Balanced Judicious Bipartition is Fixed-Parameter Tractable

Daniel Lokshtanov∗, Saket Saurabh†, Roohani Sharma‡, Meirav Zehavi§

October 17, 2017

Abstract

The family of judicious partitioning problems, introduced by Bollobás and Scott to the field of extremal combinatorics, has been extensively studied from a structural point of view for over two decades. This rich realm of problems aims to counterbalance the objectives of classical partitioning problems such as Min Cut, Min Bisection and Max Cut. While these classical problems focus solely on the minimization/maximization of the number of edges crossing the cut, judicious (bi)partitioning problems ask the natural question of the minimization/maximization of the number of edges lying in the (two) sides of the cut. In particular, Judicious Bipartition (JB) seeks a bipartition that is “judicious” in the sense that neither side is burdened by too many edges, and Balanced JB also requires that the sizes of the sides themselves are “balanced” in the sense that neither of them is too large. Both of these problems were defined in the work by Bollobás and Scott, and have received notable scientific attention since then. In this paper, we shed light on the study of judicious partitioning problems from the viewpoint of algorithm design. Specifically, we prove that BJB is FPT (which also proves that JB is FPT).

1 Introduction

More than twenty years ago, Bollobás and Scott [3] defined the notion of judicious partitioning problems. Since then, the family of judicious partitioning problems has been extensively studied in the field of Extremal Combinatorics, as can be evidenced by the abundance of structural results described in surveys such as [7,36]. This rich realm of problems aims to counterbalance the objectives of classical partitioning problems such as Min Cut, Min Bisection, Max Cut and Max Bisection. While these classical problems focus solely on the minimization/maximization of the number of edges crossing the cut, judicious (bi)partitioning problems ask the natural questions of the minimization/maximization of the number of edges lying in the (two) sides of the cut. Another significant feature of judicious partitioning problems that also distinguishes them from other classical partitioning problems is that they inherently and naturally encompass several objectives, aiming to minimize (or maximize) the number of edges in several sets simultaneously.

In this paper, we shed light on properties of judicious partitioning problems from the viewpoint of the design of algorithms. Up until now, the study of such problems has essentially been overlooked at the algorithmic front, where one of the underlying reasons for this discrepancy might be that standard machinery does not seem to handle them effectively. Specifically, we focus on the Judicious Bipartition problem, where we seek a bipartition that is “judicious” in the sense that neither side is burdened by too many edges, and on the Balanced Judicious Bipartition problem, where we also require that the sizes of the sides themselves are “balanced” in the sense that neither of them is too large. Both of these problems were defined in the work by Bollobás and Scott, and have received notable scientific attention since then. Formally, Balanced Judicious Partition is defined as follows.

∗University of Bergen, Norway, daniello@ii.uib.no.
†Institute of Mathematical Sciences, HBNI, Chennai, India, UMI Relax and University of Bergen, Norway, saket@imsc.res.in.
‡Institute of Mathematical Sciences, HBNI, Chennai, India and UMI ReLax, roohani@imsc.res.in.
§Ben-Gurion University, Beersheba, Israel, meiravze@bgu.ac.il.
Judicious partitioning problems have also been studied intensively. Bollobás and Scott [5] proved an upper bound of $|V(G)| - δ - 1$ for a partition of $G$ that has a bipartite-cut of size at least $δ$. They complemented these results by showing an upper bound on the number of edges in each part as well as on edges between every pair of parts. Later, Bollobás and Scott [7] gave several new results, leaving open other new questions around judicious partitioning. In [8] they showed an optimal bound for judicious partitioning on bounded-degree graphs. These problems have also been studied on general hypergraphs [4], uniform hypergraphs [24], 3-uniform hypergraphs [6] and directed graphs [26].

The special cases of judicious partitioning problems called judicious bipartitioning and balanced judicious partitioning problems have also been studied intensively. Bollobás and Scott [5] proved an upper bound on judicious partitioning and proved that every graph that achieves the essentially best known lower bound on bipartite-cut, given by Edwards in [18] and [19], also achieves this upper bound for judicious bipartitioning. In fact, they showed that this is exact for complete graphs of odd order, which are the only extremal graphs without isolated vertices. Alon et al. [1] gave a non-trivial connection between the size of a bipartite-cut in a graph and judicious partitioning into two sets. In particular, they showed that if a graph has a bipartite-cut of size at least $\frac{m}{2} + δ$ where $δ ≤ m/30$, then there exists a partition $(V_1, V_2)$ of $V(G)$ such that $|E(G[V_i])| ≤ \frac{m}{4} - \frac{δ}{2} + \frac{18δ^2}{m} + 3\sqrt{m}$ for $i ∈ \{1, 2\}$. They complemented these results by showing an upper bound on the number of edges in each part when $δ > m/30$. Bollobás and Scott [9] studied similar relations between t-cuts and t-judicious partitionings for $t ≥ 3$. Recently, these results were further refined [39, 28]. Xu et al. [38] and Xu and Yu [40] studied balanced judicious bipartitioning where both parts are of almost equal size (that is, one of the sizes is $\lceil \frac{m}{2} \rceil$). Both of these papers concern the following conjecture of Bollobás and Scott [7]: if $G$ is a graph with minimum degree of at least 2, then $V(G)$ admits a balanced bipartition $(V_1, V_2)$ such that for each $i ∈ \{1, 2\}$, $|E(G[V_i])| ≤ \frac{m}{4}$. For further results on judicious partitioning, we refer to the surveys [7, 36].

