ coNP/poly.

The classes W$[t], t \geq 0$, build the W-*hierarchy*, that is, W$[i] \subseteq$ W$[i+1]$ for all $i \geq 0$ and it is widely assumed that these are proper inclusions [15]. Moreover, it holds that FPT = W[0] $\subseteq$ W[1] $\subseteq \cdots \subseteq$ XP. Hence, we can show that a parameterized problem L is *fixed-parameter intractable*, unless FPT = W$[i]$, if L is W$[i]$-hard for some $i \geq 1$, that is, for every $L' \in$ W$[i]$, there is a parameterized reduction from L' to L. Again, it is sufficient to show that there is a parameterized reduction from L' to L for some W$[i]$-hard problem. Furthermore, we say that a parameterized problem L is contained in coW$[t], t \geq 0$, if $(\{0,1\}^* \times \mathbb{N}) \setminus L \in$ W$[t]$. Similar, L is coW$[t]$-hard if $(\{0,1\}^* \times \mathbb{N}) \setminus L$ is W$[t]$-hard. A known W[1]-hard problem is Independent Set parameterized by the size of the solution k [15].

> Independent Set
> **Input**: A graph $G = (V, E)$ and a positive integer k.
> **Question**: Is there a subset $S \subseteq V$ with $|S| \geq k$ such that $\{u, v\} \notin E$ for all $u, v \in S$?

A known W[2]-hard problem is Hitting Set parameterized by the size of the solution k [15].

> Hitting Set
> **Input**: A hypergraph $\mathcal{G} = (\mathcal{U}, \mathcal{F})$ where $\mathcal{U}$ is a finite set called the *universe*, $\mathcal{F} \subseteq \mathbb{P}(\mathcal{U})$ is a set of *hyperedges*, and a positive integer k.
> **Question**: Is there a subset $S \subseteq \mathcal{U}$ with $|S| \leq k$ such that $S \cap F \neq \emptyset$ for all $F \in \mathcal{F}$?

In contrast, the special case of Hitting Set where every hyperedge has size exactly two, called Vertex Cover, is in FPT when parameterized by k.

> Vertex Cover
> **Input**: A graph $G = (V, E)$ and a positive integer k.
> **Question**: Is there a subset $S \subseteq V$ with $|S| \leq k$ such that $S \cap e \neq \emptyset$ for all $e \in E$?

2.8 The Standard Reduction from Hitting Set

Next, we recall the reduction from Hitting Set to Colored (s,t)-Cut [10, 26, 35].

Lemma 2.4. Colored (s,t)-Cut *is* NP-*complete.*

Proof. We can verify in polynomial time that a set $\tilde{C} \subseteq C$ of size at most k is a colored (s,t)-cut in G by checking if s and t are in different connected components in $G - \ell^{-1}(\tilde{C})$. Therefore, Colored (s,t)-Cut is in NP.

Given a Hitting Set instance $(\mathcal{G} = (\mathcal{U}, \mathcal{F}), k)$, we describe how to construct an equivalent instance $I := (G = (V, E), s, t, C, \ell, k)$ of Colored (s,t)-Cut in polynomial time.

We can assume without loss of generality that $|F| \geq 2$ for all $F \in \mathcal{F}$ since an empty hyperedge $F \in \mathcal{F}$ leads to a no-instance and there is exactly one way to cover a hyperedge $F \in \mathcal{F}$ of size one. Furthermore, assume that there is an ordering on $\mathcal{F}$ and an ordering on every hyperedge $F \in \mathcal{F}$.

We set $C = \mathcal{U}$ and add two vertices s and t. Furthermore, we add for every hyperedge $F_j \in \mathcal{F}$ new vertices $v_1^j, \ldots, v_{|F_j|-1}^j$. Next, we add for every $F_j \in \mathcal{F}$ the edges $\{s, v_1^j\}$, $\{v_{|F_j|-1}^j, t\}$, and $\{v_i^j, v_{i+1}^j\}$ for all $i, 1 \leq i < |F_j| - 1$. Finally, we set for every $F_j \in \mathcal{F}$ the colors $\ell(\{s, v_1^j\}) := F_j(1), \ell(\{v_{|F_j|-1}^j, t\}) := F_j(|F_j|)$, and $\ell(\{v_i^j, v_{i+1}^j\}) := F_j(i+1)$ for all $i, 1 \leq i < |F_j| - 1$, where $F_j(y)$ denotes the yth element of F_j.

Note that there is an (s,t)-path P in G if and only if there is a hyperedge $F \in \mathcal{F}$ with $\ell(E(P)) = F$. Hence, $\mathcal{C}(I) = \mathcal{F}$ and therefore, there is a colored (s,t)-cut in G of size k if and only if there is a hitting set of size k in I'. □

Throughout this work, we will refer to the above reduction as the *standard reduction* from Hitting Set. Note that in this reduction, the following holds: the budget k in both instances is the same and $|C| = |\mathcal{U}|$. Since Hitting Set parameterized by k is W[2]-hard [15] and Hitting Set parameterized by $|\mathcal{U}|$ does not admit a polynomial kernel, unless NP $\subseteq$ coNP/poly [12, 15], the following holds.

Corollary 2.5 ([19, 27, 38])**.** Colored (s,t)-Cut *parameterized by* k *is* W[2]-*hard and* Colored (s,t)-Cut *parameterized by* $|C|$ *does not admit a polynomial kernel, unless* NP $\subseteq$ coNP/poly.

3 Structural Graph Parameters

In this section we analyze several structural graph parameters for Colored (s,t)-Cut and show that the problem is in XP for any of these parameters but there is no FPT-algorithm for Colored (s,t)-Cut when parameterized by the sum of all of these structural parameters, unless FPT = W[2].

As discussed above, Colored (s,t)-Cut parameterization by vertex deletion parameters is unlikely to lead to tractability results. We thus consider edge deletion parameters.

Definition 3.1. Let $G=(V,E)$ be a graph and $i \geq 0$ be an integer. Furthermore, let $\xi_i := \min\{|E'| \mid E' \subseteq E, G - E' \text{ has a maximum degree of } i\}$ be the *edge deletion distance to a maximum degree of* i.

Since Colored (s,t)-Cut parameterized by Δ is in XP [10], the parameter ξ_i thus measures the distance to a trivial case. Since $\Delta \leq \xi_i + i$, Colored (s,t)-Cut parameterized by ξ_i is in XP when i is constant. Note that the larger i, the smaller the parameter value ξ_i will be in most instances. We now show that even for small i, namely for $i=3$, an FPT algorithm for ξ_i is unlikely.

Proposition 3.2 ([10])**.** Colored (s,t)-Cut *is in* XP *parameterized by any of the following parameters:*

- *the budget k,*
- *the maximum degree Δ, and*
- *the edge deletion distance to a maximum degree of three ξ_3.*

Proof. The XP-algorithms are already known for k and Δ [10]. Note that we can assume that $\Delta > k$ since otherwise, the budget is large enough to select all colors of incident edges of s. These are at most Δ, and therefore the instance is a trivial yes-instance. It remains to show that Colored (s,t)-Cut is in XP parameterized by ξ_3. To this end, we show that $\xi_3 \geq \Delta - 3$. A graph G with a maximum degree of Δ contains at least one vertex $v \in V(G)$ with $\deg(v) = \Delta$, we have to delete at least $\Delta - 3$ edges incident with v to obtain a graph with maximum degree at most three. Hence, $\xi_3 \geq \Delta - 3$. $\square$

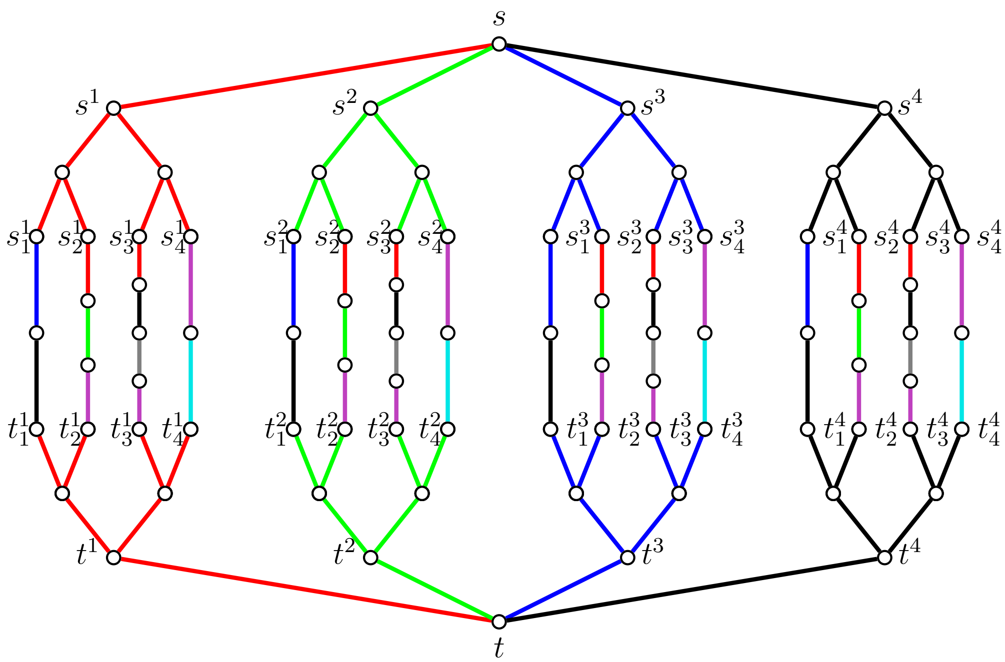


Figure 1: The Colored (s, t)-Cut instance constructed for the W[2]-hardness reduction of Theorem 3.3 for the Hitting Set instance $I = ((\mathcal{U}, \mathcal{F}), k)$ with $k = 3$, $\mathcal{U} = \{red, green, blue, black, gray, magenta, cyan\}$, and $\mathcal{F} = \{\{blue, black\}, \{red, green, magenta\}, \{red, black, gray, magenta\}, \{magenta, cyan\}\}$.

3.1 Hardness for Degree-Based Parameterizations

Since Colored (s, t)-Cut parameterized by $k, \Delta,$ and ξ_3 is in XP, we next show the fixed-parameter-intractability for the largest of these parameters ξ_3 (assuming FPT $\neq$ W[2])

Theorem 3.3. Colored (s, t)-Cut *parameterized by* ξ_3 *is* W[2]*-hard even on planar graphs.*

See Figure 1 for the instance I of Colored (s, t)-Cut constructed for an example instance of Hitting Set.

Proof. We give a parametrized reduction from Hitting Set parameterized by the size of the solution, which is known to be W[2]-complete [15]. Given a Hitting Set instance $I' = (\mathcal{G} = (\mathcal{U}, \mathcal{F}), k)$, we describe how to construct an equivalent COLORED (s,t)-CUT instance $I := (G = (V, E), s, t, C, \ell, k)$ in polynomial time and show that ξ_3 is bounded in k. An example of the construction can be seen in Figure 1.

Again, we assume without loss of generality that $|F| \geq 2$ for all $F \in \mathcal{F}$ since an empty hyperedge $F \in \mathcal{F}$ leads to a no-instance and there is exactly one way to cover a hyperedge $F \in \mathcal{F}$ of size one. Moreover, if $k \geq |\mathcal{U}|$, I' is obviously a yes-instance and if $k \leq 2$, I' can be solved in polynomial time. Hence, we can assume that $2 < k < |\mathcal{U}|$. Furthermore, assume that $\mathcal{U} = \{1, \dots, |\mathcal{U}|\}$ and that there is an ordering on $\mathcal{F}$ and an ordering on every hyperedge $F \in \mathcal{F}$.

We set $C = \mathcal{U}$ and define $G^i = (V^i, E^i)$ for all $i, 1 \leq i \leq k+1$, in the following way. The graph G^i contains two vertex disjoint balanced binary trees T_s^i and T_t^i with roots s^i, t^i and leaves s_j^i, t_j^i for all $j, 1 \leq j \leq |\mathcal{F}|$. We set $\ell(e) := i$ for all $e \in E(T_s^i) \cup E(T_t^i)$. Furthermore, we connect s_j^i and t_j^i with a path $P_j^i := (s_j^i, v_{(j,1)}^i, \dots, v_{(j,|F_j|-1)}^i, t_j^i)$ and set $\ell(\{s_j^i, v_{(j,1)}^i\}) := F_j(1), \ell(\{v_{(j,|F_j|-1)}^i, t_j^i\}) := F_j(|F_j|)$, and $\ell(\{v_{(j,x-1)}^i\}, v_{(j,x)}^i) := F_j(x)$ for all x, with $1 < x < |F_j|$, where $F_j(y)$ denotes the yth element of F_j. Note that the vertices $v_{(j,1)}^i, \dots, v_{(j,|F_j|-1)}^i$ occur only in P_j^i.

Finally, we define $G = (V, E)$ with $V := \{s, t\} \cup \bigcup_{1 \leq i \leq k+1} V^i$ and $E := \bigcup_{1 \leq i \leq k+1} (E^i \cup \{\{s, s^i\}, \{t, t^i\}\})$ and set $\ell(\{s, s^i\}) := \ell(\{t, t^i\}) := i$. That is, we connected s and t with s^i and t^i, respectively, with edges colored in i for all $i, 1 \leq i \leq k+1$. Note that G is planar.

Recall that beside s and t all vertices have degree at most three. Let $E' := \{\{s, s^i\}, \{t, t^i\} \mid 1 \leq i \leq k-2\}$ then $G - E'$ is cubic. Thus, $\xi_3 \leq |E'| = 2(k-2)$.

For the correctness of this parameterized reduction it remains to show that I is a yes-instance if and only if I' is a yes-instance. To this end, we show that $(\mathcal{G}, k)$ has a hitting set of size at most k if and only if (G, s, t, C, ℓ, k) has a colored (s,t)-cut of size at most k.

($\Rightarrow$) Let S be a hitting set of $\mathcal{G}$ with size at most k. By definition, $S \cap F_j \neq \emptyset$ for all $F_j \in \mathcal{F}$. Hence, removing all edges $\ell^{-1}(S)$ from G removes at least one edge in the path P_j^i from s_j^i to t_j^i for all i and $j, 1 \leq i \leq k+1, 1 \leq j \leq |\mathcal{F}|$. Note that for every path P from s to t in G there is at least one, j, $1 \leq j \leq |\mathcal{F}|$ such that P contains s_j^i and t_j^i for some i, $1 \leq i \leq k+1$. So by removing at least one edge from every path P_j^i, we separate s from t. It follows from definition, that S is a colored (s,t)-cut of size at most k for I.

($\Leftarrow$) Let S be a colored (s,t)-cut for I of size at most k, let $E_S := \ell^{-1}(S)$, and let $G' := G - E_S$. By construction, s and t have a path only colored in i to s_j^i and t_j^i, respectively, for all i and $j, 1 \leq j \leq |\mathcal{F}|, 1 \leq i \leq k+1$. Since S has size at most k there is at least one $i, 1 \leq i \leq k+1$, such that s and t are in the same connected component as s_j^x and t_j^x, respectively, in G' for all $j, 1 \leq j \leq |\mathcal{F}|$. The fact that S is a colored (s,t)-cut in G now implies that there is at least one edge $e_j \in E_S$ such that $e_j \in E(P_j^i)$ for each P_j^i with $j, 1 \leq j \leq |\mathcal{F}|$. Thus, $S \cap \ell(E(P_j^i)) \neq \emptyset$ for all $1 \leq j \leq |\mathcal{F}|$. By the fact that $\ell(E(P_j^i)) = F_j$ it follows that $S \cap F_j \neq \emptyset$ for all $1 \leq j \leq |\mathcal{F}|$. Consequently, S is a hitting set for $\mathcal{G}$ of size at most k. □

The next corollary follows directly from Theorem 3.3 and the fact that $\xi_3 + 3 \geq \Delta > k$.

Corollary 3.4. Colored (s,t)-Cut *parameterized by* $k + \Delta + \xi_3$ *is* W[2]-*hard even on planar graphs.*

3.2 FPT-Algorithms for Bounded Number of (s,t)-Paths

We now show that this result is tight by showing an FPT algorithm for ξ_2 which is obtained via an FPT algorithm for p, the number of (s,t)-paths in G.

The following Proposition is known [27], but we are not aware of a published proof. Hence, we give a proof for the sake of completeness.

Proposition 3.5. Colored (s,t)-Cut *is* FPT *parameterized by* p *and does not admit a polynomial kernel unless* NP $\subseteq$ coNP/poly.

Proof. First, we give an FPT-algorithm for Colored (s,t)-Cut parameterized by p. To this end, we give a parameterized reduction from Colored (s,t)-Cut parameterized by p to Hitting Set parameterized by the size of $\mathcal{F}$ which is known to be FPT [21]. Given an instance $I = (G, s, t, C, \ell, k)$, we compute the set $\mathcal{P}$ of (s,t)-paths in G in $\mathcal{O}(pn + m)$ time [4]. Hence, we can compute $\mathcal{C}(I) = \{\ell(E(P)) \mid P \in \mathcal{P}\}$ in in the same time. Then, it is obvious, that there is a colored (s,t)-cut of size at most k in G if and only if $I' = (\mathcal{G} := (C, \mathcal{C}(I)), k)$ has a hitting set of size at most k. Since Hitting Set can be solved in time $\mathcal{O}(2^{|\mathcal{F}|}|\mathcal{F}||\mathcal{U}|)$ [21], we can solve the instance I' in time $\mathcal{O}(2^p p|C|)$. By the fact that I and I' are equivalent, we can solve Colored (s,t)-Cut in time $\mathcal{O}(2^p p|C| + pn + m)$.

Note that the graph constructed by the standard reduction from Hitting Set has exactly $|\mathcal{F}|$ many (s,t)-paths. Hence, the standard reduction leads to a polynomial parameter transformation from Hitting Set parameterized by $|\mathcal{F}|$ to Colored (s,t)-Cut parameterized by p. Unless NP $\subseteq$ coNP/poly, Hitting Set parameterized by $|\mathcal{F}|$ does not admit a polynomial kernel [14] and therefore, neither does Colored (s,t)-Cut parameterized by the number of (s,t)-paths p. $\square$

Next, we show that p can be upper bounded by a computable function only depending on the feedback edge set number fes which implies an FPT-algorithm for Colored (s,t)-Cut parameterized by fes.

For a graph $G = (V, E)$, we call $F \subseteq E$ a *feedback edge set* if $G - F$ is a forest. We define with $\text{fes} := \min\{|F| \mid F \text{ is a feedback edge set}\}$ the *feedback edge set number*.

We assume that the following result might be known already but we were not able to find a proof for this particular statement. Hence, for the sake of completeness, we give a proof.

Lemma 3.6. *Let $G = (V, E)$ be a graph with feedback edge set number f, then for any $s, t \in V$ there are at most $\mathcal{O}(2^{\text{fes}+1}\,\text{fes}^{\text{fes}+1})$ many vertex simple (s,t)-paths in G.*

Proof. Let $F \subseteq E$ be a feedback edge set of G of size fes. Hence, $T := G - F$ is a forest. Since we only ask for vertex simple paths, every edge also occurs at most once in every (u,v)-path for every $u, v \in V$. We show that there are at most $2^j\,\text{fes}^j$ many (u,v)-paths P in G with $|E(P) \cap F| = j$ for every $j, 0 \leq j \leq \text{fes}$. That is,we bound the number of (u,v)-paths that contain exactly j edges of F. We show this statement by induction over j.

Since T is a forest, there is at most one (u,v)-path P in T for every $u, v \in V$. Hence, there is at most one (u,v)-path P in G with $E(P) \cap F = \emptyset$ and therefore the statement holds for $j = 0$.

Thus, assume that the statement holds for $0 \leq j-1 < \text{fes}$. We show that the statement also holds for j. Let $P = (v_1, \ldots, v_r)$ be an arbitrary (u,v)-path in G with $|E(P) \cap F| = j$ and let $e = \{v_i, v_{i+1}\} \in F$ such that $\{v_q, v_{q+1}\} \notin F$ for all $1 \leq q < i$, that is, e is the first feedback edge of P. By the induction hypothesis there is at most one (u, v_i)-path in G and at most $2^{j-1}\,\text{fes}^{j-1}$ many (v_{i+1}, v)-paths P_{j-1} in G with $|E(P_{j-1}) \cap F| = j - 1$. Since there are at most fes possible feedback edges and every such edge has two orientations, there are at most $2\,\text{fes}$ possibilities for e. Hence, there are at most $2^{j-1}\,\text{fes}^{j-1}\,2\,\text{fes} = 2^j\,\text{fes}^j$ different (u,v)-paths P in G with $|E(P) \cap F| = j$.