Algorithmic Results. While classical partitioning problems such as MIN CUT, MIN BISECTION, MAX CUT and MAX BISECTION have been studied extensively algorithmically, the same is not true about judicious partitioning problems. Apart from MIN CUT, all the above mentioned partitioning problems are NP-complete. These NP-complete partitioning problems were investigated by all algorithmic paradigms meant for coping

---

**Balanced Judicious Bipartition (BJB)**

**Input:** A multi-graph $G$, and integers $µ$, $k_1$ and $k_2$

**Question:** Does there exist a partition $(V_1, V_2)$ of $V(G)$ such that $|V_1| = µ$ and for all $i ∈ \{1, 2\}$, it holds that $|E(G[V_i])| ≤ k_i$?

We note that in the literature, the term BJB refers to the case where $µ = \lceil \frac{|V(G)|}{2} \rceil$, and hence it is more restricted than the definition above. By dropping the requirement that $|V_1| = µ$, we get the JUDICIAL BIPARTITION (JB) problem. By using new crucial insights into these problems on top of the most advanced machinery in Parameterized Complexity to handle partitioning problems, we are able to resolve the question of the Parameterized Complexity of BJB (and hence also of JB). In particular, we prove the following theorem.

**Theorem 1.** BJB can be solved in time $2^{k \cdot O(1)} \cdot |V(G)|^{O(1)}$.

**Structural Results.** Denote $n = |V(G)|$ and $m = |E(G)|$. To survey several structural results about judicious partitioning problems, we first define the notions of $t$-cut and max (min) $t$-judicious partitioning. Given a partition of $V(G)$ into $t$ parts, a $t$-cut is the number of edges going across the parts, while a max (min) judicious $t$-partitioning is the maximum (minimum) number of edges in any of the parts. When $t = 2$, we use the standard terms bipartite-cut and judicious bipartitioning, respectively. Furthermore, by $t$-judicious partitioning we mean max $t$-judicious partitioning. As stated earlier, Bollobás and Scott [3] defined the notion of judicious partitioning problems in 1993. In that paper, they showed that for any positive integer $t$ and graph $G$, we can partition $V(G)$ into $t$ sets, $V_1, \ldots, V_t$, so that $|E(G[V_i])| ≤ \frac{t}{\mu} m$ for all $i ∈ \{1, \ldots, t\}$. Bollobás and Scott also studied this problem on graphs of maximum degree $Δ$, and showed that there exists a partition of $V(G)$ into $t$ sets $V_1, \ldots, V_t$ so that it simultaneously satisfies an upper bound and a lower bound on the number of edges in each part as well as on edges between every pair of parts. Later, Bollobás and Scott [7] gave several new results, leaving open other new questions around judicious partitioning. In [8] they showed an optimal bound for judicious partitioning on bounded-degree graphs. These problems have also been studied on general hypergraphs [4], uniform hypergraphs [24], 3-uniform hypergraphs [6] and directed graphs [26].
with NP-complete, including approximation algorithms and parameterized complexity. In what follows, we discuss known results related to these problems in the realm of parameterized complexity.

First, note that for every graph \( G \), there always exists a bipartition of the vertex set into two parts (in fact equal parts) such that at least \( m/2 \) edges go across. This immediately implies that Max Cut and Max Bisection are FPT when parameterized by the cut size (the number of edges going across the partition). This led Mahajan and Raman [29] to introduce the notion of above-guarantee parameterization. In particular, they showed that one can decide whether a graph has a bipartite-cut of size \( \frac{m}{2} + k \) in time \( O(m + n + k^4) \). However, Edwards [18] showed that every connected graph \( G \) has a bipartite-cut of size \( \frac{m}{2} + \frac{2k-1}{k} \). Thus, a more interesting question asks whether finding a bipartite-cut of size at least \( \frac{m}{2} + \frac{n-1}{k} + k \) is FPT. Crowston et al. [16] showed that indeed this is the case as they design an algorithm with running time \( O(8^k n^4) \). Recently, Etscheid and Mnich [21] discovered a kernel with a linear number of vertices (improving upon a kernel by Crowston et al. [15]), and the aforementioned algorithm was sped-up to run in time \( O(8^k m) \) [20]. Gutin and Yeo studied an above-guarantee version of Max Bisection [22], proving that finding a balanced bipartition such that it has at least \( \frac{m}{2} + k \) edges is FPT (also see [33]).

In this context Max Bisection, it is also relevant to mention the \((k, n-k)\)-Max Cut, which asks for a bipartite-cut of size at least \( p \) where one of the sides is of size exactly \( k \). Parameterized by \( k \), this problem is \( \text{W}[1]\)-hard [11], but parameterized by \( p \), this problem is solvable in time \( O^*(2^p) \) [35] (this result improved upon algorithms given in [10, 37]).

Until recently, the parameterized complexity of Min Bisection was open. Approaches to tackle this problem materialized when the parameterized complexity of \( \ell\)-Way Cut was resolved. Here, given a graph \( G \) and positive integers \( k \) and \( \ell \), the objective is to delete at most \( k \) edges from \( G \) such that it has at least \( \ell \) components. Kawarabayashi and Thorup [25] showed that this problem is FPT. Later, Chitnis et al. [13] developed a completely new tool based on this, called randomized contractions, to deal with plethora of cut problems. Other cut problems that have been shown to be FPT include the generalization of Min Cut to Multiway Cut and MultiCut [12, 31, 32]. Eventually, Cygan et al. [17], combining ideas underlying the algorithms developed for Multiway Cut, MultiCut, \( \ell\)-Way Cut and randomized contractions together with a new kind of decomposition, showed Min Bisection to be FPT. Finally, let us also mention the min \( c \)-judicious partitioning (which is a maximization problem), called \( c \)-Load Coloring, where given a graph \( G \) and a positive integer \( k \), the goal is to decide whether \( V(G) \) can be partitioned into \( c \) parts so that each part has at least \( k \) edges. Barbero et al. [2] showed that this problem is FPT (also see [21]).

Despite the abundance of work described above, the parameterized complexity of JB and BJB has not yet considered. We fill this gap in our study by showing that both of these problems are FPT. It is noteworthy to remark that one can show that the generalization of Min Bisection to \((c, \ell)\)-Min Bisection, where the objective is to find a partition into \( c \)-parts such that each part are almost equal and there are at most \( k \) edges going across different parts, is FPT [17]. However, such a generalization is not possible for either JB or BJB. Indeed, even the existence of an algorithm with running time \( n^{\ell(k)} \), for any arbitrary function \( f \), would imply a polynomial-time algorithm for 3-Coloring, where \( k \) is set to 0.

**Our Approach.** For the sake of readability, our strategy of presentation of our proof consists of the definition of a series of problems, each more “specialized” (in some sense) than the previous one, where each section shows that to eventually solve BJB, it is sufficient to focus on some such problem rather than the previous one. We start by showing that we can focus on the solution of the case of BJB where the input graph is bipartite at the cost of the addition of annotations. For this purpose, we present a (not complicated) Turing reduction that employs a known algorithm for the Odd Cycle Transversal problem (see Section 3). The usefulness of the ability to assume that the input graph is bipartite is a key insight in our approach. In particular, the technical parts of our proof crucially rely on the observation that a connected bipartite graph has only two bipartitions (here, we consider bipartitions as ordered pairs). Keeping this intuition in mind, our next step is to reduce the current annotated problem to one where the input graph is also assumed to be connected (this specific argument relies on a simple application of dynamic programming).

Having at hand an (annotated) problem where the input graph is assumed to be a connected bipartite graph, we proceed to the technical part of our proof, which employs the (heavy) machinery developed by Cygan.
et al.\cite{17}. While this machinery primarily aims to tackle problems where one seeks small cuts in addition to some size constraint, our problem involves a priori seemingly different type of constraints. Nevertheless, we observe that once we handle a connected graph, the removal of any set of $k$ edges (to deal with the size constraint and annotations) would not break the graph to more than $k+1$ connected components, and each of these components would clearly be a bipartite graph. Hence, we can view (in some sense) our problem as a cut problem. In practice, the relation between our problem and a cut problem is quite more intricate, and to realize our idea, we crucially rely on the fact that the connected components are bipartite graphs, which allows us to “guess” a binary vector specifying the bipartition of their vertex sets in the final solution. This operation entitles the employment of coloring functions (employing $k$ colors) and their translation into bipartitions (which at a certain point in our paper, we would start viewing as colorings employing two colors). Let us remark that the machinery introduced by [17] is the computation of a special type of tree decomposition. Accordingly, our approach would eventually involve the introduction of a specialization of BJB that aims to capture the work to perform when handling a bag of the tree decomposition. The definition of this specific problem is very technical, and hence we defer the description of related intuitive explanations to the appropriate locations in Section 5 where we have already set up the required notations to discuss it.

2 Preliminaries

General Notation. Let $f : A \to B$ be some function. Given $A' \subseteq A$, the notation $f(A') = b$ indicates that for all $a \in A'$, it holds that $f(a) = b$. An extension $f'$ of the function $f$ is a function whose domain $A'$ is a superset of $A$ and whose range is $B$, such that for all $a \in A$, it holds that $f'(a) = f(a)$. For any $A' \subseteq A$, the restriction $f|_{A'}$ of $f$ is a function from $A'$ to $B$ such that for any $a \in A'$, $f|_{A'}(a) = f(a)$. Bold face lowercase letters are used to denote tuples (vectors). For any tuple $v$, we let $v[i]$ denote the $i$th coordinate of $v$. Given some condition $\psi$, we define $[\psi] = 1$ if $\psi$ is true and $[\psi] = 0$ otherwise. For any positive integer $x$, we denote by $[x]$ the set $\{1, 2, \ldots, x\}$ and by $[x]_0$ the set $\{0, 1, \ldots, x\}$.

Graph Theory. Given a graph $G$, we let $V(G)$ and $E(G)$ denote the vertex-set and the edge-set of $G$, respectively. For a subset $A \subseteq V(G)$, we denote by $\delta(A)$ the set of boundary vertices of $A$, that is, $\delta(A) = \{v \in A : \text{there exists } u \in V(G) \setminus A \text{ such that } \{u, v\} \in E(G)\}$. We let $G \setminus A$ denote the subgraph of $G$ induced by $V(G) \setminus A$. A bipartite graph is a graph $G$ such that there exists a bipartition $(X, Y)$ of $V(G)$ where $X$ and $Y$ are independent sets. In this paper, we treat such bipartitions as ordered pairs. That is, if $(X, Y)$ is a bipartition of some bipartite graph $G$, then $(Y, X)$ is assumed to be a different bipartition of the graph $G$. For connected bipartite graphs, we have the following simple yet powerful insight.