Thus, there are at most $\sum_{j=0}^{\text{fes}} 2^j \text{fes}^j \in \mathcal{O}(2^{\text{fes}+1} \text{fes}^{\text{fes}+1})$ many (u,v)-paths in G and therefore at most $\mathcal{O}(2^{\text{fes}+1} \text{fes}^{\text{fes}+1})$ many (s,t)-paths in G. □

The following can be obtained by applying Proposition 3.5.

Proposition 3.7. Colored (s,t)-Cut *is* FPT *parameterized by the feedback edge set number* fes *and does not admit a polynomial kernel, unless* NP ⊆ coNP/poly.

Proof. By Lemma 3.6 the number of (s,t)-paths p is bounded from above by a computation function h only depending on the feedback edge set number fes. Obviously, fes can be computed in $\mathcal{O}(n+m)$ time. Hence, we can use the FPT-algorithm from Proposition 3.5 to solve Colored (s,t)-Cut in $\mathcal{O}(2^{h(\text{fes})}h(\text{fes})|C| + h(\text{fes})n + m)$ time.

Next, we show the kernel lower bound. Note that the graph constructed by the standard reduction has a feedback edge set number of at most $|\mathcal{F}|-1$: by removing all incident edges of s except one, we can turn the graph into a tree. Hence, the standard reduction leads to a polynomial parameter transformation from Hitting Set parameterized by $|\mathcal{F}|$ to Colored (s,t)-Cut parameterized by fes where fes $= |\mathcal{F}| - 1$. Unless NP ⊆ coNP/poly, Hitting Set parameterized by $|\mathcal{F}|$ does not admit a polynomial kernel [14] and therefore, neither does Colored (s,t)-Cut parameterized by the feedback edge set number fes. □

Corollary 3.8. Colored (s,t)-Cut *is* FPT *parameterized by* ξ_2 *and does not admit a polynomial kernel, unless* NP ⊆ coNP/poly.

Proof. Let $I = (G = (V,E), s, t, C, \ell, k)$ be an instance of Colored (s,t)-Cut. Since we that G is connected, we get that $m \geq n-1$ and that the feedback edge set number is fes $= m - n + 1$. We show that $\xi_2 \geq \text{fes} - 1$. Note that for a graph $G' = (V', E')$ with a maximum degree of at most two, it holds that $|E'| \leq |V'|$. Hence, $m - \xi_2 \leq n$ and therefore $\xi_2 \geq m - n = \text{fes} - 1$. Therefore, we can use the FPT-algorithm from above to solve I in time $\mathcal{O}(2^{h(\xi_2+1)}h(\xi_2+1)|C| + h(\xi_2+1)n + m)$.

Next, we show the kernel lower bound. Note that for the standard reduction it holds that $\xi_2 = 2(|\mathcal{F}| - 2)$, since by removing all edges incident with s or t except two each, we can turn G into graph with maximum degree two. Hence, the standard reduction leads to an polynomial parameter transformation from Hitting Set parameterized by $|\mathcal{F}|$ to Colored (s,t)-Cut parameterized by ξ_2 where $\xi_2 = 2 * (|\mathcal{F}| - 2)$. Unless NP $\subseteq$ coNP/poly, Hitting Set parameterized by $|\mathcal{F}|$ does not admit a polynomial kernel [14] and therefore, neither does Colored (s, t)-Cut parameterized by ξ_2. $\square$

4 Competitive Colored Cut Games

In this section we introduce the *colored cut games* which can be seen as competitive defender-attacker games between two agents, a defender and an attacker. First, 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. Afterwards, we show that if the number of alternations is unbounded, the problems are PSPACE-complete.

4.1 Polynomial-time Hierarchy Versions

The following problems can be seen as a competitive game between a defender and an attacker with the following rules. The defender starts the game. In the jth *turn* of an agent, he has to choose a set $D_j \subseteq C$ or $A_j \subseteq C$ of given size d_j and a_j, respectively which is disjoint to all previous choses sets. The integers d_j and a_j, respectively are called the *budget* of the agent of turn j. Every agent has exactly i turns and the attacker wins the game if the union of sets he chose is a colored (s,t)-cut in G. Consequently, the defender wins if the attacker loses the game.

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

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

In $(\mathrm{DA})^i$-V we ask if the attacker has a *winning strategy*, that is, if he is able to react on the turns of defender and complete a colored (s,t)-cut in G. Analogously, we ask in $(\mathrm{DA})^i$-R if the defender has a winning strategy. Since one of the agents has a winning strategy, $(\mathrm{DA})^i$-R and $(\mathrm{DA})^i$-V are complement problems for all $i \geq 1$, that is, I is a yes-instance of $(\mathrm{DA})^i$-R if and only if I is a no-instance of $(\mathrm{DA})^i$-V.

Lemma 4.1. *For all $i \geq 1$ it holds that $(\mathrm{DA})^i$-V is Π_{2i}^{P}-hard and $(\mathrm{DA})^i$-R is Σ_{2i}^{P}-hard.*

Proof. We show this statement by a polynomial-time reduction from QSAT_{2i}-3-CNF. To this end, recall the definition.

> QSAT_{2i}-3-CNF
> **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 $\forall \tilde{X}_1 \subseteq X_1.\exists \tilde{Y}_1 \subseteq Y_1. \cdots .\forall \tilde{X}_i \subseteq X_i.\exists \tilde{Y}_i \subseteq Y_i : \tau_Z(\tilde{X}_1 \cup \tilde{Y}_1 \cup \cdots \cup \tilde{X}_i \cup \tilde{Y}_i) \models \Phi$?

Note that the QSAT_{2i}-3-CNF can also be seen as a competitive game between two players, where Player 1 and Player 2 have to 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, the combined assignment satisfies Φ.

Given an instance $I' = (Z, \Phi, X_1, Y_1, \ldots, X_i, Y_i)$ of QSAT_{2i}-3-CNF, we construct an instance $I = (G = (V, E), s, t, C, \ell, (d_1, \ldots, d_i), (a_1, \ldots, a_i))$ of $(\mathrm{DA})^i$-V such that I' is a yes-instance if and only if I is a yes-instance. Let $X_j = \{x_k^j \mid 1 \leq k \leq |X_j|\}, Y_j = \{y_k^j \mid 1 \leq k \leq |Y_j|\}$ for all $1 \leq j \leq i$ and let $\mathcal{L} := \mathcal{L}(Var)$.

We set $C := \mathcal{L} \cup C_d \cup C_a$ with $C_d := \{\alpha_j^d, \beta_j^d \mid 2 \leq j \leq i\}$, and $C_a := \{\alpha_j^a, \beta_j^a \mid 1 \leq j \leq i\}$. The idea is that every agent should choose an assignemnt for a set of variables in each turn. The colors of C_d and C_a are auxiliary colors so that we can blow up the budgets of the defender and the attacker, respectively in each turn to a value of at least two. With that, both agents have the possibility to win the game if the other agent has not chosen an assignment in the previous turn. Therefore, we 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 turns or otherwise they will lose the game. Observe that the only assignment on the variables of X_j that is disjoint to D_j is the complement assignment. Hence, the attacker is forced to pick A_j such that $A_j \cap X_j = X_j \setminus D_j$.

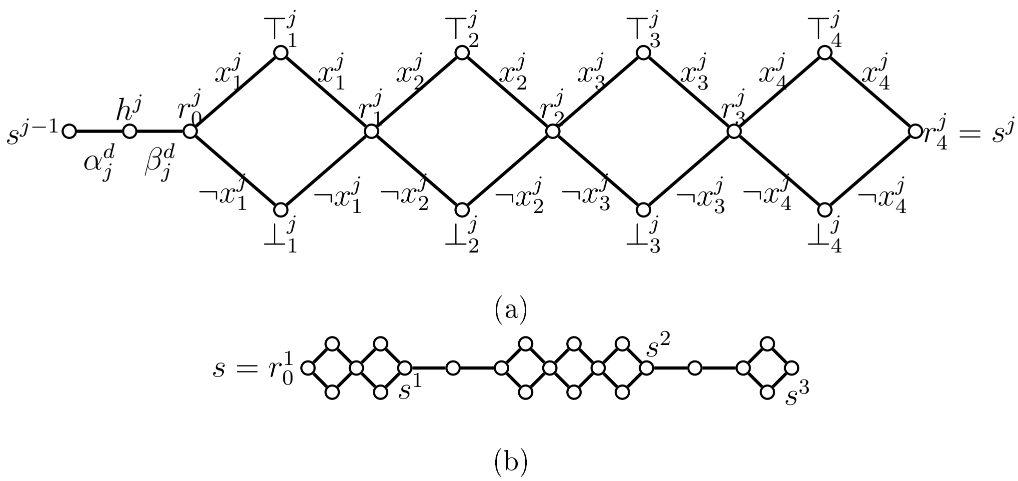


Figure 2: (a) shows the gadget for the defender for the variables of X_j with $|X_j| = 4$. (b) shows $G_D = (V_D, E_D)$ where $|X_1| = 2$, $|X_2| = 3$, and $|X_3| = 1$.

The graph we construct, 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 represent the variable gadget for the defender and attacker, respectively and V_Φ, E_Φ represent the gadget for the evaluation of the clauses. First, we introduce the variable gadgets for the defender which can be seen in Figure 2:

$$V_d := \{h^j \mid 2 \le j \le i\} \cup \bigcup_{j=1}^{i} \{r_0^j, r_k^j, \top_k^j, \bot_k^j \mid 1 \le k \le |X_j|\}$$

$$E_d := \bigcup_{j=1}^{i} \{\{r_{k-1}^j, \top_k^j\}, \{r_{k-1}^j, \bot_k^j\}, \{\top_k^j, r_k^j\}, \{\bot_k^j, r_k^j\} \mid 1 \le k \le |X_j|\}$$
$$\cup \left\{\{r_{|X_{j-1}|}^{j-1}, h^j\}, \{h^j, r_0^j\} \mid 2 \le j \le i\right\}$$

and set the colors:

$$\ell(\{r_{k-1}^j, \top_k^j\}) := \ell(\{\top_k^j, r_k^j\}) := x_k^j,$$
$$\ell(\{r_{k-1}^j, \bot_k^j\}) := \ell(\{\bot_k^j, r_k^j\}) := \neg x_k^j,$$
$$\ell(\{r_{|X_{j-1}|}^{j-1}, h^j\}) := \alpha_j^d \text{ and } \ell(\{h^j, r_0^j\}) := \beta_j^d.$$

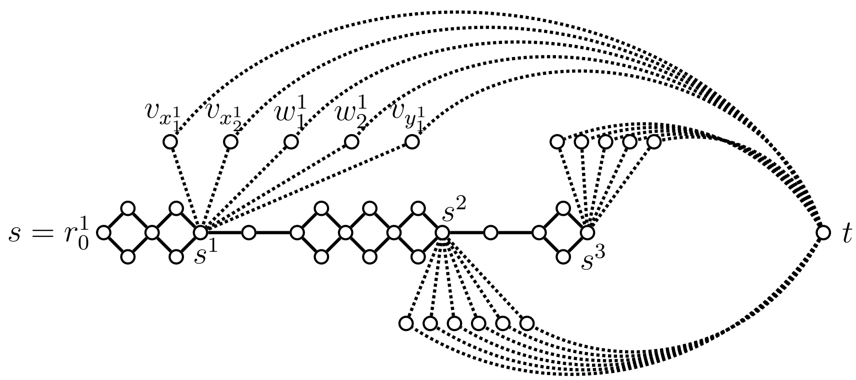


Figure 3: The variable gadget of the defender connected with the variable gadget of the attacker where $|Y_1| = 1, |Y_2| = 1$, and $|Y_3| = 2$. Solid edges belong to E_d and dotted edges belong to E_a.

In the following, let $s := s^0 := r_0^1$ and $s^j := r^j_{|X_j|}$ for all $1 \leq j \leq i$, that is, the vertex that will be connected to the gadget for the attacker for the variables of $X_j \cup Y_j$. The idea is that the defender has to choose in his jth turn the auxiliary colors $\{\alpha_j^d, \beta_j^d \mid j > 1\}$ together with an assignment of the variables of X_j, or otherwise the attacker can complete a colored (s,t)-cut by taking at most two colors in his next turn.

Next, we define the variable gadgets for the attacker. Figure 3 shows how the attacker and the defender gadget are connected:

$$
\begin{aligned}
V_a :=& \{t\} \cup \{v_x \mid x \in Z\} \cup \{w_1^j, w_2^j \mid 1 \le j \le i\} \\
E_a :=& \bigcup_{j=1}^{i} \{\{s^j, w_1^j\}, \{s^j, w_2^j\}, \{s^j, v_x\} \mid x \in X_j \cup Y_j\} \\
& \cup \{\{w, t\} \mid w \in V_a \setminus \{t\}\}
\end{aligned}
$$

and set the colors:

$$
\begin{aligned}
\ell(\{s^j, w_1^j\}) :=& \ell(\{w_1^j, t\}) := \alpha_j^a, \\
\ell(\{s^j, w_2^j\}) :=& \ell(\{w_2^j, t\}) := \beta_j^a, \\
\ell(\{s^j, v_x\}) :=& x \text{ and } \ell(\{v_x, t\}) := \neg x, x \in X_j \cup Y_j
\end{aligned}
$$

The idea is that the attacker has to choose in his jth turn the auxiliary colors α_j^a and β_j^a together with an assignment of the variables of $X_j \cup Y_j$, or otherwise the defender can complete a colored (s,t)-connector by taking 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 assignment to the assignment on the variables of X_j of the defender.

Finally, we define the gadget for evaluating the clauses. To this end, we assume an ordering on every clause $\phi_r \in \Phi$ and denote with $\phi_j(y)$ the yth element of ϕ_j. The final graph can be seen in Figure 4.

$$
\begin{aligned}
&V_\Phi := \{b_1^j, b_2^j \mid 1 \le j \le |\Phi|\} \\
&E_\Phi := \{\{s^i, b_1^j\}, \{b_1^j, b_2^j\}, \{b_2^j, t\} \mid 1 \le j \le |\Phi|\} \\
&\ell(\{s^i, b_1^j\}) := \phi_j(1), \\
&\ell(\{b_1^j, b_2^j\}) := \phi_j(2), \\
&\ell(\{b_2^j, t\}) := \phi_j(3)
\end{aligned}
$$

That is, for every $\phi \in \Phi$, we added a new (s^i, t)-path P with $\ell(E(P)) = \phi$. We set $d_1 := |X_1|, a_1 := |X_1| + |Y_1| + 2, d_j := |X_j| + 2$, and $a_j := |X_j| + |Y_j| + 2$ for all $1 < j \le i$. 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 $\ell(E(P)) = \phi$. Before we show the equivalence

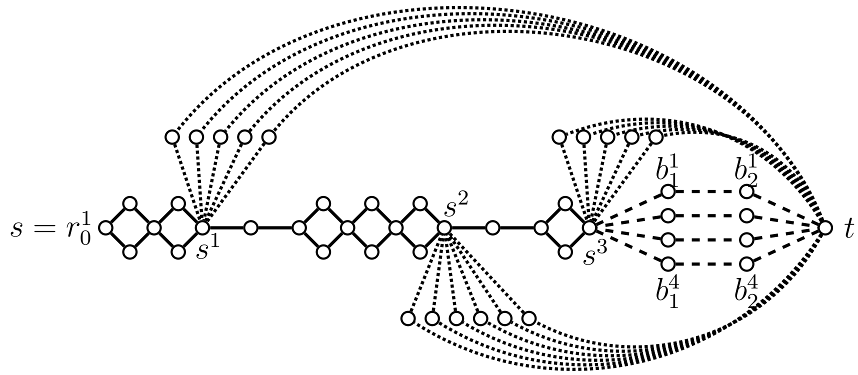


Figure 4: The final graph of the construction with $|\Phi| = 4$. Solid edges belong to E_d, dotted edges belong to E_a, and dashed edges belong to E_Φ. Note that the gadget for the clauses are connected with s^3 and t.

between I and I', we take 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. Observe that the set of (s^i,t)-paths in G_Φ is $\{P_q := (s^i, b_1^q, b_2^q, t) \mid 1 \leq q \leq |\Phi|\}$. Let $\tilde{C} \subseteq \mathcal{L}$ be an assignment for Z. By definition, $\tilde{C}$ is a colored (s^i,t)-cut in G_Φ if and only if $\tilde{C} \cap \ell(E(P_q)) \neq \emptyset$ for all $1 \leq q \leq |\Phi|$. Since $\ell(E(P_q)) = \phi_q$ for all $1 \leq q \leq |\Phi|$ this is the case if and only if $\tilde{C} \cap \ell(E(P_q)) = \tilde{C} \cap \phi_q \neq \emptyset$ which is by definition the case if and only if $\tilde{C} \models \Phi$. ■

Next, we define when a turn of an agent is 'nice'. Afterwards, we will show that the agent who first choses a set of colors that is not nice, will lose the game.

Definition 4.3. We call a set of colors $D_1 \subseteq C$ *nice* if D_1 is an assignment for X_1. Furthermore, we call $D_k \subseteq C, 2 \leq k \leq i$, nice if $D_k \cap \mathcal{L}$ is an assignment for X_k and $D_k \setminus \mathcal{L} = \{\alpha_k^d, \beta_k^d\}$. Analogously, we call the set of colors $A_k \subseteq C, 1 \leq k \leq i$, *nice* if $A_k \cap \mathcal{L}$ is an assignment for $X_k \cup Y_k$ and $A_k \setminus \mathcal{L} = \{\alpha_k^a, \beta_k^a\}$.

Note that the defined budgets allow every agent to pick a nice set of colors in each turn. Based on the defined budgets, if for some variable $z \in X_j$ it holds that $\{z, \neg z\} \subseteq D_j$, then $\{\alpha_j^d, \beta_j^d\} \not\subseteq D_j$ or there is a variable $z' \in X_j$ such that $\{z', \neg z'\} \cap D_j = \emptyset$. Hence, if D_j is not nice and $\{\alpha_j^d, \beta_j^d\} \subseteq D_j$, 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 the attacker chooses. With the following claims, we will show that the first agent who chooses a set of colors that is not nice, will lose the game. We use this to argue in the proof of the equivalence between I and I' that agents only choose nice sets of colors.

We start by showing that if the defender choses nice sets of colors in his first $k, 1 \leq k \leq i$, turns then he has completed a colored (s, s^k)-connector.

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

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

Assume that $D_j, 1 \leq j \leq i$ is nice. 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 $\ell(E(P_\top)) = \{x_q^j\}$ and $\ell(E(P_\bot)) = \{\neg x_q^j\}$, respectively for all $q, 1 \leq q \leq |X_j|$. 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 $\ell(E(P_q^j)) \subseteq D_j \subseteq \tilde{D}_j$. Hence, there is an (r_0^j, s^j)-path P^j in G with $\ell(E(P^j)) \subseteq D_j$. If $j = 1$, then P^1 is an (s^0, s^1)-path. If $j > 1$, then $P^{r_j} := (s^{j-1}, h^j, r_0^j)$ is a path in G with $\ell(E(P^{r_j})) = \{\alpha_j^d, \beta_j^d\} \subseteq D_j$. In combination, there is an (s^{j-1}, s^j)-path P' in G with $\ell(E(P')) \subseteq D_j$ if D_j is nice and therefore, there is an (s, s^k)-path P in G with $\ell(E(P)) \subseteq \tilde{D}_k$ if D_j is nice for all $1 \leq j \leq k$. ■

Next, we show that if the defender choses nice sets of colors in his first $k, 1 \leq k < i$ turns then he is able to complete a colored (s, t)-connector in in his next turn. Hence, he can win if the attacker does not chose colors to intersect all of these colored (s, t)-connector in his next turn.

Claim 4.5. *Let $1 \leq k \leq i$ and $\tilde{D}_k := \bigcup_{j=1}^{k} D_j$ such that D_j is nice for all $1 \leq j \leq k$, then the following sets are colored (s, t)-connectors in G:*

1. $\tilde{D}_k \cup \{\alpha_k^a\}$,
2. $\tilde{D}_k \cup \{\beta_k^a\}$, *and*
3. $\tilde{D}_k \cup \{x, \neg x\}$ *for any* $x \in X_k \cup Y_k$.

Proof. By Claim 4.4, there is an (s, s^k)-path P in G with $\ell(E(P)) \subseteq \tilde{D}_k$.

1. $\tilde{D}_k \cup \{\alpha_k^a\}$**.** By definition, then the path $P' = (r^k_{|X_k|}, w_1^k, t)$ is in G and $\ell(E(P')) = \{\alpha_k^a\}$. Hence, $\ell(E(P'')) = \ell(E(P)) \cup \{\alpha_k^a\} \subseteq \tilde{D}_k \cup \{\alpha_k^a\}$ with $P'' := P \multimap P'$.