Proposition 1 (Folklore). Any connected bipartite graph $G$ has exactly 2 bipartitions, $(X, Y)$ and $(Y, X)$.

The treewidth of a graph aims to measure how close the graph is to a tree. Formally, this notion is defined as follows.

Definition 1. A tree decomposition of a graph $G$ is a pair $(T, \beta)$ such that $T$ is a rooted tree, $\beta : V(T) \to 2^{V(G)}$, and the following conditions are satisfied.

1. For all $\{u, v\} \in E(G)$, there exists $t \in V(T)$ such that $u, v \in \beta(t)$.
2. For all $v \in V(G)$, the subgraph of $T$ induced by $X_v = \{t : v \in \beta(t)\}$ is a (connected) subtree of $T$ on at least one node.

Given $t, \hat{t} \in V(G)$, the notation $\hat{t} \preceq t$ indicates that $\hat{t}$ is a descendant of $t$ in $T$. Note that $t$ is a descendant of itself. For any $t \in V(T)$, let $t'$ denote the unique parent of $t$ in $T$. We also need the standard notations $\sigma(t) = \beta(t) \cap \beta(t')$ and $\gamma(t) = \bigcup_{\hat{t} \preceq t} \beta(\hat{t})$.

Proposition 2 (Folklore). Let $(T, \beta)$ be a tree decomposition of a graph $G$. Given a node $t \in V(T)$, let $t_1, \ldots, t_s$ denote the children of $t$ in $T$, and for all $i \in [s]$, define $V_{t_i} = \gamma(t_i) \setminus \beta(t)$. Let $V_t = V(G) \setminus \beta(t) \cup \bigcup_{i=1}^s V_{t_i}$. Then, the vertex-set of each connected component of $G \setminus \beta(t)$ is a subset of one of $V_{t_1}, \ldots, V_{t_s}, V_t$. 

4
Let $H$ be some hypergraph. A spanning forest of $H$ is a subset $E' \subseteq E(H)$ of minimum size such that the set containing all endpoints of the hyperedges in $E'$ is equal to $V(H)$. In this paper, we implicitly assume that hypergraphs contain no isolated vertices.

**Unbreakability.** A separation of a graph $G$ is a pair $(X, Y)$ such that $X, Y \subseteq V(G)$ and $X \cup Y = V(G)$. The order of a separation $(X, Y)$ is equal to $|X \cap Y|$.

**Definition 2.** Let $G$ be a graph, $A \subseteq V(G)$, and $q, k \in \mathbb{N}$. The set $A$ is said to be $(q,k)$-unbreakable in $G$ if for every separation $(X, Y)$ of $G$ of order at most $k$, either $|X \setminus Y) \cap A| \leq q$ or $|(Y \setminus X) \cap A| \leq q$.

We also define a notion of unbreakability in the context of functions.

**Definition 3.** A function $g : U \rightarrow [k]_0$ is called $(q,k)$-unbreakable if there exists $i \in [k]_0$ such that

$$\sum_{j \in [k]_0 \setminus \{i\}} |g^{-1}(j)| \leq q.$$ 

Let us now claim that there do no exist “too many” $(q,k)$-unbreakable functions.

**Lemma 1.** For all $q, k \in \mathbb{N}$, the number of $(q,k)$-unbreakable functions from a universe $U$ to $[k]_0$ is upper bounded by $\sum_{l=0}^{q} \binom{|U|}{l} \cdot q^k \cdot (k + 1)$.

**Proof.** Let $g : U \rightarrow [k]_0$ be some $(q,k)$-unbreakable function. By the definition of a $(q,k)$-unbreakable function, there exists $i \in [k]_0$ such that $\sum_{j \in [k]_0 \setminus i} |g^{-1}(j)| \leq q$. There are $(k + 1)$ ways of choosing such an index $i$, $\sum_{l=0}^{q} \binom{|U|}{l}$ ways of choosing at most $q$ elements that are not mapped to $i$, and at most $q^k$ ways of partitioning this set of at most $q$ elements into $k$ parts. Thus, the total number of such functions $g$ is upper bounded by $\sum_{l=0}^{q} \binom{|U|}{l} q^k (k + 1)$.

\[\square\]

### 3 Solving Balanced Judicious Bipartition

In this section, we prove Theorem 1 under the assumption that we are given an algorithm for an annotated, yet restricted, variant of BJB. Throughout this section, an instance of BJB is denoted by BJB($G, \mu, k_1, k_2$), and we define $k = k_1 + k_2$. Given a partition $(V_1, V_2)$ that witnesses that an instance BJB($G, \mu, k_1, k_2$) is a YES-instance, we think of the vertices in $V_1$ as colored 1 and the vertices in $V_2$ as colored 2; hence, we call such a partition a witnessing coloring of BJB($G, \mu, k_1, k_2$). To prove Theorem 1, we first define the **Odd Cycle Transversal** problem. Here, given a graph $G$, a set $S \subseteq V(G)$ is called an odd cycle transversal if $G \setminus S$ is a bipartite graph.

**Odd Cycle Transversal (OCT)**

**Input:** An undirected multi-graph graph $G$, and an integer $k$.

**Question:** Does $G$ have an odd cycle transversal of size at most $k$?

An instance of Odd Cycle Transversal is denoted by OCT($G, k$). The algorithm given by the result below shall be a central component in the design of our algorithm for BJB.

**Proposition 3 ([27]).** Odd Cycle Transversal can be solved in time $2.3146^k n^{O(1)}$.

Apart from OCT, we also need to define an auxiliary problem that we call **Annotated Bipartite-BJB** (AB-BJB). As we proceed with our proofs, we shall continue defining auxiliary problems, where each problem captures a task more specific and technically more challenging than the previous one. The choice of this structure aims to ease the readability of our paper. Intuitively, AB-BJB is basically the BJB problem on bipartite graphs, with an extra constraint that demands that certain vertices are assigned a particular color.
Theorem 2. Thus, denote \( l \) some bipartition \((|E|\) of minimum size such that every edge in \( E \).

Let \( S_1 \) be a set of vertices of \( G \) such that \( G \) is bipartite. Therefore, \( S_1 \) is an odd cycle transversal of \( G \) of size at most \( k \). Thus, \( \text{OCT}(G,k) \) is a YES-instance.

Henceforth, let \( S \) be an odd cycle transversal of \( G \) of size at most \( k \). Then, \( G \setminus S \) is a bipartite graph. Fix some bijection \((P,Q)\) of \( G \setminus S \). Let \( \mathcal{F} \) be the family of all subsets of \( S \), that is, \( \mathcal{F} = 2^S \). For any \( F \in \mathcal{F} \), denote \( l_1^F = |E(G[F])| \) and \( l_2^F = |E(G[S \setminus F])| \), and let \( G_F \) be the graph constructed as follows (see Fig. 1).

- \( V(G_F) = V(G \setminus S) \cup \{w_F, x_F, y_F, z_F\} \), where \( w_F, x_F, y_F, z_F \) are new distinct vertices.
- \( E(G_F) = E(G \setminus S) \cup E_{wF} \cup E_{xF} \cup E_{yF} \cup E_{zF} \), where the multisets \( E_{wF}, E_{xF}, E_{yF} \) and \( E_{zF} \) are defined as follows.
  - \( E_{wF} = \{e_u = (w_F, u) : u \in P \} \) and there exists \( v \in F \) such that \( (u,v) \in E(G) \},
  - \( E_{xF} = \{e_u = (x_F, u) : u \in Q \} \) and there exists \( v \in F \) such that \( (u,v) \in E(G) \},
  - \( E_{yF} = \{e_u = (y_F, u) : u \in Q \} \) and there exists \( v \in S \setminus F \) such that \( (u,v) \in E(G) \},
  - \( E_{zF} = \{e_u = (z_F, u) : u \in P \} \) and there exists \( v \in S \setminus F \) such that \( (u,v) \in E(G) \}.

Observe that \( G_F \) is a bipartite graph with \((P \cup \{x_F, y_F\}, Q \cup \{w_F, z_F\})\) as a bipartition.

**Annotated Bipartite-BJB (AB-BJB)**

**Parameter:** \( k_1 + k_2 \)

**Input:** A bipartite multi-graph \( G \) with bipartition \((P,Q)\), \( A, B \subseteq V(G) \) such that \( A \cap B = \emptyset \), and integers \( \mu, k_1 \) and \( k_2 \).

**Question:** Does there exist a partition \((V_1, V_2)\) of \( V(G) \) such that \( A \subseteq V_1 \), \( B \subseteq V_2 \), \( |V_1| = \mu \) and for \( i \in \{1, 2\}, |E(G[V_i])| \leq k_i? \)

An instance of AB-BJB is denoted by AB-BJB\((G, A, B, \mu, k_1, k_2)\). A partition \((V_1, V_2)\) satisfying the above properties is called a *witnessing coloring* of AB-BJB\((G, A, B, \mu, k_1, k_2)\). Furthermore, we need the following theorem, proven later in this paper.

**Theorem 2.** AB-BJB can be solved in time \( 2^{O(1)} \cdot \mu^{O(1)} \).

Let us now turn to focus on the proof of Theorem 1.

**Proof of Theorem 1.** Given an instance BJB\((G, \mu, k_1, k_2)\), call the algorithm given by Proposition 8 with the instance \( \text{OCT}(G,k) \) as input.

**Claim 1.** If \( \text{OCT}(G,k) \) is a NO-instance, then \( \text{BJB}(G, \mu, k_1, k_2) \) is a NO-instance.

**Proof.** Suppose BJB\((G, \mu, k_1, k_2)\) is a YES-instance. Let \((V_1, V_2)\) be a witnessing coloring for this instance. Let \( E' = E(G[V_1]) \cup E(G[V_2]) \). Then, observe that \( G \setminus E' \) is a bipartite graph. Let \( V' \) be a set of vertices of minimum size such that every edge in \( E' \) has at least one endpoint in \( V' \). Since \( |E'| \leq k \), it holds that \( |V'| \leq k \). Moreover, \( G \setminus V' \) is bipartite. Therefore, \( V' \) is an odd cycle transversal of \( G \) of size at most \( k \).