2. $\tilde{D}_k \cup \{\beta_k^a\}$**.** This case is analogous.

3. $\tilde{D}_k \cup \{x, \neg x\}$ **for any** $x \in X_k \cup Y_k$**.** Let $x \in X_k \cup Y_k$, the path $P' = (r^k_{|X_k|}, v_x, t)$ is in G and $\ell(E(P')) = \{x, \neg x\}$. Hence, for $P'' := P \multimap P'$ it holds that $\ell(E(P'')) = \ell(E(P)) \cup \{x, \neg x\} \subseteq \tilde{D}_k \cup \{x, \neg x\}$. ■

Hence, if the defender has chosen nice sets of colors in his first k turns and the attacker chose a set of colors in turn k that is not nice, then the defender can win the game in turn $k + 1$. The wining strategy for this will be shown in the proof of the equivalence between I and I'.Therefore, the attacker has no winning strategy if he is the first agent who choses a set of color which is not nice.

Next, we show analogous claims for the attacker. That is, we show in Claim 4.6 and 4.7 that the attacker can win the game in turn $k + 1$ if he chose nice sets of colors in his first k turns and the defender did not chose a nice set of colors in turn $k + 1$.

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

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 an induction over k.

By construction, $s^1 \in V(P)$ for every (s, t)-path P in G. Hence, the Claim holds for $k = 1$.

Assume that the statement is true for some $k' = k - 1$, $0 \leq k < i$. We show that the statement is true for k. Assume towards a contradiction that there is an (s, t)-path P in G with $\ell(E(P)) \cap \tilde{A}_k = \emptyset$ and $s^k \notin V(P)$. By the induction hypothesis, we know that $s^{k'} \in V(P)$. Note that by construction for every $(s^{k'}, t)$-path P^k with $h^k \in V(P^k)$ it holds that $s^k \in V(P^k)$. Assume towards a contradiction that $h^k \notin V(P)$ and therefore, $V(P) \cap (\{v_x \mid x \in X_{k'} \cup Y_{k'}\} \cup \{w_1^{k'}, w_2^{k'}\}) \neq \emptyset$.

Case 1: $\{w_1^{k'}, w_2^{k'}\} \cap V(P) \neq \emptyset$**.** Then, $\{\{w_1^{k'}, t\}, \{w_2^{k'}, t\}\} \cap E(P) \neq \emptyset$ and therefore $\ell(E(P)) \cap \{\alpha_{k'}^a, \beta_{k'}^a\} \neq \emptyset$. Since $A_{k'}$ is nice, it follows that $\{\alpha^a_{k'}, \beta^a_{k'}\} \subseteq A_{k'}$ and therefore $\ell(E(P)) \cap A_{k'} \neq \emptyset$ which is a contradiction.

Case 2: $\{v_x \mid x \in X_{k'} \cup Y_{k'}\} \cap V(P) \neq \emptyset$. 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 $\{x, \neg x\} \subseteq \ell(E(P))$. Since $A_{k'}$ is nice, it follows that $A_{k'} \cap \{x, \neg x\} \neq \emptyset$ and hence $\ell(E(P)) \cap A_{k'} \neq \emptyset$ which is a contradiction. ■

Next, we show that if the attacker choses nice colors in his first k turns then he is able to complete a colored (s,t)-cut in his next turn. Hence, he can win if the defender does not chose colors to intersect all of these colored (s,t)-cuts in his next turn.

Claim 4.7. *Let $1 \leq k \leq i$ and $\tilde{A}_k := \bigcup_{j=1}^{k-1} A_j$ such that A_j is nice for all $1 \leq j < k$, then the following sets are colored (s,t)-cuts in G:*

- $\tilde{A}_k \cup \{\alpha^d_k\}$ *if $k > 1$,*
- $\tilde{A}_k \cup \{\beta^d_k\}$ *if $k > 1$, and*
- $\tilde{A}_k \cup \{x, \neg x\}$ *for any $x \in X_k$.*

Proof. Let $1 < k \leq i$ and assume that A_j is nice for all $1 \leq j < k$. By Claim 4.6, $s^{k-1}, s^k \in V(P)$ for every (s,t)-path P in G with $\ell(E(P)) \cap \tilde{A}_k = \emptyset$. Therefore, we show that for every (s^{k-1}, s^k)-path P^k in G with $t \notin V(P^k)$ it holds that $\ell(E(P^k)) \cap \{x, \neg x\} \neq \emptyset$ for all $x \in X_k$, and if $k > 1$ it holds that $\{\alpha^d_k, \beta^d_k\} \subseteq \ell(E(P^k))$. We can assume that $t \notin V(P^k)$ since otherwise there is an (s,t)-path P in G with $\ell(E(P)) \cap \tilde{A}_k = \emptyset$ 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 $r^k_0, \ldots, r^k_{|X_k|}$, and h^k if $k > 1$. Thus, $\ell(E(P^k)) \cap \{x, \neg x\} \neq \emptyset$ for all $x \in X_k$, and if $k > 1$, we get that $\{\alpha^d_k, \beta^d_k\} \subseteq \ell(E(P^k))$. ■

Hence, if the attacker has chosen nice sets of colors in his first $k-1$ turns and the defender choses a set of colors in turn k which is not nice, then the attacker can win the game in turn k. The exact strategy for that will be described in the proof of the equivalence between I and I'. Hence, the defender has no winning strategy if he is the first agent who choses a set of colors which is not nice. In combination with the conclusion of Claim 4.5, the first agent who does not choose a nice set of colors, loses the game. Hence, we will assume that all chosen sets of colors are nice 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 set of literals he chose is a satisfying assignment for Φ.

Claim 4.8. *Let D_j, A_j be nice for all $1 \le j \le i$ and $\tilde{A} := \bigcup_{j=1}^{i} A_j$, then $\tilde{A}$ is a colored (s,t)-cut in G if and only if $\tilde{A} \cap \mathcal{L}$ 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 $\ell(E(P)) \cap \tilde{A} = \emptyset$. Since A_i is nice, by construction, all (s^i,t)-paths $P' = (s^i, w, t)$ with $w \in V_a \cap N(s^i)$ are cut. Thus, $b_1^q \in V(P)$ for some $1 \le q \le |\Phi|$. By Claim 4.4 there is an (s,s^i)-path P_d in G with $\ell(E(P_d)) \subseteq \tilde{D}_i$. 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$_{2i}$-3-CNF instance is a yes-instance if and only if the constructed $(\mathrm{DA})^i$-V 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 $1 \le k \le i$ 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]. 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$_{2i}$-3-CNF instance. We will use these functions to describe a winning strategy for the attacker in the $(\mathrm{DA})^i$-V 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 $a_1 \ge 2$ 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 $A_1 := \{\alpha_1^a, \beta_1^a\} \cup \tau_{X_1 \cup Y_1}(\overline{D}_1 \cup f_1(\overline{D}_1))$ which is nice and disjoint from D_1.

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

So, we can assume that D_j is nice and A_j is defined as described for all $1 \leq j \leq i$. 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 $\tilde{A}_i \cap \mathcal{L}$ 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 $\tilde{A}_i \cap \mathcal{L} = \tau_Z(\overline{D}_1 \cup f_1(\overline{D}_1) \cup \cdots \cup \overline{D}_i \cup f_i(\bigcup_{k=1}^{i} \overline{D}_k)) \models \Phi$. 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 contra position. 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 $1 \leq k \leq i$ 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$_{2i}$-3-CNF instance. We will use these functions to describe a winning strategy for the defender in the (DA)^i-V 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 $\alpha_1^a \notin A_1, \beta_1^a \notin A_1$, or $\{x, \neg x\} \cap A_1 = \emptyset$ for some $x \in X_1 \cup Y_1$. By Claim 4.5, $D_1 \cup \{\alpha_1^a\}, D_1 \cup \{\beta_1^a\}$, and $D_1 \cup \{x, \neg x\}, x \in X_1 \cup Y_1$ are all colored (s,t)-connectors in G. If $i = 1$, then the attacker cannot choose a colored (s,t)-cut anymore and otherwise $d_2 \geq 2$ by definition. In both cases the defender has a winning strategy. So, we assume that A_1 is nice. Therefore, $A_1 \cap \mathcal{L}$ is an assignment for $X_1 \cup Y_1$. Since D_1 is also nice, $D_1 \cap \mathcal{L}$ 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 $1 < j \leq i$ such that D_r and A_r are nice for all $1 \leq r < j$. Let $\tilde{Y}_r := A_r \cap Y_r$ for all $1 \leq r < j$, that is, 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 $D_j := \{\alpha_j^d, \beta_j^d\} \cup \tau_{X_j}(X_j \setminus \overline{A}_j)$. 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 $\alpha_j^a \notin A_j, \beta_j^a \notin A_j$, or $\{x, \neg x\} \cap A_j = \emptyset$ for some $x \in X_j \cup Y_j$. By Claim 4.5, $\tilde{D}_j \cup \{\alpha_j^a\}, \tilde{D}_j \cup \{\beta_j^a\}$, and $\tilde{D}_j \cup \{x, \neg x\}, x \in X_j \cup Y_j$ are all colored (s,t)-connectors 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, $A_j \cap \mathcal{L}$ is an assignment for $X_j \cup Y_j$. Since D_j is also nice, $D_j \cap \mathcal{L}$ 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 $1 \leq j \leq i$. 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 $\tilde{A}_i \cap \mathcal{L}$ 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.

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

So far, we analyzed competitive games in which the defender starts. Next, we introduce the analog problems in which the attacker starts the game.

A(DA)^i Colored (s,t)-Cut Robustness (A(DA)^i-R)
Input: A colored graph $(G = (V, E), s, t, C, \ell)$ and two integer vectors $\vec{d} := (d_2, \ldots, d_{i+1}) \in \mathbb{N}^i$ and $\vec{a} := (a_1, \ldots, a_{i+1}) \in \mathbb{N}^{i+1}$ such that $a_1 + \sum_{k=2}^{i+1}(d_k + a_k) \leq |C|$.
Question: $\forall A_1 \in \binom{C}{a_1}.\exists D_2 \in \binom{C\setminus A_1}{d_2}.\forall A_2 \in \binom{C\setminus(A_1 \cup D_2)}{a_2}. \cdots .\forall A_{i+1} \in \binom{C\setminus((\bigcup_{k=1}^{i}(A_k \cup D_{k+1}))}{a_{i+1}}$: such that $\bigcup_{k=1}^{i+1} A_k$ is not a colored (s,t)-cut in G?

A(DA)^i Colored (s,t)-Cut Vulnerability (A(DA)^i-V)
Input: A colored graph $(G = (V, E), s, t, C, \ell)$ and two integer vectors $\vec{d} := (d_2, \ldots, d_{i+1}) \in \mathbb{N}^i$ and $\vec{a} := (a_1, \ldots, a_{i+1}) \in \mathbb{N}^{i+1}$ such that $a_1 + \sum_{k=2}^{i+1}(d_k + a_k) \leq |C|$.
Question: $\exists A_1 \in \binom{C}{a_1}.\forall D_2 \in \binom{C\setminus A_1}{d_2}.\exists A_2 \in \binom{C\setminus(A_1 \cup D_2)}{a_2}. \cdots .\exists A_{i+1} \in \binom{C\setminus((\bigcup_{k=1}^{i}(A_k \cup D_{k+1}))}{a_{i+1}}$: such that $\bigcup_{k=1}^{i+1} A_k$ is a colored (s,t)-cut in G?

Observe that $\mathrm{A(DA)}^0$-V is equivalent to COLORED (s,t)-CUT and for all $i \geq 1$ $\mathrm{A(DA)}^{i-1}$-V is a special case of $\mathrm{(DA)}^i$-V where the budget of the first turn of the defender is zero and $\mathrm{(DA)}^i$-V is a special case of $\mathrm{A(DA)}^i$-V where the budget of the first turn of the attacker is zero. Hence, COLORED (s,t)-CUT is a special case of all the problems $\mathrm{(DA)}^i$-V and $\mathrm{A(DA)}^i$-V. Note that $\mathrm{(DA)}^i$-R, $\mathrm{(DA)}^i$-V, $\mathrm{A(DA)}^i$-R, and $\mathrm{A(DA)}^i$-V are games that end with the attacker as the last agent. This is the case since after the attacker performed his last turn, he can only win if and only if he has already chosen a colored (s,t)-cut in G. Hence, a turn of the defender afterwards is unnecessary.

Corollary 4.9. *For all $i \geq 0$, it holds that* $\mathrm{A(DA)}^i$-V *is* $\Sigma^{\mathrm{P}}_{2i+1}$*-hard and* $\mathrm{A(DA)}^i$-R *is* Π^{P}_{2i+1}*-hard.*

Proof. We show this statement by a polynomial-time reduction from the $\Sigma^{\mathrm{P}}_{2i+1}$-complete problem QSAT_{2i+1}-3-CNF [36]. To this end, recall the definition of QSAT_{2i+1}-3-CNF.

> QSAT_{2i+1}-3-CNF
> **Input**: A set Φ of clauses in 3-CNF over the variables Z and a partition $(Y_1, X_2, \ldots, X_{i+1}, Y_{i+1})$ of Z.
> **Question**: Is it true that $\exists \tilde{Y}_1 \subseteq Y_1 . \forall \tilde{X}_2 \subseteq X_2 . \cdots . \forall \tilde{X}_{i+1} \subseteq X_{i+1} . \exists \tilde{Y}_{i+1} \subseteq Y_{i+1} : \tau_Z(\tilde{Y}_1 \cup \tilde{X}_2 \cup \cdots \cup \tilde{X}_{i+1} \cup \tilde{Y}_{i+1}) \models \Phi$?

Note that QSAT_{2i+1}-3-CNF starts and ends with an existential quantified set. Hence, it is a special case of $\mathrm{QSAT}_{2(i+1)}$-3-CNF where the first universal quantified set is empty.

Given an instance $I = (Z, \Phi, Y_1, \ldots, X_{i+1}, Y_{i+1})$ of QSAT_{2i+1}-3-CNF, then I is equivalent to the instance $I_2 = (Z, \Phi, X_1 = \emptyset, Y_1, \ldots, X_{i+1}, Y_{i+1})$ of $\mathrm{QSAT}_{2(i+1)}$-3-CNF. Therefore, we use the reduction 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 $\mathrm{(DA)}^{i+1}$-V. Note that by construction of I'_2 we get that $d_1 = |X_1| = 0$. Let $I' = (G = (V, E), s, t, C, \ell, (a_1, \ldots, a_{i+1}), (d_2, \ldots, d_{i+1}))$ be an $\mathrm{A(DA)}^i$-V instance. Clearly, I'_2 is equivalent to I'. Therefore, we constructed for the I an equivalent instance I' of $\mathrm{A(DA)}^i$-V in polynomial time. Hence, $\mathrm{A(DA)}^i$-V is $\Sigma^{\mathrm{P}}_{2i+1}$-hard and $\mathrm{A(DA)}^i$-R is Π^{P}_{2i+1}-hard. □

Theorem 4.10. *For all $i \geq 0$, it holds that* $\mathrm{A(DA)}^i$-V *is* $\Sigma^{\mathrm{P}}_{2i+1}$*-complete and* $\mathrm{A(DA)}^i$-R *is* Π^{P}_{2i+1}*-complete and for all $i \geq 1$ it holds that* $\mathrm{(DA)}^i$-V *is* Π^{P}_{2i}*-complete and* $\mathrm{(DA)}^i$-R *is* Σ^{P}_{2i}*-complete.*

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

By the fact that $\mathrm{A}(\mathrm{DA})^0$-V and Colored (s,t)-Cut are equivalent, we get that $\mathrm{A}(\mathrm{DA})^0$-V is $\mathrm{NP} = \Sigma_1^{\mathrm{P}}$-complete and $\mathrm{A}(\mathrm{DA})^0$-R is Π_1^{P}-complete [10]. Hence, the statement holds for $i = 0$.

Assume that the statement is true for some $0 \leq j-1$, we show that the statement is also true for j. To this end, we show two inductive steps.

First, we show that the statment is true for $(\mathrm{DA})^j$-R if it is true for $\mathrm{A}(\mathrm{DA})^{j-1}$-V. Let $I = (G, s, t, C, \ell, (d_1, \dots, d_j), (a_1, \dots, a_j))$ be an instance of $(\mathrm{DA})^j$-R 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, 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 $I' = (G', s, t, C \setminus D_1, \ell', (a_1, \dots, a_j), (d_2, \dots, d_j))$ of $\mathrm{A}(\mathrm{DA})^{j-1}$-R. Since $\mathrm{A}(\mathrm{DA})^{j-1}$-R is Π_{2j-1}^{P}-complete due to the induction hypothesis, it follows that $(\mathrm{DA})^j$-R is Σ_{2j}^{P}-complete and $(\mathrm{DA})^j$-V is Π_{2j}^{P}-complete.

Finally, we show that the statment is true for $\mathrm{A}(\mathrm{DA})^j$-V if it is true for $(\mathrm{DA})^j$-R. Let $I = (G, s, t, C, \ell, (a_1, \dots, a_{j+1}), (d_2, \dots, d_{j+1}))$ be an instance of $\mathrm{A}(\mathrm{DA})^j$-V 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 $I' = (G', s, t, C', \ell, (d_2, \dots, d_{j+1}), (a_2, \dots, a_{j+1}))$ of $(\mathrm{DA})^j$-V. Since $(\mathrm{DA})^j$-V is Π_{2j}^{P}-complete due to the induction hypothesis, it follows that $\mathrm{A}(\mathrm{DA})^j$-V is $\Sigma_{2j+1}^{\mathrm{P}}$-complete and $\mathrm{A}(\mathrm{DA})^j$-R is Π_{2j+1}^{P}-complete. □

Recall the property of colored-cut-equivalent graphs. If $\mathcal{H} = (G, s, t, C, \ell)$ and $\mathcal{H}' = (G', s', t', C, \ell')$ are colored-cut-equivalent, 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'. Therefore, the following Claims follow directly.

Claim 4.11. *Let $i \geq 1$, then the $(\mathrm{DA})^i$-R instances $I = (G, s, t, C, \ell, \vec{d}, \vec{a})$ and $I' = (G', s', t', C, \ell', \vec{d}, \vec{a})$ are equivalent if (G, s, t, C, ℓ) and (G', s', t', C, ℓ') are colored-cut-equivalent.*

Claim 4.12. *Let $i \geq 0$, then the* $\mathrm{A(DA)}^i$-R *instances* $I = (G, s, t, C, \ell, \vec{a}, \vec{d})$ *and* $I' = (G', s', t', C, \ell', \vec{a}, \vec{d})$ *are equivalent if* (G, s, t, C, ℓ) *and* (G', s', t', C, ℓ') *are colored-cut-equivalent.*

Next, we will use these claims to show that $\mathrm{(DA)}^i$-V is Π^{P}_{2i}-hard and $\mathrm{A(DA)}^i$-V is $\Sigma^{\mathrm{P}}_{2i+1}$-hard even on bipartite planar graphs. To this end, we first show the following.

Proposition 4.13. *For every colored graph* $\mathcal{H} = (G = (V, E), s, t, C, \ell)$ *there is a colored-cut-equivalent graph* $\mathcal{H}' = (G' = (V', E'), s', t', C, \ell')$ *where* G' *is bipartite and can be computed in polynomial time.*

Proof. Given a colored graph $\mathcal{H} = (G = (V, E), s, t, C, \ell)$, we define a colored graph $\mathcal{H}' = (G' = (V', E'), s', t', C, \ell')$ where G' is bipartite and show that $\mathcal{H}$ and $\mathcal{H}'$ are colored-cut-equivalent.

We construct $\mathcal{H}'$ 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 subdivided every edge $\{u, w\} \in E$ into two edges $\{u, v_{\{u,w\}}\}$ and $\{v_{\{u,w\}}, w\}$ of the same color. By construction, G' is bipartite, since all vertices $v \in V$ have only neighbors in $V' \setminus V$ and vice versa. This can be done in polynomial time.

Next, we show that $\mathcal{C}(\mathcal{H}) = \mathcal{C}(\mathcal{H}')$, which implies that $\mathcal{H}$ and $\mathcal{H}'$ are colored-cut-equivalent. By construction, it follows that $P = (u_1, \ldots, u_k)$ 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'. Hence, $\ell(E(P))) = \ell'(E(P'))$ and therefore $\mathcal{C}(\mathcal{H}) = \mathcal{C}(\mathcal{H}')$. □

Corollary 4.14. *For all $i \geq 1$, it holds that* $\mathrm{(DA)}^i$-V *is* Π^{P}_{2i}*-hard even on bipartite planar graphs.*

Proof. Let $I = (G, s, t, C, \ell, (d_1, \ldots, d_i), (a_1, \ldots, a_i))$ be an instance of $\mathrm{(DA)}^i$-V defined as described in the proof of Lemma 4.1. We show that there is a colored graph (G', s', t', C, ℓ') that is colored-cut-equivalent to (G, s, t, C, ℓ) where G' is bipartite and planar and that can be computed in polynomial time.