Figure 1: The construction in the proof of Theorem 1.
Claim 2. BJB$(G, \mu, k_1, k_2)$ is a YES-instance if and only if there exists $F \in \mathcal{F}$ such that AB-BJB$(G_F, \{w_F, x_F\}, \{y_F, z_F\}, \mu - |F| + 2, k_1 - l_1^F, k_2 - l_2^F)$ is a YES-instance.

Proof. In the forward direction, suppose that BJB$(G, \mu, k_1, k_2)$ is a YES-instance, and let $(V_1, V_2)$ be a witnessing coloring for BJB$(G, \mu, k_1, k_2)$. Moreover, let $F = V_1 \cap S$. Now, we define a partition $(V'_1, V'_2)$ of $V(G_F)$ as follows: $V'_1 = (V_1 \setminus S) \cup \{x_F, y_F\}$ and $V'_2 = (V_2 \setminus S) \cup \{w_F, z_F\}$. Let us now argue that $(V'_1, V'_2)$ is a witnessing coloring for AB-BJB$(G_F, \{w_F, x_F\}, \{y_F, z_F\}, \mu - |F| + 2, k_1 - l_1^F, k_2 - l_2^F)$.

First, by the construction of $(V'_1, V'_2)$, we have that $\{x_F, y_F\} \subseteq V'_1$ and $\{w_F, z_F\} \subseteq V'_2$. Second, as $V'_1 = (V_1 \setminus S) \cup \{x_F, y_F\}$, we also have that $|V'_1| = |V_1| - |F| + 2 = \mu + |F| + 2$. Third, observe that for any $i \in \{1, 2\}$, $|E(G[V'_i])| = |E(G[V_i])| - |E(G[F])|$. Thus, $|E(G[V'_1])| \leq k_1 - l_1^F$.

In the backward direction, suppose that there exists an $F \in \mathcal{F}$ such that AB-BJB$(G_F, \{w_F, x_F\}, \{y_F, z_F\}, \mu - |F| + 2, k_1 - l_1^F, k_2 - l_2^F)$ is a YES-instance, and let $(V'_1, V'_2)$ be a witnessing coloring for this instance. We now define a partition $(V_1, V_2)$ of $V(G)$ as follows: $V_1 = (V'_1 \cap V(G)) \cup F$ and $V_2 = (V'_2 \cap V(G)) \cup (S \setminus F)$. Let us now argue that $(V_1, V_2)$ is a witnessing coloring for BJB$(G, \mu, k_1, k_2)$. From the definition of $V_1$, and since $V(G) = (V(G_F) \setminus \{w_F, x_F, y_F, z_F\}) \cup F$ and $F \cap V(G_F) = \emptyset$, we have that $|V_1| = |V'_1| - |\{x_F, y_F\}| + |F| = \mu - |F| + 2 - 2 + |F| = \mu$. Moreover, observe that $|E(G[V'_1])| = |E(G[V_1])| + |E(G[F])| \leq k_1 + l_1^F$ and $|E(G[V'_2])| = |E(G[V_2])| + |E(G[S \setminus F])| \leq k_2 + l_2^F$. This concludes the proof of the claim. 

Thus, to solve an instance of BJB, it is enough to solve $2^{|S|} \leq 2^k$ instances of AB-BJB. Hence, by Theorem 2, BJB can be solved in time $2^{k^{O(1)} \cdot n^{O(1)}}$. 

4 Solving Annotated Bipartite-BJB

Recall the problem definition of Annotated Bipartite-BJB (AB-BJB) from Section 3. In this section, we prove Theorem 2. For this purpose, let us define another auxiliary problem, which we call Annotated Bipartite Connected-BJB (ABC-BJB). Intuitively, ABC-BJB is exactly the same problem as AB-BJB where we are interested in an answer for every choice of $\mu \in [n]_0$, $l_1 \in [k_1]_0$ and $l_2 \in [k_2]_0$, and additionally we demand the input graph to be connected.

**Annotated Bipartite Connected-BJB (ABC-BJB)**

**Parameter:** $k_1 + k_2$

**Input:** A connected bipartite multi-graph $G = (P, Q)$, $A, B \subseteq V(G)$ such that $A \cap B = \emptyset$, and integers $k_1$ and $k_2$.

**Output:** For all $\mu \in [n]_0$, $l_1 \in [k_1]_0$ and $l_2 \in [k_2]_0$, output a binary value, $\text{aJP}[^{\mu, l_1, l_2}]$, which is 1 if and only if there exists a partition $(V_1, V_2)$ of $V(G)$ such that

- $A \subseteq V_1$ and $B \subseteq V_2$,
- $|V_1| = \mu$, and
- for $i \in \{1, 2\}$, $|E(G[V_i])| \leq l_i$.

For any $\mu \in [n]_0$, $l_1 \in [k_1]_0$, $l_2 \in [k_2]_0$, a partition witnessing that $\text{aJP}[\mu, l_1, l_2] = 1$ is called a witnessing coloring for $\text{aJP}[\mu, l_1, l_2] = 1$. Moreover, an instance of ABC-BJB is denoted by ABC-BJB$(G, A, B, k_1, k_2)$. In the rest of this paper, we prove the following theorem.

**Theorem 3.** ABC-BJB can be solved in time $2^{k^{O(1)} \cdot n^{O(1)}}$.

Having Theorem 3 at hand, a simple application of the method of dynamic programming results in the proof of Theorem 2.

**Proof of Theorem 2.** Let AB-BJB$(G, A, B, \mu, k_1, k_2)$ be an instance of AB-BJB. Let $C_1, \ldots, C_r$ be the connected components of $G$. For all $i \in [r]$, let $A_i = A' \cap C_i$ and $B_i = B' \cap C_i$. Let $I_i = \text{ABC-BJB}(C_i, A_i, B_i, k_1, k_2)$. Let $\text{aJP}_i$ be the output table for the instance $I_i$, returned by the algorithm of Theorem 3. For any $j \in [r]$, let $G_j = G[\bigcup_{i \in [j]} C_i]$. Note that $G = G_r$. Let us define a 4-dimensional binary table $H$ in the following way. For all...
$i \in [r]$, $\mu' \in ||V(G)||_0$, $l_1 \in [k_1]_0$ and $l_2 \in [k_2]_0$. $M[i, \mu', l_1, l_2] = 1$ if and only if there exists a partition $(V_1, V_2)$ of $V(G)$ such that $(A \cap G_i) \subseteq V_1$, $(B \cap G_i) \subseteq V_2$, $|V_1| = \mu'$ and for $j \in \{1, 2\}$, $|E(G[V_j])| \leq l_j$. Observe that AB-BJB($A, B, \mu, k_1, k_2$) is a YES-instance if and only if $M[r, \mu, k_1, k_2] = 1$. We now compute $M[r, \mu, k_1, k_2]$ recursively using the following recurrences.

$$M[1, \mu', l_1, l_2] = aJP_1(\mu', l_1, l_2)$$

For all $i \in \{2, \ldots, r\}$, $\mu' \in ||V(G)||_0$, $l_1 \in [k_1]_0$ and $l_2 \in [k_2]_0$,

$$M[i, \mu', l_1, l_2] = \bigvee_{\mu'=\mu_1^1+\mu_2^2, l_1^1=l_1^1+l_2^2, l_2^2=l_2^1+l_2^2} (M[i - 1, \mu_1^1, l_1^1, l_2^2] \wedge aJP_i(\mu_2^2, l_2^1, l_2^2)),$$

where for all $j \in \{1, 2\}$, $\mu^j$, $l_1^j$ and $l_2^j$ are non-negative integers.

Note that the time taken to compute $M[r, \mu, k_1, k_2]$ is at most $(r \cdot n^2 \cdot k_1^2 \cdot k_2^2 \cdot \tau)$, where $\tau$ is the time taken to solve an instance of ABC-BJB. Since from Theorem 3 an instance of ABC-BJB can be solved in time $2^{k^{O(1)} \cdot n^{O(1)}}$ and $r \leq n$, AB-BJB can be solved in time $2^{k^{O(1)} \cdot n^{O(1)}}$.

\section{Solving Annotated Bipartite Connected-BJB}

Recall the problem definition of ABC-BJB from Section 5. In this section, we prove Theorem 3. Let us start by stating a known result that is a crucial component of our proof. By this result, we would have an instance of ABC-BJB by our algorithm.

Theorem 4 ([17]). There exists an $2^{O(k^2)} \cdot n^2 \cdot m$-time algorithm that, given a connected graph $G$ together with an integer $k$, computes a tree decomposition $(T, \beta)$ of $G$ with at most $n$ nodes such that the following conditions hold, where $\eta = 2^{O(k)}$.

1. For each $t \in V(T)$, the graph $G[\gamma(t) \setminus \sigma(t)]$ is connected and $\eta = n^2 \cdot n^{O(1)}$.

2. For each $t \in V(T)$, the set $\beta(t)$ is $(\eta, k)$-unbreakable in $G[\gamma(t)]$.

3. For each non-root $t \in V(T)$, we have that $|\sigma(t)| \leq \eta$ and $\sigma(t)$ is $(2k, k)$-unbreakable in $G[\gamma(parent(t))]$.

In order to process such a tree decomposition in a bottom-up fashion, relying on the method of dynamic programming, we need to address a specific problem associated with every bag, called Hypergraph Painting (HP). We chose the name HP to be consistent with the choice of problem name in [17], yet we stress that our problem is more general than the one in [17] (since the handling of a bag in our case is more intricate than the one in [17]).