Note that the constructed graph in the proof of Lemma 4.1 is already planar. By Proposition 4.13 there is colored-cut-equivalent graph (G', s', t', C, ℓ') that can be computed in polynomial time where G' is bipartite. Note that G' is still planar since it was constructed by only inserting a new vertex in the middle of each edge of G. Hence, G' is bipartite planar and therefore I is a yes-instance if and only if $I' = (G', s', t', C, \ell', (d_1, \ldots, d_i), (a_1, \ldots, a_i))$ is a yes-instance due to Claim 4.11. $\square$

The following corollary can be shown in the same way.

Corollary 4.15. *For all $i \geq 0$, it holds that* $\mathrm{A(DA)}^i$-V *is* $\Sigma^{\mathrm{P}}_{2i+1}$*-hard even on bipartite planar graphs.*

4.2 PSPACE **Version**

In this subsection we show that if the number of alternations between the attacker and the defender is unbounded, it is PSPACE-complete to determine which agent has a winning strategy. Therefore, we start by defining these generalized versions of $(\mathrm{DA})^i$-R and $(\mathrm{DA})^i$-V.

> COLORED (s,t)-CUT ROBUSTNESS GAME (CCRG)
> **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**: $\exists D_1 \in \binom{C}{d_1}.\forall A_1 \in \binom{C \setminus D_1}{a_1}.\exists D_2 \in \binom{C\setminus(D_1 \cup A_1)}{d_2}. \cdots .\forall A_i \in \binom{C\setminus((\bigcup_{k=1}^{i-1}(D_k \cup A_k)) \cup D_i)}{a_i}$: such that $\bigcup_{k=1}^{i} A_k$ is not a colored (s,t)-cut in G?

> COLORED (s,t)-CUT VULNERABILITY GAME (CCVG)
> **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**: $\forall D_1 \in \binom{C}{d_1}.\exists A_1 \in \binom{C \setminus D_1}{a_1}.\forall D_2 \in \binom{C\setminus(D_1 \cup A_1)}{d_2}. \cdots .\exists A_i \in \binom{C\setminus((\bigcup_{k=1}^{i-1}(D_k \cup A_k)) \cup D_i)}{a_i}$: such that $\bigcup_{k=1}^{i} A_k$ is a colored (s,t)-cut in G?

By definition, an instance I of CCVG or CCRG is a tuple of an integer i and an instance $\tilde{I}$ of $(\mathrm{DA})^i$-R or $(\mathrm{DA})^i$-V, respectively and we ask, if $\tilde{I}$ is a yes-instance. Hence, $(\mathrm{DA})^i$-R (respectively, $(\mathrm{DA})^i$-V) is a special case of CCRG (respectively, CCVG) for all $i \geq 1$. Note that we could also define problems in which the attacker starts and afterwards every agent has i turns for an unbounded $i \geq 1$. But since $(\mathrm{DA})^i$-R is a special case of $\mathrm{A(DA)}^i$-R which is a special case of $(\text{DA})^{i+1}$-R, a problem in which the attacker starts would be equivalent to CCRG (respectively, CCVG).

In the following we will show that CCRG and CCVG are PSPACE-complete. More precisely, we show that both CCRG and CCVG are already PSPACE-complete if every budget in every turn is exactly one. Therefore, we introduce the following problem.

> Unit Budget Colored (s,t)-Cut Robustness Game (Unit-CCRG)
> **Input**: A colored graph $(G = (V, E), s, t, C, \ell)$ where $|C| = 2i$ for some $i \in \mathbb{N}$.
> **Question**: $\exists d_1 \in C.\forall a_1 \in C \setminus \{d_1\}.\exists d_2 \in C \setminus \{d_1, a_1\}.\cdots.\forall a_i \in C \setminus (\{d_j, a_j \mid 1 \leq j < i\} \cup \{d_i\})$: such that $\{d_j \mid 1 \leq j \leq i\}$ is a colored (s,t)-connector in G?

This problem can also be seen as a Shannon Switching Game [18]. In this case, Unit-CCRG is a game between two agents where 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. Furthermore, we ask if the defender has a winning strategy. In contrast to the Shannon Switching Game where every agent selects an edge every turn instead of a color, which is known to be polynomial-time-solvable [7, 9], we will show that Unit-CCRG is PSPACE-complete.

Lemma 4.16. Unit Budget Colored (s,t)-Cut Robustness Game *is* PSPACE-*hard.*

Proof. We show this statement by a polynomial-time reduction from Shannon Switching Game on the Vertices which is known to be PSPACE-complete [18].

> Shannon Switching Game on the Vertices (SSG-V)
> **Input**: A graph $G = (V, E)$ and two distinct vertices $s, t \in V$ such that $|V \setminus \{s, t\}| = 2i$.
> **Question**: $\exists d_1 \in V \setminus \{s, t\}.\forall a_1 \in V \setminus \{s, t, d_1\}.\exists d_2 \in V \setminus \{s, t, d_1, a_1\}.\cdots.\forall a_i \in V \setminus (\{d_j, a_j \mid 1 \leq j < i\} \cup \{s, t, d_i\})$: such that s and t are in the same connected component in $G[\{d_j \mid 1 \leq j \leq i\}]$?

Given an instance $I = (G = (V, E), s, t)$ of SSG-V, we describe how to construct an instance $I' = (G' = (V', E'), s', t', C, \ell)$ of UNIT-CCRG in polynomial time such that I is a yes-instance of SSG-V if and only if I' is a yes-instance of UNIT-CCRG. We can assume without loss of generality that there is no edge $\{s, t\}$ connecting s and t in E.

Let $\tilde{V} := V \setminus \{s, t\}$. We set $s' := s, t' := t, V' := V \cup \{v_{\{u,w\}} \mid \{u, w\} \in E, \{u, w\} \cap \{s, t\} = \emptyset\}, E' := \{e \in E \mid e \cap \{s, t\} \neq \emptyset\} \cup \{\{u, v_{\{u,w\}}\} \mid \{u, w\} \in E, \{u, w\} \cap \{s, t\} = \emptyset\}$, in other words, we subdivided every edge of E which is not incident to s or t. Note that for every edge $e' \in E'$ there is exactly one $u \in e' \cap \tilde{V}$. Furthermore, we set $C := \tilde{V}$ and set $\ell(\{u, w\}) := u$ for all $\{u, w\} \in E', u \in \tilde{V}$. That is, every vertex $v \in \tilde{V}$ is also a color and all incident edges of v in G' are colored in v.

Next, we show that I is a yes-instance of SSG-V if and only if I' is a yes-instance of UNIT-CCRG. To this end, we show that a set $\tilde{D} \subseteq C$ is a colored (s, t)-connector in G' if and only if s and t are connected in $G[\{s, t\} \cup \tilde{D}]$.

($\Rightarrow$) Let $\tilde{D} \subseteq C = \tilde{V}$ be a colored (s, t)-connector in G'. Then there is an (s, t)-path P' in G' with $\ell(E(P')) \subseteq \tilde{D}$. By construction, $P' = (s, u_1, v_{\{u_1,u_2\}}, u_2, \dots, v_{\{u_{r-1},u_r\}}, u_r, t)$ for some $r \geq 1$ and $u_j \in \tilde{V} = C$ for all $j, 1 \leq j \leq r$. Furthermore, $\ell(E(P')) = \{u_1, \dots, u_r\}$. Hence, $P = (s, u_1, \dots, u_r, t)$ is an (s, t)-path in G and therefore s and t are in the same connected component in $G[\{s, t\} \cup \tilde{D}]$.

($\Leftarrow$) Let $\tilde{D} \subseteq \tilde{V} = C$ such that s and t are in the same connected component in $G[\{s, t\} \cup \tilde{D}]$. Hence, there is an (s, t)-path $P = (s, u_1, \dots, u_r, t)$ with $r \geq 1$ in G where $\{u_1, \dots, u_r\} \subseteq \tilde{D}$. By construction, the path $P' = (s, u_1, v_{\{u_1,u_2\}}, u_2, \dots, v_{\{u_{r-1},u_r\}}, u_r, t)$ is an (s, t)-path in G' with $\ell(E(P')) = \{u_1, \dots, u_r\} \subseteq \tilde{D}$ and therefore $\tilde{D}$ is a colored (s, t)-connector in G'.

Hence, a winning strategy for the defender in the UNIT-CCRG instance I' is also a winning strategy for the defender in the SSG-V instance I and vice versa. Therefore, I is a yes-instance of SSG-V if and only if I' is a yes-instance of UNIT-CCRG. □

Theorem 4.17. COLORED (s, t)-CUT ROBUSTNESS GAME, COLORED (s, t)-CUT VULNERABILITY GAME *and* UNIT BUDGET COLORED (s, t)-CUT ROBUSTNESS GAME *are* PSPACE-*complete.*

Proof. By Lemma 4.16 and the fact that UNIT-CCRG is a special case of CCRG, CCRG is also PSPACE-hard. It remains to show that all these problems are in PSPACE. Let $I = (i, G, s, t, C, \ell, \vec{d}, \vec{a})$ be an instance of CCRG then I is a yes-instance if and only if $I' = (G, s, t, C, \ell, \vec{d}, \vec{a})$ is a yes-instance of $(\mathrm{DA})^i$-R. Due to Theorem 4.10, for all $i \geq 1$ $(\mathrm{DA})^i$-R is contained in Σ_{2i}^{P} and since $\Sigma_{2i}^{\mathrm{P}} \subseteq$ PSPACE, CCRG is contained also in PSPACE.

By the fact that the class of PSPACE-complete problems is closed under complement, CCVG is also PSPACE-complete. □

Recall the property of colored-cut-equivalent graphs. If $\mathcal{H} = (G, s, t, C, \ell)$ and $\mathcal{H}' = (G', s', t', C, \ell')$ are colored-cut-equivalent, 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'. Therefore, the following Claims follow directly.

Claim 4.18. *The* CCVG *instances* $I = (\mathcal{H}, \vec{d}, \vec{a})$ *and* $I' = (\mathcal{H}', \vec{d}, \vec{a})$ *are equivalent if* $\mathcal{H}$ *and* $\mathcal{H}'$ *are colored-cut-equivalent.*

Corollary 4.19. *Both* Colored (s, t)-Cut Robustness Game *and* Colored (s, t)-Cut Vulnerability Game *are* PSPACE-*complete even on planar graphs.*

Proof. We show this statement by a polynomial-time reduction from QSAT-3-CNF to CCVG. To this end, recall the definition.

> QSAT-3-CNF
> **Input**: A set Φ of clauses in 3-CNF over the variables $Z = \{x_j, y_j \mid 1 \leq j \leq i\}$ for some $i \in \mathbb{N}$.
> **Question**: Is it true that $\forall \tilde{x}_1 \in \{x_1, \neg x_1\}.\exists \tilde{y}_1 \in \{y_1, \neg y_1\}. \cdots .\forall \tilde{x}_i \in \{x_i, \neg x_i\}.\exists \tilde{y}_i \in \{y_i, \neg y_i\} : \{\tilde{x}_j, \tilde{y}_j \mid 1 \leq j \leq i\} \models \Phi$?

Given an instance $I = (Z, \Phi)$ of QSAT-3-CNF with $|Z| = 2i$, we construct an instance $I' = (i, \mathcal{H} = (G, s, t, C, \ell), \vec{d}, \vec{a})$ of CCVG where G is planar such that I is a yes-instance of QSAT-3-CNF if and only if I' is a yes-instance of CCVG. We can use the same construction as in Lemma 4.1 to construct $\mathcal{H}$ and the budget vectors $\vec{d}$ and $\vec{a}$ in polynomial time. Afterwards, the proof that I is a yes-instance of QSAT-3-CNF if and only if I' is a yes-instance of CCRG can also be shown identical to the proof in Lemma 4.1. By construction, G is planar and therefore, Colored (s, t)-Cut Robustness Game and Colored (s, t)-Cut Vulnerability Game are PSPACE-complete even on planar graphs. □

5 Parameterizations for Colored Cut Games

In this section we analyze the parameterized complexity of the colored cut games. In Sections 5.1 – 5.3, we investigate parameters that are related to the budget of the agents. In Section 5.4, 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 in Section 5.4 use parameters larger than $|C|$.

Let us introduce some notations that are necessary for the parameterizations of this section. We denote for an instance $I = (G, s, t, C, \ell, \vec{d}, \vec{a})$ of $(\text{DA})^i$-R or $(\text{DA})^i$-V with $\mathcal{B}_d(I) := \{d_j \mid 1 \leq j \leq i\}$ and $\mathcal{B}_a(I) := \{a_j \mid 1 \leq j \leq i\}$ the set of budgets of the defender and attacker, respectively and with $B_d(I) := \sum_{x=1}^{i} d_x$ and $B_a(I) := \sum_{x=1}^{i} a_x$ the sum of the budget of the defender and attacker, respectively. Furthermore, we denote with $\mathcal{B}(I) := \mathcal{B}_d(I) \cup \mathcal{B}_a(I)$ the set of all budgets. Moreover, we set $B(I) := B_d(I) + B_a(I)$. For A(DA)^i-R, A(DA)^i-V, CCRG, and CCVG, we define the functions $\mathcal{B}_d, \mathcal{B}_a, \mathcal{B}, B_d, B_a$, and B analogously.

We use the abbreviation DA-V and DA-R for $(\text{DA})^1$-V and $(\text{DA})^1$-R, respectively. Furthermore, since there is only one turn per agent, we define $a := a_1$ and $d := d_1$.

To determine whether or not $(\text{DA})^i$-R admits an XP-algorithm when parameterized by some parameter k^*, it is sufficient to show that $(\text{DA})^i$-V admits an XP-algorithm when parameterized by k^* and vice versa, since XP is closed under complement [15]. The same holds for FPT-algorithms. Hence, we only investigate parameterizations for either $(\text{DA})^i$-R or $(\text{DA})^i$-V. Clearly, this observation also holds for the other pairs of complement games, A(DA)^i-R and A(DA)^i-V, as well as CCRG and CCVG.

5.1 Parameterization by the Full Budget $B(I)$

First, we investigate the parameter $B(I)$. Colored (s,t)-Cut is in XP and W[2]-hard when parameterized by $k = B(I)$ due to Corollary 2.5. Similar to Colored (s,t)-Cut, we will show that all the colored cut games are in XP and W[2]-hard when parameterized by $B(I)$. Moreover, we show that all colored cut games are in FPT and do not admit polynomial kernels when parameterized $|C|$.

Proposition 5.1. $(\text{DA})^i$-R*,* $i \geq 1$*,* A(DA)^i-R*,* $i \geq 0$*, and* CCRG *parameterized by* $B(I)$ *are in* XP*.*

Proof. There is a trivial XP-algorithm for $(\mathrm{DA})^i$-R when parameterized by $B(I)$. The idea of this algorithm is to compute an and-or tree of depth $2i+1$ where the first i odd levels belong to choices of the defender, and the even levels belong to the attacker. A non-leaf node branches into all possible subsets of unchosen colors of size equal to the budget of the corresponding turn. A leaf is evaluated as true if the union of the choices of the attacker on the path from this leaf to the root is not a colored (s,t)-cut. A node belonging to the defender is evaluated as true if there is at least one child that is evaluated as true, whereas, a node belonging to the attacker is evaluated as true if all its children are evaluated as true. To determine if an instance of $(\mathrm{DA})^i$-R is a yes-instance, we thus have to check if the root of the corresponding and-or tree is evaluated as true. This algorithm clearly runs in time $\mathcal{O}(\binom{|C|}{d_1}\binom{|C|-d_1}{a_1}\dots\binom{|C|-\sum_{j=1}^{i-1}(d_j+a_j)-d_i}{a_i}(n+m)) \subseteq \mathcal{O}(|C|^{B(I)}(n+m))$. $\mathrm{A(DA)}^i$-R and CCRG can be solved analogously. □

By definition, $B(I) \leq |C|$. Hence, the XP-algorithms of Proposition 5.1 also imply FPT-algorithms when parameterized by $|C|$.

Corollary 5.2. $(\mathrm{DA})^i$-R*,* $\mathrm{A(DA)}^{i-1}$-R*,* $i \geq 1$*, and* CCRG *can be solved in* $\mathcal{O}(\min(|C|^{|C|}, 2^{2i|C|})(n+m))$ *time and do not admit a polynomial kernel when parameterized by* $|C|$*, unless* NP $\subseteq$ coNP/poly.

Proof. Since Colored (s,t)-Cut parameterized by $|C|$ does not admit a polynomial compression [38], which is a stronger result than not admit a polynomial kernel [12], unless NP $\subseteq$ coNP/poly, none of the colored cut gamesadmits a polynomial kernel when parameterized by $|C|$, unless NP $\subseteq$ coNP/poly.

With the algorithm described in Proposition 5.1 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 $0 \leq j \leq |C|$ 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. □

Next, we improve the running time of the FPT-algorithm of Corollary 5.2 for DA-R and DA-V.

Proposition 5.3. DA-R *can be solved in* $\mathcal{O}(2^{|C|}(n+m))$ *time.*

Proof. Recall that in DA-R 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 $\tilde{D} \subseteq C$ of size at least a *safe* 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 $|C| - d$. Now, 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 algorithms 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 $a \leq j \leq |C| - d$. 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 $a \leq j < |C| - d$. 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 and therefore we can compute S_{j+1} in $\mathcal{O}(\binom{|C|}{j+1}|C|)$ time. Therefore, 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 since we can assume without loss of generality that $|C| \leq m$. □

Note that this algorithm uses $\mathcal{O}(2^{|C|})$ space and therefore might only be interesting for theoretical use and not in practice.

Finally, we show that it is unlikely to find an FPT-algorithm for any of the colored cut games when parameterized by $B(I)$.

Proposition 5.4. (DA)^i-V*,* $i \geq 1$,A(DA)^i-V*,* $i \geq 0$*, and* CCVG *parameterized by* $B(I)$ *are* W[2]*-hard.*

Proof. 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 (DA)^i-V,A(DA)^i-V, and CCVG 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 [19, 38]. □

5.2 Parameterization by All Budgets Except One

In the previous section, we have shown that for all colored cut games there is an XP-algorithm when the parameter is $B(I)$. Recall that $\mathcal{B}(I)$ denotes the set of all budgets of both agents. In this subsection, we show that it is unlikely to find an XP-algorithm for any of these problems when they are parameterized by $B(I) - b$ for any budget $b \in \mathcal{B}(I)$. In other words, even if all budgets except one sum up to a constant, all these problems cannot be solved in polynomial time, unless P = NP.

The first statement follows directly from the fact that Colored (s,t)-Cut is a special case of DA-V with $d = 0$. Hence, DA-V is NP-hard even if $d = 0$, and therefore DA-R is coNP-hard even if $d = 0$.

Corollary 5.5. DA-R *is* coNP-*hard and* DA-V *is* NP-*hard even if* $d = 0$.

Next, we show that all the introduced games can be solved in polynomial time if the attacker is only allowed to chose at most one color in the entire instance.

Proposition 5.6. *Let I be an instance of* $(\text{DA})^i$-R $i \geq 1$*,* $\text{A}(\text{DA})^i$-R $i \geq 0$*, or* CCRG*, then I can be solved in $\mathcal{O}(i + |C|(n+m))$ time if $B_a(I) \leq 1$.*

Proof. First, we show that DA-R can be solved in $\mathcal{O}(|C|(n+m))$ time if $a \leq 1$. Let $I = (G = (V, E), s, t, C, \ell, d, a)$ be an instance of DA-R. We can assume that s and t are in the same connected component in G, since otherwise I is a trivial no-instance. Hence, if $a = 0$, I is a trivial yes-instance. If $a = 1$, 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 $\mathcal{O}(|C|(n+m))$ 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 CCRG can be solved in $\mathcal{O}(i + |C|(n+m))$ time if $B_a(I) \leq 1$. Let $I = (i, G = (V, E), s, t, C, \ell, (d_1, \ldots, d_i), (a_1, \ldots, a_i))$ be an instance of CCRG with $B_a(I) \leq 1$. Recall that if $B_a(I) = 0$, the defender wins if and only if s and t are in the same connected component in G. Thus, assume that $B_a(I) = 1$. Hence, there is some $x, 1 \leq x \leq i$, such that $a_x = 1$ and $a_y = 0$ for all $1 \leq y \leq i, y \neq x$. 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 $(\text{DA})^x$-R 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-R 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, CCRG can be solved in $\mathcal{O}(i+|C|(n+m))$ time. Since, $\text{A}(\text{DA})^i$-R is a special case of CCRG, $(\text{DA})^i$-R and $\text{A}(\text{DA})^i$-R can be solved in the same time. □

In contrast, we now show that $(\text{DA})^i$-R, $\text{A}(\text{DA})^i$-R, $i \geq 1$, and CCRG are NP-hard if the attacker is allowed to choose at least two colors.

Proposition 5.7. DA-R *is* NP-*complete even if* $a = 2$.

Proof. We reduce from Vertex Cover. Let $I' = (G' = (V', E'), k')$ be an instance of Vertex Cover. We construct in polynomial time an instance $I = (G = (V, E), s, t, C, \ell, d, a)$ of DA-R with $a = 2$ such that I is a yes-instance if and only if I' is a yes-instance.

Assume $E' := \{e_1, \dots e_{|E'|}\}$ and an ordering on V' and set

- $C := V'$,
- $V := \{v_0, v_i^<, v_i^>, v_i \mid 1 \leq i \leq |E'|\}, s := v_0, t := v_{|E'|}$, and
- $E := \{\{v_{i-1}, v_i^<\}, \{v_{i-1}, v_i^>\}, \{v_i^<, v_i\}, \{v_i^>, v_i\} \mid 1 \leq i \leq |E'|\}$.

Furthermore, we set $\ell(\{v_{i-1}, v_i^<\}) := \ell(\{v_i^<, v_i\}) := u_i$ and $\ell(\{v_{i-1}, v_i^>\}) := \ell(\{v_i^>, v_i\}) := w_i$ for all $e_i = \{u_i, w_i\}, u_i < w_i$. That is, we connect the vertices v_{i-1} and v_i with two paths $P^<$ and $P^>$ such that $\ell(E(P^<)) = u_i$ and $\ell(E(P^>)) = w_i$. Finally, we set $d = k'$ and $a = 2$.

Next, we show that I is a yes-instance of DA-R if and only if I' is a yes-instance of Vertex Cover. By construction, $v_i \in V(P)$ for every (s,t)-path P in G and for every $i, 0 \leq i \leq |E'|$. By removing the edges $\ell^{-1}(\{u_i, w_i\})$ from G for some $\{u_i, w_i\} \in E'$, the vertices v_{i-1} and v_i are in different connected components. Thus, $\{u_i, w_i\}$ is a colored (v_{i-1}, v_i)-cut in G and therefore also a colored (s,t)-cut in G. Therefore, the defender has to pick D_1 such that $D_1 \cap \{u,v\} \neq \emptyset$ for all $e \in E'$ (which is a colored (s,t)-connector in G) or otherwise, the attacker can pick a colored (s,t)-cut of size two. Consequently, D_1 is a colored (s,t)-connector in G if and only if D_1 is a vertex cover in G'. Thus, I is a yes-instance of DA-R if and only if I' is a yes-instance of Vertex Cover.

It remains to show that DA-R is contained in NP if $a = 2$. Let $I = (G = (V,E), s, t, C, \ell, d, a)$ be an instance of DA-R with $a = 2$. The defender has a winning strategy if and only if the attacker has no winning strategy, that is, if the defender can choose $D_1 \in \binom{C}{d}$ such that there is no colored (s,t)-cut $A_1 \subseteq C \setminus D_1$ of size at most two in G. Since we can compute the set $\mathcal{F}$ of all colored (s,t)-cuts of size at most two in $\mathcal{O}(\binom{|C|}{2}(n+m))$ we can construct the instance $I^* = ((C,\mathcal{F}), d)$ of Hitting Set in polynomial time. Clearly, I is a yes-instance if and only if I^* is a yes-instance. We gave a reduction from DA-R with $a = 2$ to Hitting Set and therefore DA-R with $a = 2$ is contained in NP since Hitting Set is contained in NP. □

The next theorem follows from Propositions 5.6, and 5.7, and the fact that DA-R is a special case of $(\mathrm{DA})^i$-R, $\mathrm{A(DA)}^i$-R, $i \geq 1$, and CCRG.

Theorem 5.8. *The problems* $(\mathrm{DA})^i$-R*,* $\mathrm{A(DA)}^i$-R*,* $i \geq 1$*, and* CCRG *can be solved in* $\mathcal{O}(i + |C|(n+m))$ *time if* $B_a(I) \leq 1$ *and are* NP*-hard otherwise.*

With Corollary 5.5 and Proposition 5.7, we are now able to show that it is unlikely to find an FPT-algorithm for any of the colored cut games when parameterized by $B(I) - b$ for any budget $b \in \mathcal{B}(I)$.

Corollary 5.9. $(\mathrm{DA})^i$-R*,* $i \geq 1$*,* $\mathrm{A(DA)}^i$-R*,* $i \geq 0$ *and* CCRG *parameterized by* $B(I) - b$ *are not in* XP *for every* $b \in \mathcal{B}(I)$*, unless* P = NP*.*

Proof. Let $1 \leq x \leq i$. We show that $(\mathrm{DA})^i$-R is NP-hard even if $B(I) - d_x = 2$ and coNP-hard even if $B(I) - a_x = 0$. In both cases we reduce from DA-R.

Given an instance $I' = (G, s, t, C, \ell, d, a)$ of DA-R with $a = 2$, we define the instance $I := (G, s, t, C, \ell, (d_1, \dots, d_i), (a_1, \dots, a_i))$ with $a_x = 2, d_x = d$, and $a_j := d_j := 0$ for all $1 \leq j \leq i, j \neq x$. By definition, $B(I) - d_x = 2$. Moreover, I and I' are equivalent. Hence, $(\mathrm{DA})^i$-R is NP-hard even if $B(I) - d_x = 2$ since DA-R is NP-hard even if $a = 2$ due to Proposition 5.7.

Given an instance $I'' = (G, s, t, C, \ell, d, a)$ of DA-R with $d = 0$, we define the instance $I := (G, s, t, C, \ell, (d_1, \dots, d_i), (a_1, \dots, a_i))$ with $a_x = a, d_x = 0$, and $a_j := d_j := 0$ for all $1 \leq j \leq i, j \neq x$. By definition, $B(I) - a_x = 0$

and I and I'' are equivalent instances. Hence, $(\mathrm{DA})^i$-R is coNP-hard even if $B(I) - a_x = 0$ since DA-R is coNP-hard even if $d = 0$ due to Corollary 5.5.

Hardness for $\mathrm{A}(\mathrm{DA})^i$-R and CCRG can be shown in the same way. □

The next theorem follows from Propositions 5.1, and 5.4 and Corollary 5.9.

Theorem 5.10. *The problems* $(\mathrm{DA})^i$-R*,* $\mathrm{A}(\mathrm{DA})^i$-R*,* $i \geq 1$*, and* CCRG *are in* XP *and* W[2]*-hard when parameterized by* $B(I)$ *and not in* XP *when parameterized by* $B(I) - b$ *for every budget* $b \in \mathcal{B}(I)$*, unless* P = NP*.*

5.3 Parameterization by the Number of Unchosen Colors

As the last parameterization that is related to the budget, we investigate $|C| - b$ for any $b \in \mathcal{B}(I)$. That is, the number of colors that cannot be chosen in any turn with budget b. Recall that in the case of DA-R, $\mathcal{B}(I) = \{d, a\}$. Thus, for DA-R we investigate parameterizations by $|C| - d$ and $|C| - a$ and generalize the hardness results we obtain to all colored cut games.

Proposition 5.11. $(\mathrm{DA})^i$-R*,* $i \geq 1$*,* $\mathrm{A}(\mathrm{DA})^i$-R*,* $i \geq 0$*, and* CCRG *are in* XP *when parameterized by* $|C| - b$ *for every* $b \in \mathcal{B}(I)$*.*

Proof. Let I be an instance of $(\mathrm{DA})^i$-R and let $b \in \mathcal{B}(I)$ be an arbitrary budget. With the algorithm described in Proposition 5.1 and the fact that $B(I) \leq |C|$ we can solve I in time $\mathcal{O}(|C|^{|C|-b}\binom{|C|}{b}(n+m)) = \mathcal{O}(|C|^{|C|-b}\binom{|C|}{|C|-b}(n+m)) \subseteq \mathcal{O}(|C|^{|C|-b}|C|^{|C|-b}(n+m))$ which implies an XP-algorithm for $(\mathrm{DA})^i$-R parameterized by $|C| - b$. The statements for $\mathrm{A}(\mathrm{DA})^i$-R and CCRG can be shown analogously. □

Next, we show that it is unlikely to find an FPT-algorithm for any of the colored cut games when parameterized by $|C| - b$ for $b \in \mathcal{B}_d(I)$. To this end, we first show the fixes-parameter intractability for DA-R parameterized by $|C| - d$.

Corollary 5.12. DA-R *parameterized by* $|C| - d$ *is* W[1]*-hard.*

Proof. In the proof of Proposition 5.7 we gave an implicit parameterized reduction from the Independent Set problem parameterized by the size of the solution (which is known to be W[1]-hard [15]) to DA-R parameterized by $|C| - d$. Hence, DA-R parameterized by $|C| - d$ is also W[1]-hard. □

To analyze the parameterized complexity of DA-R parameterized by $|C| - a$, we first show the following connection to COLORED PATH.

Lemma 5.13. *An instance $I = (G, s, t, C, \ell, d, a)$ of* DA-R *with $d + a = |C|$ is a yes-instance if and only if the instance $I' = (G, s, t, C, \ell, d)$ of* COLORED PATH *is a yes-instance.*

Proof. If I is a yes-instance, then there is $D_1 \in \binom{C}{d}$ such that s and t are in the same connected component in $G - \ell^{-1}(C \setminus D_1)$. Hence, there is an (s,t)-path in $G - \ell^{-1}(C \setminus D_1)$ and therefore also in G with $\ell(E(P)) \subseteq D_1$. By construction, D_1 is a colored (s,t)-connector in G.

If I' is a yes-instance, then there is a colored (s,t)-connector L of size at most d in G and therefore choosing $D_1 \supseteq L$ is a winning strategy for the defender. □

Since COLORED PATH is NP-complete and W[2]-hard when parameterized by the size of the solution k [19], the following also holds.

Corollary 5.14. DA-R *parameterized by $|C| - a$ is* W[2]*-hard and* NP*-complete if $d + a = |C|$.*

Since DA-R is a special case of $(\mathrm{DA})^i$-R, $\mathrm{A(DA)}^i$-R, and CCRG, the next result follows directly from the Corollaries 5.12 and 5.14.

Corollary 5.15. $(\mathrm{DA})^i$-R*, $i \geq 1$,* $\mathrm{A(DA)}^i$-R*, $i \geq 1$, and* CCRG *parameterized by $|C| - b_d$ are* W[1]*-hard for every $b_d \in \mathcal{B}_d(I)$ and* W[2]*-hard when parameterized by $|C| - b_a$ for every $b_a \in \mathcal{B}_a(I)$.*

The main theorem of this subsection now follows from Proposition 5.11 and Corollary 5.15.

Theorem 5.16. $(\mathrm{DA})^i$-R*,* $\mathrm{A(DA)}^i$-R*, $i \geq 1$, and* CCRG *parameterized by $|C| - b$ are in* XP *for every $b \in \mathcal{B}(I)$ and* W[1]*-hard if $b \in \mathcal{B}_d(I)$ and* W[2]*-hard if $b \in \mathcal{B}_a(I)$.*

5.4 Polynomial Kernels for Parameters Larger than the Number of Colors

Since it is unlikely that any of the colored cut games admits a polynomial kernel when parameterized by $|C|$, we have to investigate (comined) parameters that are larger than $|C|$ in order to obtain polynomial kernels.

Recall that unless P = NP. Colored (s, t)-Cut parameterized by the vertex cover number does not admit an XP-algorithm [10] and therefore neither does any of the colored cut games. We will show that all the colored cut games admit polynomial kernels when parameterized by both $|C|$ and the vertex cover number. More general, we show that even for smaller parameters we can find polynomial kernels. To this end, we consider a generalization of the vertex cover number to the r-COC number.

Definition 5.17 ([16, 28])**.** Let $G = (V, E)$ be a graph and $r \geq 1$ be an integer. We call $\Gamma \subseteq V$ a *r-component order connectivity set* (r-COC set) if for all connected components H in $G - \Gamma$ it holds that $|H| \leq r$. Furthermore, we call $\kappa_r := \min\{|\Gamma| \mid \Gamma \subseteq V, \Gamma \text{ is a } r\text{-COC set}\}$ the *r-component order connectivity number* (r-COC number) of G.

Note that a set $\Gamma \subseteq C$ is a 1-COC set if and only if Γ is a vertex cover.

Lemma 5.18. *Let $r \geq 1$ be a constant and let $\mathcal{H} = (G = (V, E), s, t, C, \ell)$ be a colored graph with r-COC number κ_r. Then, there is a colored-cut-equivalent graph $\mathcal{H}' = (G' = (V', E'), s', t', C, \ell')$ with $|\mathcal{H}'| \leq (\kappa_r + |C|)^{\mathcal{O}(r)}$ that can be computed in $|\mathcal{H}|^{\mathcal{O}(r)}$ time.*

The idea of the following construction is to first compute an r-COC set Γ in polynomial time such that $s \in \Gamma$ and $t \in \Gamma$. Second, we compute all color sets $A_{\{a,b\}}$ of (a, b)-paths in the graph induced by for all distinct $a, b \in \Gamma$. Finally, we remove all connected components in $G - \Gamma$ and add an (a, b)-paths with the colors L for all $a, b \in \Gamma$ and for all $L \in A_{\{a,b\}}$. Thus, we construct a colored graph $\mathcal{H}'$ in polynomial time where the number of connected components in $G' - \Gamma$ is bounded from above by a polynomial function only depending on $|C|$ and $|\Gamma|$.

Proof. We can find an r-COC set Γ' of size at most $\kappa_r(r + 1)$ in $\mathcal{O}(n + m)$ time due to the $(r+1)$-approximation algorithm for r-COC [28]. Afterwards, we set $\Gamma := \Gamma' \cup \{s, t\}$ which is an r-COC set of size at most $\kappa_r(r + 1) + 2$. Let q be the number of connected components in $G - \Gamma$ and $H_1, \ldots, H_q \subseteq V$ be the vertex sets of the connected components in $G - \Gamma$. Recall that $|H_i| \leq r$ for all $1 \leq i \leq q$. Therefore, for every $\{a, b\} \in \binom{\Gamma}{2}$ and every (a, b)-path P in G with $V(P) \cap \Gamma = \{a, b\}$ it holds that $|V(P)| \leq r + 2$. The idea is that we compute for every $\{a, b\} \in \binom{\Gamma}{2}$ the set $\mathcal{P}_{\{a,b\}}$ of all (a, b)-paths in $G - (\Gamma \setminus \{a, b\})$. This can be done in $\mathcal{O}(\binom{|\Gamma|}{2} q \cdot r^r)$ time by computing all (a, b)-paths in $G[\{a,b\} \cup H_i]$ for all $\{a,b\} \in \binom{\Gamma}{2}$ and all $i, 1 \leq i \leq q$ since there are at most r^r many (a,b)-paths in $G[\{a,b\} \cup H_i]$. Note that $G[\{a,b\} \cup H_i]$ might also contain a direct edge $\{a,b\}$. Then, we compute the sets $A_{\{a,b\}} := \{\ell(E(P)) \mid P \in \mathcal{P}_{\{a,b\}}\}$ of color sets of paths in $\mathcal{P}_{\{a,b\}}$ in time $\mathcal{O}(\binom{|\Gamma|}{2}(q \cdot r^r + |C|^{r+1}))$, since there are at most $|C|^{r+1}$ paths with different sets of colors of size at most $r+1$. Furthermore, assume an ordering on every $L \in A_{\{a,b\}}, \{a,b\} \in \binom{\Gamma}{2}$ and an ordering on the vertices of Γ.

Now, we define the instance $\mathcal{H}'$. We start with an empty graph and add all vertices of Γ, set $s' = s$, and $t' = t$. Next, for every set $\{a,b\} \in \binom{\Gamma}{2}, a < b$, and $L \in A_{\{a,b\}}$ we add vertices $v_{L,1}^{\{a,b\}}, \ldots, v_{L,r}^{\{a,b\}}$ and edges $\{a, v_{L,1}^{\{a,b\}}\}, \{v_{L,r}^{\{a,b\}}, b\}$, and $\{v_{L,i}^{\{a,b\}}, v_{L,i+1}^{\{a,b\}}\}$ for all $i, 1 \leq i < r$. Let $L(y)$ denote the yth color in L. We set $\ell'(\{a, v_{L,1}^{\{a,b\}}\}) := L(1), \ell'(\{v_{L,r}^{\{a,b\}}, t\}) := L(|L|)$, and $\ell'(\{v_{L,i}^{\{a,b\}}, v_{L,i+1}^{\{a,b\}}\}) := L(\min(i+1, |L|)$ for all $i, 1 \leq i < r$. This finishes the definition of $\mathcal{H}'$.

Note that for $P_L^{\{a,b\}} = (a, v_{L,1}^{\{a,b\}}, \ldots, v_{L,r}^{\{a,b\}}, b)$ it holds that $\ell'(E(P_L^{\{a,b\}})) = L$. By construction, no edge $e \in E$ belongs to E', and therefore every (a,b)-path in $G' - (\Gamma \setminus \{a,b\})$ is of the form $P_L^{\{a,b\}}$ for some $L \in A_{\{a,b\}}$. Therefore, there is an (a,b)-path P in $G - (\Gamma \setminus \{a,b\})$ with $\ell(E(P)) = L$ if and only if there is an (a,b)-path $P_L^{\{a,b\}}$ in $G' - (\Gamma \setminus \{a,b\})$ with $\ell'(E(P_L^{\{a,b\}})) = L$. Since every path $P_L^{\{a,b\}}$ contains exactly $r+1$ edges and there are at most $\binom{|\Gamma|}{2}|C|^{r+1}$ such paths, the number of edges in G' is at most $\binom{|\Gamma|}{2}|C|^{r+1}(r+1) \leq (\kappa_r)^2(r+1)^3|C|^{r+1}$ and the number of vertices is at most $|\Gamma| + \binom{|\Gamma|}{2}|C|^{r+1}r \leq ((\kappa_r)^2 + \kappa_r)(r+1)^3|C|^{r+1}$. Hence, $|\mathcal{H}'| \leq (\kappa_r + |C|)^{\mathcal{O}(r)}$. Furthermore, $\mathcal{H}'$ and can be computed in $|\mathcal{H}|^{\mathcal{O}(r)}$ time.

Finally, we show that $\mathcal{H}$ and $\mathcal{H}'$ are colored-cut-equivalent. To this end, we prove 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 $L_{\mathcal{H}'} \in \mathcal{C}(\mathcal{H}')$ such that $L_{\mathcal{H}'} \subseteq L_{\mathcal{H}}$) and that for every $L_{\mathcal{H}'} \in \mathcal{C}(\mathcal{H}')$ there is $L_{\mathcal{H}} \in \mathcal{C}(\mathcal{H})$ such that $L_{\mathcal{H}} \subseteq L_{\mathcal{H}'}$.

First, we show $\mathcal{C}(\mathcal{H}) \subseteq \mathcal{C}(\mathcal{H}')$. Let $P = (u_1, \ldots, u_q)$ be an (s,t)-path in G with $u_1 = s$ and $u_q = t$. We show that there is an (s,t)-path P' in G' with $\ell'(E(P')) = \ell(E(P))$. Let $\{u_{i_1}, \ldots, u_{i_z}\} := \{u \in V(P) \mid u \in \Gamma\}$ such that $i_j < i_{j+1}$ for all $j, 1 \leq j < z$. Recall that $s, t \in \Gamma$, which implies that $i_1 = 1$ and $i_z = q$. Now, let $P_j := (u_{i_j}, u_{i_j+1}, \ldots, u_{i_{(j+1)}-1}, u_{i_{(j+1)}})$ for all $1 \leq j < z$. By construction, $|V(P_j)| \leq r+2$ and also $L_j := \ell(E(P_j)) \in A_{\{u_{i_j}, u_{i_{j+1}}\}}$. Hence, for all $j, 1 \leq j < z$, G contains a path $P_{L_j}^{\{u_{i_j}, u_{i_{j+1}}\}}$ with $\ell'(E(P_{L_j}^{\{u_{i_j}, u_{i_{j+1}}\}})) = L_j$. Therefore, there is an (s,t)-path P' in G'

with $\ell'(E(P')) = \ell(E(P))$ and we get $\mathcal{C}(\mathcal{H}) \subseteq \mathcal{C}(\mathcal{H}')$.

It remains to show that for every $L_{\mathcal{H}'} \in \mathcal{C}(\mathcal{H}')$ there is $L_{\mathcal{H}} \in \mathcal{C}(\mathcal{H})$ such that $L_{\mathcal{H}} \subseteq L_{\mathcal{H}'}$. Let P' be an (s,t)-path in G'. We know by construction that $P' = P'_1 \multimap \cdots \multimap P'_{q-1}$ with $V(P'_y) = \{u'_y, v^{\{u'_y,u'_{y+1}\}}_{L_y,1}, \ldots, v^{\{u'_y,u'_{y+1}\}}_{L_y,r}, u'_{y+1}\}$ for some $u'_y, u'_{y+1} \in \Gamma, 1 \leq y < q$ and $L_y \in A_{\{u'_y,u'_{y+1}\}}, 1 \leq y < q$. Note that $\ell'(E(P'_j)) = L_y$. Then, for every $y, 1 \leq y < q$, there is also an (u'_y, u'_{y+1})-path P_y in G with $\ell(E(P)) = L_y$. By connecting all paths $P_1 \multimap \cdots \multimap P_{q-1}$, we get an (s,t)-path P in G with $\ell(E(P)) = \bigcup_{j=1}^{q-1} \ell(E(P_j)) = \ell'(E(P'))$. Note that P might not be vertex-simple. But then we know from Proposition 2.1 that there is a vertex-simple (s,t)-path $\tilde{P}$ in G with $\ell(E(\tilde{P})) \subseteq \ell(E(P)) = \ell'(E(P'))$. Hence, $\ell(E(\tilde{P})) \in \mathcal{C}(\mathcal{H})$. Thus, there is some $L_{\mathcal{H}} \in \mathcal{C}(\mathcal{H})$ such that $L_{\mathcal{H}} \subseteq L_{\mathcal{H}'}$ for every $L_{\mathcal{H}'} \in \mathcal{C}(\mathcal{H}')$. □

With this lemma we can now show the kernelization algorithm.

Theorem 5.19. *Let $r \geq 1$ be a constant, then* (DA)^i-R, $i \geq 1$, A(DA)^i-R, $i \geq 0$, *and* CCRG *parameterized by the r-COC number κ_r and $|C|$ admit a polynomial kernel of size $(\kappa_r + |C|)^{\mathcal{O}(r)}$ that can be computed in $|I|^{\mathcal{O}(r)}$ time.*

Proof. Let $I = (\mathcal{H}, \vec{d}, \vec{a})$ be an instance of (DA)^i-R. By Lemma 5.18 we can compute in $|I|^{\mathcal{O}(1)}$ time a colored graph $\mathcal{H}'$ which is colored-cut-equivalent to $\mathcal{H}$ such that $|\mathcal{H}'| \leq (\kappa_r + |C|)^{\mathcal{O}(1)}$. Due to Claim 4.11, I is a yes-instance if and only if $I' = (\mathcal{H}', \vec{d}, \vec{a})$ is a yes-instance and therefore (DA)^i-R admits a polynomial kernel when parameterized by the r-COC number κ_r and $|C|$. The statements for A(DA)^i-R and CCRG can be shown analogously, since by Claim 4.12 and 4.18, one can replace the colored graph $\mathcal{H}$ of an instance of A(DA)^i-R or CCRG with a colored-cut-equivalent graph $\mathcal{H}'$ and obtain an equivalent instance. □

It would be also 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 in the standard reduction from Hitting Set the vertex deletion distance to degree two is only two. 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. Due to Corollary 2.5, such a kernel does not exists, unless NP $\subseteq$ coNP/poly.

Finally, we also show a polynomial kernel for all colored cut games when parameterized by both $|C|$ and the number of (s,t)-paths p.

Lemma 5.20. *Let $\mathcal{H} = (G = (V, E), s, t, C, \ell)$ be a colored graph and let p be the number of (s,t)-paths in G. Then, there is a colored-cut-equivalent graph $\mathcal{H}' = (G' = (V', E'), s', t', C, \ell')$ with $|\mathcal{H}'| \leq \mathcal{O}(p|C|)$ that can be computed in $\mathcal{O}(p(n + |C|) + m)$ time.*

Proof. Recall that we can compute $\mathcal{C}(\mathcal{H})$ in $\mathcal{O}(p(n + |C|) + m)$ time due to Proposition 3.5. We start with a graph only containing the vertices s' and t'. Next, we can assume an order on every $L \in \mathcal{C}(\mathcal{H})$ and add the vertices $v_1^L, \dots, v_{|L|}^L$ and the edges $\{s', v_1^L\}, \{v_{|L|}^L, t'\}$, and $\{v_i^L, v_{i+1}^L\}$ for all $1 \leq i < |L|$. Furthermore, let $L(y)$ denote the yth element of L. We set $\ell'(\{s', v_1^L\}) := L(1), \ell'(\{v_L^L, t'\}) := L(|L|)$, and $\ell'(\{v_i^L, v_{i+1}^L\}) := L(i + 1)$ for all $1 \leq i < |L|$. That is, we added for every $L \in \mathcal{C}(\mathcal{H})$ the (s', t')-path $P^L = (s', v_1^L, \dots, v_{|L|}^L, t')$ in G' such that $\ell'(E(P^L)) = L$. By construction, $|E(P^L)| = |L|+1$ to prevent parallel edges.

Next, we show that $\mathcal{H}$ and $\mathcal{H}'$ are colored-cut-equivalent. To this end, we show that $\mathcal{C}(\mathcal{H}) = \mathcal{C}(\mathcal{H}')$. Let $L \in \mathcal{C}(\mathcal{H})$, then P^L is an (s', t')-path in G' with $\ell'(E(P^L)) = L$. By construction, every (s', t')-path is of the form P^L for some $L \in \mathcal{C}(\mathcal{H})$. Hence, $L \in \mathcal{C}(\mathcal{H})$ if and only if $L \in \mathcal{C}(\mathcal{H}')$.

Finally, we show that $|\mathcal{H}'| \leq \mathcal{O}(p|C|)$. Clearly, $|\mathcal{C}(\mathcal{H})| \leq p$ and $|L| \leq |C|$ for all $L \in \mathcal{C}(\mathcal{H})$. Therefore, every path P^L contains exactly $|L| + 1$ edges and $|L|$ vertices distinct from s and t. Hence, $|V'| \leq p|C| + 1$ and $|E'| \leq p(|C| + 1)$. □

Corollary 5.21. $(\text{DA})^i$-R, $i \geq 1$, $\text{A}(\text{DA})^i$-R, $i \geq 0$, *and* CCRG *parameterized by the number of (s,t)-paths p and $|C|$ admit a polynomial kernel of size at most $\mathcal{O}(p|C|)$ that can be computed in $\mathcal{O}(p^2(n + |C|) + pm)$ time.*

Proof. Let $I = (G, s, t, C, \ell, \vec{d}, \vec{a})$ be an instance of $(\text{DA})^i$-R. We can compute p in $\mathcal{O}(p(pn + m))$ time by checking if there are exactly p' many (s,t)-paths in G for all $0 \leq p' \leq p$ [4]. By Lemma 5.20 we can compute in $\mathcal{O}(p(n + |C|) + m)$ time a colored graph (G', s', t', C, ℓ') which is colored-cut-equivalent to (G, s, t, C, ℓ). Hence, due to Claim 4.11, I is a yes-instance if and only if $I' = (G', s', t', C, \ell', \vec{d}, \vec{a})$ is a yes-instance and therefore $(\text{DA})^i$-R admits a kernel of size at most $\mathcal{O}(p|C|)$ when parameterized by the number of (s,t)-paths p and $|C|$. The statements for $\text{A}(\text{DA})^i$-R and CCRG can be shown analogously, since by Claim 4.12 and 4.18, one can replace the colored graph $\mathcal{H}$ of an instance of $\text{A}(\text{DA})^i$-R or CCRG with a colored-cut-equivalent graph $\mathcal{H}'$ and obtain an equivalent instance. □

6 Restricted Instances of Colored Cut Games

In this section we take a closer look at the classic complexity of $(\text{DA})^i$, $\text{A}(\text{DA})^i$, and CCRG on restricted instances. First, we investigate restricted graph classes like cubic and complete graphs. Second, we analyze two restricted classes of colored graphs for which Colored (s,t)-Cut is polynomial-time-solvable and show that DA-R is NP-complete on these restricted colored graphs.

6.1 Computational Complexity on Graphs with Restricted Degree

First, we show that $(\text{DA})^i$-R, $\text{A}(\text{DA})^i$-R, and $\text{CCRG}, i \geq 1$, 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 P = NP. Second, we show that none of these problems can be solved in polynomial time on complete graphs.

Recall that $(\text{DA})^i$-R, $\text{A}(\text{DA})^i$-R, and CCRG ask if the defender has a wining strategy, whereas $(\text{DA})^i$-V, $\text{A}(\text{DA})^i$-V, and CCVG ask if the attacker has a winning strategy.

Claim 6.1. *Let* $I = (G, s, t, C, \ell, (d_1, \dots, d_i), (a_1, \dots, a_i))$ *be an instance of* $(\text{DA})^i$-R *and let* $j = \min(\{i\} \cup \{k \mid 1 \leq k \leq i, \sum_{r=1}^{k} a_r \geq |\mathcal{C}(I)|\})$, *that is, the first turn in which the sum of the budget of the attacker so far is at least the number of color sets of vertex-simple* (s,t)*-paths in* G. *Then* I *is a yes-instance if and only if the instance* $I' = (G, s, t, C, \ell, (d_1, \dots, d_j), (a_1, \dots, a_j))$ *of* $(\text{DA})^j$-R *is a yes-instance.*

Proof. The attacker can intersect at least one $L \in \mathcal{C}(I)$ for each color he choses. Hence, after choosing at most $|\mathcal{C}(I)|$ colors, he has completed a colored (s,t)-cut, unless the defender has chosen a colored (s,t)-connector before. Therefore, the outcome of the game is determined at the latest after the attackers jth turn of the attacker since either he completed a colored (s,t)-cut in G or the defender has already completed a colored (s,t)-connector in G. □

Proposition 6.2. $(\text{DA})^1$-R *and* $(\text{DA})^2$-R *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 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.

First, let $I_1 = (\mathcal{H}, d, a)$ be an instance of $(\text{DA})^1$-R. We show that we can solve I_1 in polynomial time. Note that I_1 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 Proposition 5.6, 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.

Finally, let $I_2 = (\mathcal{H}, (d_1, d_2), (a_1, a_2))$ be an instance of $(\text{DA})^2$-R. 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 Proposition 5.6, 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 $(\text{DA})^1$-R instance $(\mathcal{H}, d_1, a_1)$ due to Claim 6.1 and therefore can be solved in polynomial time. Moreover, if $a_1 = 0$, I_2 is equivalent to the $(\text{DA})^1$-R 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.

If $d_1 < |L_1| - d_2 + |L_2| - d_2$ then the defender is not able to choose 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 only win by choosing D_2 such that $(D_1 \cup D_2) \supseteq L_1$ 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. □

We will use this statement as the base case for an induction to show the following.

Corollary 6.3. (DA)^i-R*,* A(DA)^i-R*, and* CCRG *can be solved in polynomial time on graphs with maximum degree at most two.*

Proof. We prove the statement by an induction over i. Note that Proposition 6.2 is 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 (DA)^j-R. If $a_1 = 0$ or $a_2 = 0$, then we can construct an equivalent instance I' of (DA)$^{j-1}$-R 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 (DA)^2-R instance $I_2 = (G, s, t, C, \ell, (d_1, d_2), (a_1, a_2))$ due to Claim 6.1. By Proposition 6.2, we can solve I_2, and therefore also I, in polynomial time.

Since A(DA)^i-R is a special case of (DA)$^{i+1}$-R the statement for A(DA)^i-R follows directly. □

This result might not be very surprising since most problems can be solved in polynomial time on graphs with maximum degree at most two, but we will show next that, for all $i \geq 1$, (DA)^i-R and A(DA)^i-R cannot be solved in polynomial time on graphs with degree at least three, unless P = NP. To this end, we first show the following.

Lemma 6.4. *Let $\mathcal{H} = (G = (V, E), s, t, C, \ell)$ be a colored graph and $\alpha \in C$ such that $\alpha \in L$ for every $L \in \mathcal{C}(\mathcal{H})$. Then, there is a colored-cut-equivalent graph $\mathcal{H}' = (G' = (V', E'), s', t', C, \ell')$ where G' has a maximum degree of three and can be computed in polynomial time.*

Proof. Given a colored graph $\mathcal{H} = (G = (V, E), s, t, C, \ell)$ and $\alpha \in C$ such that $\alpha \in L$ for every $L \in \mathcal{C}(\mathcal{H})$, we describe how to construct a colored graph $\mathcal{H}' = (G' = (V', E'), s', t', C, \ell')$ where G' has a maximum degree of three and show that $\mathcal{H}$ and $\mathcal{H}'$ are colored-cut-equivalent.

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 $b^v_{u_1}, \dots, b^v_{u_r}$ where $N_G(v) = \{u_1, \dots, u_r\}$. 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 $\ell'(\{b^v_w, b^w_v\}) := \ell(\{v, w\})$. This can be done in polynomial time. For every $v \in V, x, y \in N(v)$ 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, $\ell'(E(P^v_{x,y})) = \ell'(E(P^v_x)) = \{\alpha\}$.

By construction, G' has a maximum degree of three 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 $P = (v_0, \dots, v_r)$ 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 $P' = P^{v_0}_{v_1} \cdot P^{v_1}_{v_0,v_2} \cdot \dots \cdot P^{v_{r-1}}_{v_{r-2},v_r} \cdot \overleftarrow{P^{v_r}_{v_{r-1}}}$ is an (s, t)-path in G' and $\ell'(E(P')) \supseteq \bigcup_{j=0}^{r-1} \ell'(\{b^{v_j}_{v_{j+1}}, b^{v_{j+1}}_{v_j}\}) = \bigcup_{j=0}^{r-1} \ell(\{v_j, v_{j+1}\}) = \ell(E(P))$. 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, $\alpha \in \ell(E(P))$ and therefore $\ell'(E(P')) = \ell(E(P))$.

Finally, we show that for every $L_{\mathcal{H}'} \in \mathcal{C}(\mathcal{H}')$ there is $L_{\mathcal{H}} \in \mathcal{C}(\mathcal{H})$ such that $L_{\mathcal{H}} \subseteq L_{\mathcal{H}'}$. Let P' be an (s, t)-path in G'. Then, we know by construction that $P' = P^{v_0}_{v_1} \cdot P^{v_1}_{v_0,v_2} \cdot \dots \cdot P^{v_{r-1}}_{v_{r-2},v_r} \cdot \overleftarrow{P^{v_r}_{v_{r-1}}}$ 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, $P = (v_0, \dots, v_r)$ is an (s, t)-path in G and $\ell(E(P)) \subseteq \ell'(E(P'))$. Note that P might not be vertex-simple. But then we know from Proposition 2.1 that there is a vertex-simple (s, t)-path $\tilde{P}$ in G with $\ell(E(\tilde{P})) \subseteq \ell(E(P)) \subseteq \ell'(E(P'))$. □

It is easy to see that if G is planar, we can also construct G' planar.

We will use this lemma to show that $(\mathrm{DA})^i$-R and $\mathrm{A}(\mathrm{DA})^i$-R are Σ_{2i}^{P}-hard even on cubic graphs.

Proposition 6.5. *For all $i \geq 1$, $(\mathrm{DA})^i$-R and $\mathrm{A}(\mathrm{DA})^i$-R are Σ_{2i}^{P}-hard even on graphs with maximum degree at least three.*

Proof. Since $(\mathrm{DA})^i$-R is a special case of $\mathrm{A}(\mathrm{DA})^i$-R, we only have to show that $(\mathrm{DA})^i$-R is Σ_{2i}^{P}-hard even on graphs with a maximum degree of three. We reduce from $(\mathrm{DA})^i$-R where $a_1' \geq 1$. Given an instance $I' = (G' = (V', E'), s', t', C', \ell', (d_1', \ldots, d_i'), (a_1', \ldots, a_i'))$ of $(\mathrm{DA})^i$-R with $a_1' \geq 1$, we describe how to construct an instance $\tilde{I}$ of $(\mathrm{DA})^i$-R such that I' is a yes-instance if and only of $\tilde{I}$ is a yes-instance.

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 $1 < j \leq i$. Furthermore, let $\mathcal{H} = (G = (V, E), s, t, C, \ell)$.

Next, we show that $I = (\mathcal{H}, (d_1, \ldots, d_i), (a_1, \ldots, a_i))$ is a yes-instance if and only if I' is a yes-instance. By construction, $\{s, s'\} \in E(P)$ for every (s, t)-path P in G and therefore, the defender has to choose α in his first turn because $a_1 \geq 1$ and $\{\alpha\}$ is a colored (s, t)-cut in G. Clearly, the defender has a winning strategy for I if and only if he has a winning strategy for I'.

With Lemma 6.4 and the fact that $\alpha \in L$ for every $L \in \mathcal{C}(I)$ there is a colored graph $\tilde{\mathcal{H}} = (\tilde{G}, \tilde{s}, \tilde{t}, C, \tilde{\ell})$ with a maximum degree of three which is colored-cut-equivalent to $\mathcal{H}$. Hence, by Claim 4.11, I is a yes-instance if and only if $\tilde{I} := (\tilde{\mathcal{H}}, (d_1, \ldots, d_i), (a_1, \ldots, a_i))$ is a yes-instance. Therefore, $\tilde{I}$ is a yes-instance if and only if I' is a yes-instance. Since $\tilde{I}$ can be computed in polynomial time, $(\mathrm{DA})^i$-R and $\mathrm{A}(\mathrm{DA})^i$-R are Σ_{2i}^{P}-hard even on graphs with maximum degree at least three. $\square$

The next theorem follows from Corollary 6.3 and Proposition 6.5.

Theorem 6.6. *The problems $(\mathrm{DA})^i$-R, $\mathrm{A}(\mathrm{DA})^i$-R, $i \geq 1$, and* CCRG *can be solved in polynomial time on graphs with a maximum degree of at most two and cannot be solved in polynomial time on graphs with degree at least three, unless* $\mathrm{P} = \mathrm{NP}$.

With the constructions of Proposition 4.13 and Proposition 6.5 to compute colored-cut-equivalent graphs, we are also able to show the following.

Corollary 6.7. *For all $i \geq 1$, $(\mathrm{DA})^i$-R and $\mathrm{A}(\mathrm{DA})^i$-R are Σ_{2i}^{P}-hard even on bipartite planar subcubic graphs.*

Proof. Since $(\mathrm{DA})^i$-R is a special case for $\mathrm{A}(\mathrm{DA})^i$-R, we only have to show that $(\mathrm{DA})^i$-R is Σ_{2i}^{P}-hard even on bipartite, planar and cubic graphs.

By Corollary 4.14, $(\mathrm{DA})^i$-R is Σ_{2i}^{P}-hard even on planar graphs. Let $I = (\mathcal{H} = (G, s, t, C, \ell), \vec{d}, \vec{a})$ be an instance of $(\mathrm{DA})^i$-R where G is planar. If we apply the construction of Proposition 6.5 on I, we get an equivalent instance $\tilde{I} := (\tilde{\mathcal{H}} = (\tilde{G}, \tilde{s}, \tilde{t}, C, \tilde{\ell}), \vec{d'}, \vec{a'})$ of $(\mathrm{DA})^i$-R where $\tilde{G}$ is subcubic. Note that $\tilde{G}$ is also planar, since G is planar and we only replaced every vertex by an binary tree and added one edge to a new vertex. By Proposition 4.13, we can construct a colored-cut-equivalent graph $\mathcal{H}' = (G', s', t', C', \ell')$ to $\tilde{\mathcal{H}}$ in polynomial time such that G' is bipartite.

By construction of Proposition 4.13, G' was constructed by subdividing every edge of $\tilde{G}$. Hence, G' is still planar, and subcubic. By Claim 4.11 $\tilde{I}$ is an equivalent instance to $I' = (\mathcal{H}', \vec{d'}, \vec{a'})$. Hence, I' is also equivalent to I. Moreover, I' can be computed in polynomial time. Hence, $(\mathrm{DA})^i$-R is Σ_{2i}^{P}-hard even on bipartite planar subcubic graphs. □

The same construction can be used in combination with Theorem 4.19 and Claim 4.18 to show the following corollary.

Corollary 6.8. Colored (s,t)-Cut Vulnerability Game *and* Colored (s,t)-Cut Robustness Game *are* PSPACE-*complete even on bipartite planar cubic graphs.*

Finally, we analyze the computational complexity for $(\mathrm{DA})^i$-V, $\mathrm{A}(\mathrm{DA})^i$-V, and CCVG on complete graphs. To this end, we use techniques similar to the ones we used to show Proposition 6.5.

Lemma 6.9. *Let $\mathcal{H} = (G = (V, E), s, t, C, \ell)$ be a colored graph and $\alpha \in C$ such that $\{\alpha\} \in \mathcal{C}(\mathcal{H})$. Then, there is a colored-cut-equivalent graph $\mathcal{H}' = (G' = (V', E'), s', t', C, \ell')$ where G' is a complete graph. Moreover, $\mathcal{H}'$ can be computed in polynomial time.*

Proof. Given a colored graph $\mathcal{H} = (G = (V, E), s, t, C, \ell)$ and $\alpha \in C$ such that $\{\alpha\} \in \mathcal{C}(\mathcal{H})$, we define a colored graph $\mathcal{H}' = (G' = (V', E'), s', t', C, \ell')$ where G' is a complete graph and show that $\mathcal{H}$ and $\mathcal{H}'$ are colored-cut-equivalent.

We construct $\mathcal{H}'$ by adding the edges $e \in \binom{V}{2} \setminus E$ to $\mathcal{H}$ and set $\ell'(e) := \alpha$ for all $e \in \binom{V}{2} \setminus E$. By construction, $E' = \binom{V}{2}$ and therefore G' is a complete graph. This can be done in polynomial time.

Now, we 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 $L_{\mathcal{H}} \in \mathcal{C}(\mathcal{H})$ such that $L_{\mathcal{H}} \subseteq L_{\mathcal{H}'}$ and vice versa.

Clearly, if P is an (s,t)-path in G, then P is also an (s,t)-path in G' and $\ell(E(P)) = \ell'(E(P))$. Hence, $\mathcal{C}(\mathcal{H}) \subseteq \mathcal{C}(\mathcal{H}')$ and therefore, for every $L_{\mathcal{H}} \in \mathcal{C}(\mathcal{H})$ there is $L_{\mathcal{H}'} \in \mathcal{C}(\mathcal{H}')$ such that $L_{\mathcal{H}'} \subseteq L_{\mathcal{H}}$.

It remains to show that for every $L_{\mathcal{H}'} \in \mathcal{C}(\mathcal{H}')$ there is $L_{\mathcal{H}} \in \mathcal{C}(\mathcal{H})$ such that $L_{\mathcal{H}} \subseteq L_{\mathcal{H}'}$. Let P' be an (s,t)-path in G'. If $\alpha \in \ell'(E(P'))$, then by assumption, $\ell'(E(P')) \supseteq \{\alpha\} \in \mathcal{C}(\mathcal{H})$. Thus, assume that $\alpha \notin \ell'(E(P'))$. Recall that $\ell'(e) = \alpha$ for all $e \in E' \setminus E$. Hence, $E(P') \subseteq E$ and therefore P' is also an (s,t)-path in G and $\ell(E(P')) = \ell'(E(P'))$. □

Proposition 6.10. *For all $i \geq 0$, $\mathrm{A(DA)}^i$-V and $\mathrm{(DA)}^{i+1}$-V are $\Sigma^{\mathrm{P}}_{2i+1}$-complete even on complete graphs.*

Proof. Since $\mathrm{A(DA)}^i$-V is a special case of $\mathrm{(DA)}^{i+1}$-V, we only have to show that $\mathrm{A(DA)}^i$-V is $\Sigma^{\mathrm{P}}_{2i+1}$-hard even on complete graphs. The problem $\mathrm{A(DA)}^0$-V (Colored (s,t)-Cut) is complete for NP $= \Sigma^{\mathrm{P}}_1$ on complete graphs [38]. Thus, it remains to show that, for $i \geq 1$, $\mathrm{A(DA)}^i$-V is $\Sigma^{\mathrm{P}}_{2i+1}$-hard even on complete graphs and that $\mathrm{(DA)}^{i+1}$-V is contained in $\Sigma^{\mathrm{P}}_{2i+1}$ on complete graphs. We reduce from $\mathrm{A(DA)}^i$-V where $d'_2 \geq 1$. Let $I' = (G' = (V', E'), s', t', C', \ell', (a'_1, \ldots, a'_{i+1}), (d'_2, \ldots, d'_{i+1}))$ be an instance of $\mathrm{A(DA)}^i$-V with $d'_2 \geq 1$.

We add a new color α, a new vertex v_α, and set $s := s', t := t', a_1 := a'_1 + 1$, and $d_j := d'_j$ and $a_j := a'_j$ for all $1 < j \leq i+1$. Furthermore, we set $V := V' \cup \{v_\alpha\}, E := E' \cup \{\{s, v_\alpha\}, \{v_\alpha, t\}\}, C := C' \cup \{\alpha\}$, and $\ell(\{s, v_\alpha\}) := \ell(\{v_\alpha, t\}) := \alpha$. Moreover, set $\mathcal{H} = (G, s, t, C, \ell)$.

Next, we show that $I = (\mathcal{H}, (a_1, \ldots, a_{i+1}), (d_2, \ldots, d_{i+1}))$ is a yes-instance if and only if I' is a yes-instance. By construction, $P = (s, v_\alpha, t)$ is an (s,t)-path in G and therefore, the attacker has to choose α in his first turn because $d_2 \geq 1$. Clearly, the attacker has a winning strategy for I if and only if he has a winning strategy for I'.

With Lemma 6.9 and the fact that $\{\alpha\} \in \mathcal{C}(I)$, we know that there is a colored graph $\tilde{\mathcal{H}} = (\tilde{G}, \tilde{s}, \tilde{t}, C, \tilde{\ell})$ which is colored-cut-equivalent to $\mathcal{H}$ and where $\tilde{G}$ is a complete graph. Hence, by Claim 4.12, I is a yes-instance if and only if $\tilde{I} := (\tilde{G}, \tilde{s}, \tilde{t}, C, \tilde{\ell}, (d_1, \ldots, d_i), (a_1, \ldots, a_i))$. Therefore, $\tilde{I}$ is a yes-instance if and only if I' is a yes-instance. Moreover, $\tilde{I}$ can be computed in polynomial time.

It remains to show that $(\mathrm{DA})^{i+1}$-V is contained in $\Sigma^{\mathrm{P}}_{2i+1}$. Let $I = (G = (V,E), s, t, C, \ell, (d_1, \ldots, d_{i+1}), \vec{a})$ be an instance of $(\mathrm{DA})^{i+1}$-V where G is a complete graph. If $d_1 \geq 1$, then the defender can win in his first turn by choosing $D_1 \ni \ell(\{s,t\})$ since $\{s,t\} \in E$. If $d_1 = 0$, then I is equivalent to the $\mathrm{A}(\mathrm{DA})^i$-V instance $(G, s, t, C, \ell, \vec{a}, (d_2, \ldots, d_{i+1}))$. Hence, $(\mathrm{DA})^{i+1}$-V is $\Sigma^{\mathrm{P}}_{2i+1}$-complete on complete graphs, since $\mathrm{A}(\mathrm{DA})^i$-V is $\Sigma^{\mathrm{P}}_{2i+1}$-complete due to Theorem 4.10.

□

A similar proof in combination with Theorem 4.17 and Claim 4.18 can be used to show the following corollary.

Corollary 6.11. Colored (s,t)-Cut Vulnerability Game *and* Colored (s,t)-Cut Robustness Game *are* PSPACE-*complete even on complete graphs.*

6.2 Restricted Colored Graphs

In this subsection, we analyze the complexity of DA-R on instances where every color appear in at most two (s,t)-paths or where every color is only given to one edge each. In these cases, Colored (s,t)-Cut is polynomial-time-solvable [10,22,38]. In contrast, we will show that DA-R is NP-complete on both of them. Hence, for any $i \geq 1$, $(\mathrm{DA})^i$-R and $\mathrm{A}(\mathrm{DA})^i$-R cannot be solved in polynomial time on these restricted colored graphs, unless P = NP.

Proposition 6.12. DA-R *is* NP-*hard even if every color appears in at most two (s,t)-paths.*

Proof. First, we show that DA-R is contained in NP if every color appears in at most two (s,t)-paths by describing a verifier. To this end, we show that if the defender has a winning strategy, then D_1, the set of colors he defender chooses, is a certificate. In other words, given an instance $I = (G = (V,E), s, t, C, \ell, d, a)$ of DA-R where every color appears in at most two (s,t)-paths and a set of colors $D_1 \subseteq S$ with $|D_1| \leq d$, we show that we can determine in polynomial time whether or not there is a colored (s,t)-cut A_1 of size at most a in G such that $D_1 \cap A_1 = \emptyset$. 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 $\ell(E(P)) \subseteq D_1$. Since the resulting graph might have parallel edges, we also subdivide every edge. In this graph, every color still appears on at most two (s,t)-paths [30]. Hence, D_1 is a winning strategy for the defender if the COLORED (s,t)-CUT instance (G', s, t, C, ℓ', a) is a no-instance. This can be determined in polynomial time since every color appears in at most two (s,t)-paths in G' [38]. This verifier is correct, since if the vertices $u, v \in V$ are connected with an (u,v)-path P in G with $\ell(E(P)) \subseteq D_1$, there is no colored (u,v)-cut $A_1 \subseteq C \setminus D_1$. Hence, DA-R is contained in NP if every color appears in at most two (s,t)-paths.

Second, we show that DA-R is NP-hard even if every color appears in at most two (s,t)-paths by giving a polynomial-time reduction from the NP-hard problem MATCHING INTERDICTION [41].

> 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 $r+1$ distinct edges with pairwise disjoint endpoints in $G - S$. Given an instance $I = (G = (V, E), b, r)$ of MATCHING INTERDICTION, we build in polynomial time an instance $I' = (G' = (V', E'), s, t, C, \ell, d, a)$ of DA-R where every color appears in at most two (s,t)-paths such that I is a yes-instance of MATCHING INTERDICTION if and only if I' is a yes-instance of DA-R. 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. Furthermore, we add for every $v \in V$ an (s,t)-path P_v in G' such that $\ell(E(P_v)) = \{\alpha_j^v \mid 0 \leq j \leq d\} \cup \{e \in E \mid v \in e\}$.

Note that for distinct vertices $v, w \in V$, $\ell(E(P_v)) \cap \ell(E(P_w)) = \{\{v, w\}\}$ if $\{v, w\} \in E$ and $\ell(E(P_v)) \cap \ell(E(P_w)) = \emptyset$ 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 wining 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 if and only if I' is a yes-instance.

($\Rightarrow$) 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. By construction, no color appears in more that two (s,t)-paths. Hence, for all distinct colors $\alpha, \beta \in R$ it holds that there is no (s,t)-path P' in G' with $\{\alpha, \beta\} \subseteq \ell(E(P'))$. 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 $e_1 := \{u_1, w_1\}, e_2 := \{u_2, w_2\} \in R$ in I (and therefore 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 set of colors 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 $\beta_v \in \ell(E(P_v)) \setminus D_1$. We will show that there is no matching of size $r+1$ in $G-(D_1 \cap E)$. 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 $\ell(E(P_v)) \cap M \neq \emptyset$, a contradiction. $\square$

Note that the proof of Proposition 6.12 is also a parameterized reduction from Matching Interdiction parameterized by b to DA-R parameterized by d. Since Matching Interdiction parameterized by b is W[1]-hard [24, 41], the next corollary follows directly.

Corollary 6.13. DA-R *parameterized by d is* W[1]-*hard even if every color appears in at most two (s,t)-paths.*

Note that Colored (s,t)-Cut can be solved in polynomial time on uncolored graphs, that is, when $|\ell^{-1}(\alpha)| = 1$ for all $\alpha \in C$ [17, 22]. In contrast, we will show that DA-R is NP-hard even on uncolored graphs.

Proposition 6.14. DA-R *is* NP-*complete even on uncolored graphs.*

To show this statement, we first show the NP-hardness for the following problem.

> Weighted Min (s,t)-Cut Interdiction
> **Input**: A graph $G = (V, E)$, two vertices $s, t \in V$, a cost function $c : E \to \mathbb{N}$, a weight function $\omega : E \to \mathbb{N}$, and integers d and a.
> **Question**: Is there a subset $S \subseteq E$ with $c(S) := \sum_{e \in S} c(s) \leq d$ such that for every (s,t)-cut $M \subseteq (E \setminus S)$ in G it holds that $\omega(M) := \sum_{e \in M} \omega(e) > a$?

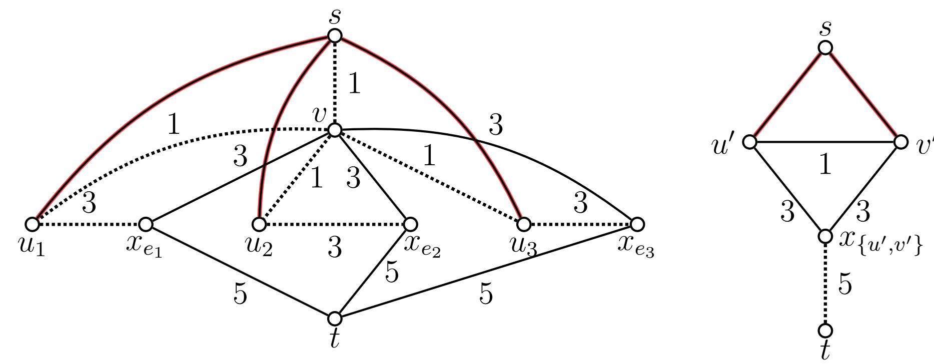
In other words, we ask if there is a set of edges $S \subseteq E$ such that there is no (s,t)-cut M in G that is disjoint to S and has weight at most a. Note that on uncolored graphs, DA-R is equivalent to Weighted Min (s,t)-Cut Interdiction where $c(e) = \omega(e) = 1$ for all $e \in E$.

Lemma 6.15. Weighted Min (s,t)-Cut Interdiction *is* NP-*complete even if* $c(e) + \omega(e) \in \mathcal{O}(|G|)$ *for all* $e \in E$.

Proof. First, we show that Weighted Min (s,t)-Cut Interdiction is contained in NP by describing a verifier. Let $I = (G = (V, E), s, t, c, \omega, d, a)$ be an instance of Weighted Min (s,t)-Cut Interdiction, we show that the set of edges $S \subseteq E$ with $|S| \leq d$ is a certificate if I is a yes-instance. We define the weight function $\omega' : E \to \mathbb{N}$ with $\omega'(e) := \omega(e)$ for all $e \in E \setminus S$ and $\omega'(e) := a + 1$ for all $e \in S$. Next, we answer yes if and only if the minimum weighted (s,t)-cut M in G with respect to ω' has weight at least $a+1$. By construction, $\omega'(M) > a$ for all $e \in S$ and therefore M has to be disjoint from S if $\omega(M) \leq a$. Hence, this algorithm is correct and runs in polynomial time [17,22]. Hence, Weighted Min (s,t)-Cut Interdiction is contained in NP.

Second, we show that Weighted Min (s,t)-Cut Interdiction is NP-hard. To this end, we reduce from the NP-hard problem Vertex Cover on cubic graphs [1,23].

Let $I = (G = (V, E), k)$ be an instance of Vertex Cover where $\deg(v) = 3$ for all $v \in V$. We describe how to construct an instance $I' = (G' = (V', E'), s, t, c, \omega, d, a)$ of Weighted Min (s,t)-Cut Interdiction in polynomial time such that I is a yes-instance of Vertex Cover if and only if I' is a yes-instance of Weighted Min (s,t)-Cut Interdiction. Recall that $n := |V|$ and $m := |E|$. Note that $m = \frac{3}{2}n$ since $\deg(v) = 3$ for all $v \in V$.



(a) (b)

Figure 5: These figures show the two subgraphs discussed in the first direction of the proof. The number on an edge denotes the weight of the corresponding edge and highlighted edges belong to S' and therefore cannot be contained in M. Moreover, dotted edges show the minimum-weight (s,t)-cut; (a) shows G'_v for $v \in \overline{S}$ with $N_G(v) = \{u_1, u_2, u_3\}$ and $e_i := \{v, u_i\}$ for $i \in \{1,2,3\}$; (b) shows $G'_{\{u',v'\}}$ for $\{u',v'\} \in E_2$.

We set $d := k$ and $a := 5m - 2(n-k) - 1$. The graph G' consists of a copy of G and two new vertices s and t. Moreover, we add edges $\{s,v\}$ to G' for all $v \in V$. Furthermore, for every edge $e := \{u,v\} \in E$ we add a vertex x_e and edges $\{u, x_e\}, \{v, x_e\}$, and $\{x_e, t\}$ to G'. For every edge $e \in E'$ we set $c(e) := 1$ if $s \in e$ and $c(e) := d+1$ otherwise. Finally, we set $\omega(\{u, x_{\{u,v\}}\}) := \omega(\{v, x_{\{u,v\}}\}) := 3$ for all $\{u,v\} \in E$. For the remaining edges $e \in E' \setminus E$, we set $\omega(e) := 5$ if $t \in e$, and $w(e) := 1$ otherwise. This completes the construction of I'.

Clearly, $c(e) + \omega(e) \in \mathcal{O}(|G'|)$ for all $e \in E'$. The idea is that a set $S' \subseteq E'$ with cost at most d has to be a subset of the edges incident with s. This is true since all other edges have cost $d+1$. Hence, S' can be seen as a choice of the vertices $S := \{v \mid \{s,v\} \in S'\}$ of G of size at most $k = d$ and vice versa.

Next, we show that I is a yes-instance of Vertex Cover if and only if I' is a yes-instance of Weighted Min (s,t)-Cut Interdiction.

($\Rightarrow$) Let $S \subseteq V$ be a vertex cover of size at most k in G. We show that $S' := \{\{s,v\} \mid v \in S\}$ is a solution with cost at most $d = k$ for I'. By construction, $c(S') := |S'| = |S| \leq k$. Hence, it remains to show that every (s,t)-cut $M \subseteq (E' \setminus S')$ in G' has weight at least $a + 1 = 5m - 2(n-k)$.

Let $\overline{S} := V \setminus S$.

To this end, we analyze the minimum-weight (s, t)-cut in constant-sized subgraphs of G'. These subgraphs will only share edges contained in S'. We will use this to argue that the minimum-weight (s, t)-cut M in G' with $M \cap S = \emptyset$ is at least the sum of all minimum-weight (s, t)-cuts of the different subgraphs. Let $G'_v := (V'_v, E'_v)$ where $V'_v := N_{G'}(v) \cup \{t, v\}$ and

$$E'_v := \{\{s, v\}, \{s, u\}, \{v, u\}, \{v, x_{\{v,u\}}\}, \{u, x_{\{v,u\}}\}, \{x_{\{v,u\}}, t\} \mid u \in N_G(v)\}$$

for all $v \in \overline{S}$. This graph is visualized in Figure 5(a). Since S is a vertex cover for G, it holds that $N_G(v) \subseteq S$ and therefore $\{s, u\} \in S'$ for all $u \in N_G(v)$. Clearly, every (s, t)-cut M_v in G'_v with $M_v \cap S' = \emptyset$ has weight at least 13. Note that G'_v is not the same graph as $G'[N_{G'}(v) \cup \{v, t\}]$ since there might be edges between the neighbors of v in G. Let $E_2 := \{e \in E \mid e \subseteq S\}$, and $E_1 := E \setminus E_2$. That is, E_2 contains the edges of E where both endpoints are contained in S and since S is a vertex cover for G, E_1 contains the edges where exactly one endpoint is contained in S. Moreover, for every $\{u, v\} \in E_2$ it holds that $\{s, u\}, \{s, v\} \in S'$. For each $e = \{u, v\} \in E_2$, let $G'_e := G'[\{s, u, v, x_e, t\}]$. This graph is visualized in Figure 5(b). Clearly, every (s, t)-cut M_e in G'_e with $M_e \cap S' = \emptyset$ has weight at least 5.

Since $\deg(v) = 3$ for all $v \in V$, it follows that $|E_1| = 3(n - k)$ and therefore $|E_2| = m - 3(n - k)$. Let $\mathcal{G} = \{G'_v \mid v \in \overline{S}\} \cup \{G'_e \mid e \in E_2\}$. By construction, for all pairwise distinct graphs $G'_1, G'_2 \in \mathcal{G}$ only share edges that are connected to s. Moreover, it holds that $E(G'_1) \cap E(G'_2) \subseteq S'$, since for every $v \in V \setminus S$, (s, v) is only contained in $E(G'_v)$. Hence, for every (s, t)-cut M in G' with $M \cap S' = \emptyset$ it holds that $\omega(M) \geq 5|E_2| + 13|\overline{S}| = 5m - 15(n - k) + 13(n - k) = 5m - 2(n - k) = a + 1$. Therefore, I' is a yes-instance of Weighted Min (s, t)-Cut Interdiction.

($\Leftarrow$) We show this direction by contraposition. Assume that I is a no-instance of Vertex Cover. We show that I' is a no-instance of Weighted Min (s, t)-Cut Interdiction. In the following, we define, for all $i \in \{1, 2, 3\}$, the sets of edged E_i where every edge $e \in E_i$ has exactly i endpoints in S. Let $S' \subseteq E$ with $c(S') \leq k$. Recall that $s \in e'$ for all $e' \in S'$ since $c(e) = d + 1$ for all $e \in E'$ with $s \notin e$. Let $S := \{v \mid \{s, v\} \in S'\}$ be the corresponding vertices of G. By assumption, S is not a vertex cover. Hence, the set $E_0 := \{e \in E \mid e \cap S = \emptyset\}$ is not empty. Furthermore, let $E_2 := \{e \in E \mid e \subseteq S\}$ and $E_1 := E \setminus (E_0 \cup E_2)$. Since $\deg(v) = 3$ for all $v \in V$, it follows that $|E_1| = \sum_{v \in V \setminus S}(3 - |N_G(v) \setminus S|) = 3(n - k) - 2|E_0|$

and therefore $|E_2| = m - (3(n-k) - 2|E_0|) - |E_0| = m - 3(n-k) + |E_0|$. Let $V'_s := \{s\} \cup S \cup \{x_e \mid e \subseteq S\}$ and $V'_t := V' \setminus V'_s = \{t\} \cup (V \setminus S) \cup \{x_e \mid e \nsubseteq S\}$. Clearly, $M := \{e' \in E' \mid e' \cap V'_s \neq \emptyset, e' \cap V'_t \neq \emptyset\}$ is an (s,t)-cut in G'. We will show that $\omega(M) \leq a$. By construction, $M = M_1 \cup M_2 \cup M_3$ where $M_1 := \{\{s,v\} \mid v \in V \setminus S\}$, $M_2 := \{\{u,v\}, \{u, x_{\{u,v\}}\} \mid \{u,v\} \in E, u \in S, v \in V \setminus S\}$, and $M_3 := \{\{x_e, t\} \mid e \subseteq S\}$. Clearly, $\omega(M_1) = n-k$, $\omega(M_2) = 4|E_1|$, and $\omega(M_3) = 5|E_2|$. Hence, $\omega(M) = n-k+4|E_1|+5|E_2| = n-k+4(3(n-k)-2|E_0|)+5(m-3(n-k)+|E_0|) = 5m-2(n-k)-3|E_0| < a$ since $E_0 \neq \emptyset$. Hence, I' is a no-instance of Weighted Min (s,t)-Cut Interdiction. □

Now, we prove Proposition 6.14. To this end, we show that Weighted Min (s,t)-Cut Interdiction is NP-hard even if $c(e) = \omega(e) = 1$ for all $e \in E$ which is equivalent to DA-R on uncolored graphs.

Proof of Proposition 6.14. Let $I = (G = (V,E), s, t, c, \omega, d, a)$ be an instance of Weighted Min (s,t)-Cut Interdiction where $c(e) + \omega(e) \in \mathcal{O}(|G|)$ for all $e \in E$. We describe how to construct an instance $I' = (G' = (V', E'), s, t, c', \omega', d, a)$ of Weighted Min (s,t)-Cut Interdiction in polynomial time such that $c'(e') = \omega'(e') = 1$ for all $e' \in E'$ and I is a yes-instance if and only if I' is a yes-instance.

The graph G' contains the vertices of V. For every edge $e = \{u,v\} \in E$ we add (u,v)-paths P^e_j for all $j, 1 \leq j \leq \omega(e)$, of length $c(e)$ such that every $x \in V(P^e_j) \setminus \{u,v\}$ only appears on P^e_j. In the special case of $c(e) = 1$ and $w(e) > 1$, we have to prevent parallel edges and therefore add an (u,v)-path P^e_1 of length one and an (u,v)-path P^e_j of length two for all $j, 2 \leq j \leq \omega(e)$.

This construction can be done in polynomial time since we add for every edge $e \in E$ at most $\mathcal{O}(c(e)\omega(e))$ new vertices and edges and by assumption, $c(e) + \omega(e) \in \mathcal{O}(|G|)$. It remains to show that I is a yes-instance if and only if I' is a yes-instance.

Clearly, for every edge $e = \{u,v\} \in E$ one can connect u and v with cost $c(e)$ in G if and only if one can connect u and v with cost $|E(P^e_1)| = c(e)$ in G'. Moreover, one has to take at least the weight $\omega(e)$ to cut e in G if and only if one has to take at least weight $\omega(e)$ to cut all the paths $P^e_j, 1 \leq j \leq \omega(e)$ in G'. Therefore, I is a yes-instance if and only if I' is a yes instance.

Hence, Weighted Min (s,t)-Cut Interdiction is NP-complete even if $c(e) = \omega(e) = 1$ and therefore DA-R on uncolored graphs is also NP-complete, since both problems are equivalent. □

Table 1: Classic Complexity of COLORED (s,t)-CUT, $(\mathrm{DA})^i$-V, $\mathrm{A}(\mathrm{DA})^i$-V, and CCVG in general and in some restricted cases.

	COLORED (s,t)-CUT	$(\mathrm{DA})^i$-V	$\mathrm{A}(\mathrm{DA})^i$-V	CCVG
In general	NP-c [10, 19]	Π^{P}_{2i}-c	$\Sigma^{\mathrm{P}}_{2i+1}$-c	PSPACE-c
$B_a(I) \leq 1$	$\in$ P	$\in$ P	$\in$ P	$\in$ P
$B_a(I) = 2$	$\in$ P	coNP-h coNP-c if $i = 1$	coNP-h	coNP-h
$\Delta(G) \leq 2$	$\in$ P	$\in$ P	$\in$ P	$\in$ P
$\Delta(G) = 3$	$\in$ P	Π^{P}_{2i}-c	Π^{P}_{2i}-h	PSPACE-c
bipartite planar	NP-c [39]	Π^{P}_{2i}-c	$\Sigma^{\mathrm{P}}_{2i+1}$-c	PSPACE-c
bipartite planar subcubic	$\in$ P	Π^{P}_{2i}-c	Π^{P}_{2i}-h	PSPACE-c
complete graphs	NP-c [38]	$\Sigma^{\mathrm{P}}_{2i-1}$-c	$\Sigma^{\mathrm{P}}_{2i+1}$-c	PSPACE-c
uncolored graphs	$\in$ P [17, 22]	coNP-h coNP-c if $i = 1$	coNP-h	coNP-h
every color in ≤ 2 (s,t)-paths	$\in$ P [38]	coNP-h coNP-c if $i = 1$	coNP-h	coNP-h

Even though, it is unlikely that DA-R can be solved in polynomial time in any of these two restricted cases, it is still interesting to observe that the in general Σ^{P}_2-complete problem becomes complete for Σ^{P}_1 =NP.

An overview on the classic complexity of the defined games can be seen in Table 1.

7 Conclusion

Finally, we summarize our results and list open question and potential future work. In this work, we extended previous work [10, 26, 35] and further analyzed Colored (s,t)-Cut from a parameterized point of view and generalized it to defender-attacker games.

7.1 Summary

In Section 3, we showed W[2]-hardness for Colored (s,t)-Cut parameterized by the maximum degree Δ and the edge deletion distance of the input graph to a maximum degree of three ξ_3. This extends the known W[2]-hardness for k [10] to an even larger parameter. In contrast, we gave FPT-algorithms for parameterizations by the feedback edge set number fes of the input graph or the edge deletion distance of the input graph to a maximum degree of two ξ_2. Both algorithms where obtained by bounding from above the number p of (s,t)-paths of the instances where fes or ξ_2 are bounded and using the known FPT-algorithm for Colored (s,t)-Cut when parameterized by p [27].

In Section 4.1, we generalized Colored (s,t)-Cut to the defender-attacker games $(\text{DA})^i$-R,$(\text{DA})^i$-V,A(DA)^i-R, and A(DA)^i-V where a defender and an attacker alternatingly choose color sets. The goal of the attacker is to complete a colored (s,t)-cut, whereas, the defender tries to prevent this. We showed that for all $i \geq 0$, A(DA)^i-V is Σ^{P}_{2i+1}-complete and A(DA)^i-R is Π^{P}_{2i+1}-complete and for all $i \geq 1$, $(\text{DA})^i$-V is Π^{P}_{2i}-complete and $(\text{DA})^i$-R is Σ^{P}_{2i}-complete. In other words, for an increasing but constant number of alternations between the agents, these games are complete for complexity classes of increasing levels of the polynomial-time hierarchy. For a non-constant number of alternations between the agents, we introduced, in Section 4.2, the games Colored (s,t)-Cut Vulnerability Game and Colored (s,t)-Cut Robustness Game and showed that both problems are PSPACE-complete even if every agent chooses only one color in each turn. In other words, the more turns an agent has in the game, the more complex it is to determine which player has a winning strategy.

In Section 5, we analyzed these games from a parameterized complexity point of view. We analyzed parameterizations by natural parameters related to budgets of the agents, and showed that they are unlikely to lead to FPT-algorithms. on the positive site, we showed that all games admit

Table 2: Parameterized Complexity of COLORED (s,t)-CUT, $(\mathrm{DA})^i$, $\mathrm{A(DA)}^i$, and CCRG. We write $\notin$ XP under the assumption that P $\neq$ NP. Furthermore, we write 'no poly kernel' under the assumption that NP $\not\subseteq$ coNP/poly. Recall that $B(I)$ denotes the sum of all budgets in I. Moreover, $\mathcal{B}_d(I)$ and $\mathcal{B}_a(I)$ denote the sets of budgets of the defender and attacker, respectively and $\mathcal{B}(I)$ is the union of both sets.

Parameter	COLORED (s,t)-CUT	$(\mathrm{DA})^i$-V, $\mathrm{A(DA)}^i$-V, CCVG
$B(I)$	$\in$ XP, W[2]-h	$\in$ XP, W[2]-h
$B(I) - b$, $b \in \mathcal{B}(I)$	$\notin$ XP	$\notin$ XP
$\|C\|$	$\in$ FPT, no poly kernel	$\in$ FPT, no poly kernel
$\|C\| - b$	$\in$ XP	$\in$ XP
$b \in \mathcal{B}_d(I)$	–	coW[1]-h
$b \in \mathcal{B}_a(I)$	W[1]-h	coW[2]-h
p	$\in$ FPT, no poly kernel	?
fes	$\in$ FPT, no poly kernel	?
ξ_2	$\in$ FPT, no poly kernel	?
ξ_3	$\in$ XP, W[2]-h	$\notin$ XP
Δ	$\in$ XP, W[2]-h	$\notin$ XP
$p + \|C\|$	$\mathcal{O}(p\|C\|)$ kernel in $\mathcal{O}(p(n + \|C\|) + m)$ time	$\mathcal{O}(p\|C\|)$ kernel in $\mathcal{O}(p(n + \|C\|) + m)$ time
$\kappa_r + \|C\|$, $r \in \mathbb{N}$ constant	$\mathcal{O}((\kappa_r + \|C\|)^r)$ kernel in $\mathcal{O}(\|I\|^r)$ time	$\mathcal{O}((\kappa_r + \|C\|)^r)$ kernel in $\mathcal{O}(\|I\|^r)$ time

FPT-algorithms when parameterized by the number of colors $|C|$. Furthermore, we showed in Section 5.4 how to achieve polynomial kernels for all colored cut games when parameterized by both the number of colors $|C|$ and the vertex cover number. We were also able to replace the vertex cover number with the vertex deletion distance κ_r to a maximum component size of r. This result is somewhat surprising, since even the PSPACE-complete games admit polynomial kernels and only few parameters are known for which COLORED (s,t)-CUT admits polynomial kernels. An overview on our parameterized complexity results is given in Table 2.

In Section 6, we analyzed the computational complexity of the colored cut games on restricted instances. First, in Section 6.1, we investigated instances with degree constraints. We showed that none of the colored cut games, except A(DA)^0-R and A(DA)^0-V, can be solved in polynomial time on bipartite planar subcubic or on complete graphs, unless P = NP. Second, in Section 6.2, we showed that (DA)^1-R is NP-complete on uncolored graphs and on colored graphs, where every color appears in at most two (s,t)-paths. This is a contrast to Colored (s,t)-Cut which can be solved in polynomial time in both cases. In one of these proofs we also introduced the Weighted Min (s,t)-Cut Interdiction problem and showed its NP-completeness. This problem is closely related to (DA)^1-R and might be interesting for the design and analysis of robust networks.

7.2 Future Work

One of the main open questions of this work is, if the colored cut games such as (DA)^i-R, or at least some of them, admit FPT-algorithms when parameterized by the number of (s,t)-paths p. If yes, then these games also admit FPT-algorithms when parameterized by the the feedback edge set number fes and the edge deletion distance to a maximum degree of two ξ_2. Another natural parameter which is related to p is the number of color sets $|\mathcal{C}(I)|$ of (s,t)-paths in the input graph. In every colored graph $\mathcal{H}$, $|\mathcal{C}(\mathcal{H})|$ cannot be larger than p and is probably much smaller in most instances. If one finds an exact FPT-algorithm to compute the set $\mathcal{C}(I)$, one can easily show an FPT-algorithm for Colored (s,t)-Cut when parameterized by $|\mathcal{C}(I)|$.

Furthermore, we were able to show that (DA)^1-R and (DA)^1-V can be solved in time $2^{|C|}|I|^{\mathcal{O}(1)}$. It would be interesting to analyze if algorithms with this running time can also be found for A(DA)^1-R or even for all colored cut games.

Since, the defined games are complete for complexity classes that are higher in the polynomial-time hierarchy than NP, one might also study whether a given parameterization allows not for an FPT-algorithm but for a parameterized reduction to a lower level of the polynomial-time hierarchy [13].

In general, it might be interesting to analyze other games or PSPACE-complete problems from a parameterized point of view and search for FPT-algorithms or even polynomial kernels.

Moreover, one can also define defender-attacker games where both agents, again, choose color sets each turn and the defender only wins if he completes a colored (s,t)-connector. It should be possible to show that, for a constant number of alternations between the agents, these games are also complete for complexity classes of different levels of the polynomial-time hierarchy. For a non-constant number of alternations, a special case of these games corresponds to UNIT-CCRG. Hence, we have already shown in Theorem 4.17 that a defender-attacker game where the defender aims to complete a colored (s, t)-connector with non-constant number of alternations between the agents is PSPACE-complete even if every agent only choses one color each turn.

All in all, it would be interesting to observe, if the positive results in this work are relevant for practice.

References

[1] Paola Alimonti and Viggo Kann. Hardness of approximating problems on cubic graphs. In *Proceedings of the Third Italian Conference on Algorithms and Complexity (CIAC '97)*, volume 1203 of *Lecture Notes in Computer Science*, pages 288–298. Springer, 1997.

[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] Etienne Birmelé, Rui A. Ferreira, Roberto Grossi, Andrea Marino, Nadia Pisanti, Romeo Rizzi, and Gustavo Sacomoto. Optimal listing of cycles and st-paths in undirected graphs. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '13)*, pages 1884–1896, 2013.

[5] 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.

[6] Hajo Broersma, Xueliang Li, Gerhard J. Woeginger, and Shenggui Zhang. Paths and cycles in colored graphs. *Australasian Journal of Combinatorics*, 31:299–312, 2005.

[7] 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.

[8] Ruay Shiung Chang and Shing-Jiuan Leu. The minimum labeling spanning trees. *Information Processing Letters*, 63(5):277–282, 1997.

[9] Stephen M. Chase. An implemented graph algorithm for winning shannon switching games. *Communications of the ACM*, 15(4):253–256, 1972.

[10] 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.

[11] 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.

[12] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015.

[13] Ronald de Haan and Stefan Szeider. Compendium of parameterized problems at higher levels of the polynomial hierarchy. *Electronic Colloquium on Computational Complexity*, 21:143, 2014.

[14] Michael Dom, Daniel Lokshtanov, and Saket Saurabh. Kernelization lower bounds through colors and ids. *ACM Transactions on Algorithms*, 11(2):13:1–13:20, 2014.

[15] Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.

[16] Pål Grønås Drange, Markus S. Dregi, and Pim van 't Hof. On the computational complexity of vertex integrity and component order connectivity. *Algorithmica*, 76(4):1181–1202, 2016.

[17] Jack Edmonds and Richard M. Karp. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the ACM*, 19(2):248–264, 1972.

[18] Shimon Even and Robert Endre Tarjan. A combinatorial problem which is complete in polynomial space. *Journal of the ACM*, 23(4):710–719, 1976.

[19] 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.

[20] Jörg Flum and Martin Grohe. *Parameterized Complexity Theory*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2006.

[21] Fedor V. Fomin, Dieter Kratsch, and Gerhard J. Woeginger. Exact (exponential) algorithms for the dominating set problem. In *Proceedings*

of the Thirtieth International Workshop on Graph-Theoretic Concepts in Computer Science (WG '04), volume 3353 of *Lecture Notes in Computer Science*, pages 245–256. Springer, 2004.

[22] Lester R. Ford and Delbert R. Fulkerson. Maximal flow through a network. In *Classic papers in combinatorics*, pages 243–248. Springer, 2009.

[23] M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.

[24] Jiong Guo and Yash Raj Shrestha. Parameterized complexity of edge interdiction problems. In *Proceedings of the Twentieth International Conference on Computing and Combinatorics (COCOON '14)*, pages 166–178, 2014.

[25] Refael Hassin, Jérôme Monnot, and Danny Segev. Approximation algorithms and hardness results for labeled connectivity problems. *Journal of Combinatorial Optimization*, 14(4):437–453, 2007.

[26] Somesh Jha, Oleg Sheyner, and Jeannette Wing. Two formal analyses of attack graphs. In *Proceedings of the Fifteenth IEEE Computer Security Foundations Workshop (CSFW '02)*, pages 49–63. IEEE, 2002.

[27] Sulamita Klein, Luerbio Faria, Ignasi Sau, Rubens Sucupira, and U Souza. On colored edge cuts in graphs. In *Sociedade Brasileira de Computaçao, Editor, Primeiro Encontro de Teoria da Computaçao—ETC*. CSBC, 2016.

[28] Mithilesh Kumar and Daniel Lokshtanov. A 2k kernel for -component order connectivity. *Computing Research Repository*, abs/1610.04711, 2016.

[29] Jérôme Monnot. The labeled perfect matching in bipartite graphs. *Information Processing Letters*, 96(3):81–88, 2005.

[30] Nils Morawietz, Niels Grüttemeier, Christian Komusiewicz, and Frank Sommer. Refined parameterizations for computing colored cuts in edge-colored graphs. In *SOFSEM 2020: Theory and Practice of Computer Science - 46th International Conference on Current Trends in Theory and Practice of Informatics, SOFSEM 2020, Limassol, Cyprus, January*

20-24, 2020, Proceedings, volume 12011 of *Lecture Notes in Computer Science*, pages 248–259. Springer, 2020.

[31] Rolf Niedermeier. *Invitation to Fixed-Parameter Algorithms.* Oxford University Press, 2006.

[32] Christos H. Papadimitriou. *Computational complexity.* Academic Internet Publ., 2007.

[33] Michal Pióro and Deepankar Medhi. *Routing, flow, and capacity design in communication and computer networks.* Morgan Kaufmann, 2004.

[34] Mauricio G. C. Resende and Panos M. Pardalos, editors. *Handbook of Optimization in Telecommunications.* Springer, 2006.

[35] Oleg Sheyner, Joshua W. Haines, Somesh Jha, Richard Lippmann, and Jeannette M. Wing. Automated generation and analysis of attack graphs. In *Proceedings of the Twenty-Third IEEE Symposium on Security and Privacy (S&P '02)*, pages 273–284. IEEE Computer Society, 2002.

[36] Larry J. Stockmeyer. The polynomial-time hierarchy. *Theoretical Computer Science*, 3(1):1–22, 1976.

[37] Larry J. Stockmeyer and Albert R. Meyer. Word problems requiring exponential time: Preliminary report. In *Proceedings of the Fifth Annual ACM Symposium on Theory of Computing (STOC '73)*, pages 1–9. ACM, 1973.

[38] Rubens André Sucupira. *Problemas de cortes de arestas maximos e mínimos em grafos.* PhD thesis, Universidade Federal do Rio de Janeiro, 2017.

[39] Yongge Wang and Yvo Desmedt. Edge-colored graphs with applications to homogeneous faults. *Computing Research Repository*, abs/1207.5439, 2012.

[40] 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.

[41] Rico Zenklusen. Matching interdiction. *Discrete Applied Mathematics*, 158(15):1676–1690, 2010.

[42] Peng Zhang and Bin Fu. The label cut problem with respect to path length and label frequency. *Theoretical Computer Science*, 648:72–83, 2016.