Roughly speaking, an input of HP would consist of the following components. First, we are given “budget” parameters $k_1$ and $k_2$ as in an instance of ABC-BJB. Second, we are given an argument $b$ which would simply be $n$ (to upper bound $|\gamma(t)|$) when we construct an instance of HP while processing some node $t$ in the tree decomposition. Third, we are given a hypergraph $H$ which would essentially be the graph $G[\beta(t)]$ to which we add hyperedges that are supposed to represent the sets $\sigma(\tilde{t})$ for the children $\tilde{t}$ of $t$. Fourth, we are given an integer $q$ whose purpose is clarified in the discussion below the definition of HP (in Definition 5). Finally, for every hyperedge $F$, we are given a function $f_F : [k_1^1_0] \times [b] \times [k_1]_0 \times [k_2]_0 \to \{0, 1\}$. To roughly understand the meaning of this function, first recall that $F$ is supposed to represent $\sigma(\tilde{t})$ for some child $\tilde{t}$ of $t$. Now, the function $f_F$ aims to capture all information obtained while we processed the child $\tilde{t}$.
of \( t \) that might be relevant to the node \( t \). In particular, let us give an informal, intuitive interpretation of an element \((\Gamma, \mu, l_1, l_2)\) in the domain of \( f_F \). For this purpose, note that when we remove at most \( k \) edges from the (connected) graph \( G[\gamma(t)] \), we obtain at most \( k+1 \) connected components. The function \( \Gamma \) can be thought of as a method to assign to each vertex in \( \sigma(t) \) the connected component in which it should lie. Such information is extremely useful since each such connected component is in particular a bipartite graph, and hence by relying on Proposition and an exhaustive search, we would be able to use it to extract a witnessing coloring for an instance of ABC-BJB. The arguments \( \mu, l_1 \) and \( l_2 \) can be thought of as those in the definition of an output of ABC-BJB. Now, the value \( f_F(\Gamma, \mu, l_1, l_2) \) aims to indicate whether \( \Gamma, \mu, l_1 \) and \( l_2 \) are “realizable” in the context of the child \( \tilde{t} \) (the precise meaning of this value would become clearer later, once we establish additional necessary definitions.)

Let us now give the formal definition of HP. In this definition, we denote \( k = k_1 + k_2 \).

**Hypergraph Painting (HP)**

**Input:** Integers \( k_1, k_2, b, d \) and \( q \), a multi-hypergraph \( H \) with hyperedges of size at most \( d \), and for all \( F \in E(H) \), a function \( f_F : [k]_0^n \times [b]_0 \times [k_1]_0 \times [k_2]_0 \rightarrow \{0,1\} \).

**Output:** For all \( 0 \leq \mu \leq b \), \( 0 \leq l_1 \leq k_1 \), \( 0 \leq l_2 \leq k_2 \), output the binary value

\[
a_{\text{HP}}(\mu, l_1, l_2) = \bigvee_{\Upsilon: V(H) \rightarrow [k]_0} \bigwedge_{F \in E(H)} f_F(\Upsilon|_{F_1}, \mu, l_1, l_2),
\]

where \( \mu = \sum_{F \in E(H)} \mu^F \), \( \sum_{F \in E(H)} l_1^F \leq l_1 \), \( \sum_{F \in E(H)} l_2^F \leq l_2 \) and each of \( \mu^F \), \( l_1^F \) and \( l_2^F \) is a non-negative integer.

For a particular choice of \( \mu, l_1 \) and \( l_2 \), a function \( \Upsilon \) witnessing that \( a_{\text{HP}}(\mu, l_1, l_2) = 1 \) is called a witnessing coloring for \( a_{\text{HP}}(\mu, l_1, l_2) \). An instance of Hypergraph Painting is denoted by \( \text{HP}(k_1, k_2, b, d, q, H, \{f_F\}_{F \in E(H)}) \).

Although we are not able to tackle HP efficiently at its full generality, we are still able to solve those instances that are constructed when we would like to “handle” a single bag in a highly connected tree decomposition. For the sake of clarity, let us now address the beneficial properties that these instances satisfy individually, where each of them ultimately aims to ease our search for a witnessing coloring. The first property, called local unbreakability, unconditionally restricts the way a function \( \Gamma : F \rightarrow [k]_0 \) to be thought of as a restriction of the witnessing coloring we seek, can color a hyperedge \( F \) so that the value of \( f_F \) is 1.

**Definition 4 (Local Unbreakability).** An instance \( \text{HP}(k_1, k_2, b, d, q, H, \{f_F\}_{F \in E(H)}) \) is locally unbreakable if every \( F \in E(H) \) satisfies the following property: for any \( \Gamma : F \rightarrow [k]_0 \) that is not \((3k^2, k)\)-unbreakable, \( f_F(\Gamma, \mu, l_1, l_2) = 0 \) for all \( 0 \leq \mu \leq b \), \( 0 \leq l_1 \leq k_1 \) and \( 0 \leq l_2 \leq k_2 \).

The second property, called connectivity, implies that if we would like to use a function \( \Gamma : F \rightarrow [k]_0 \) to color a hyperedge (as a restriction of a witnessing coloring) with more than one color, then we would have to “pay” at least 1 from our budget \( l_1 + l_2 \).

**Definition 5 (Connectivity).** An instance \( \text{HP}(k_1, k_2, b, d, q, H, \{f_F\}_{F \in E(H)}) \) is connected if every \( F \in E(H) \) satisfies the following property: for any \( \Gamma : F \rightarrow [k]_0 \) for which there exist distinct \( i, j \in [k]_0 \) such that \( |\Gamma^{-1}(i)|, |\Gamma^{-1}(j)| > 0 \), it holds that \( f_F(\Gamma, \mu, l_1, l_2) = 1 \) only if \( l_1 + l_2 \geq 1 \).

The third property, called global unbreakability, directly restricts our “solution space” by implying that we only need to determine whether there exists a \((q, k)\)-unbreakable witnessing coloring.

**Definition 6 (Global Unbreakability).** An instance \( \text{HP}(k_1, k_2, b, d, q, H, \{f_F\}_{F \in E(H)}) \) is globally unbreakable if for all \( 0 \leq \mu \leq b \), \( 0 \leq l_1 \leq k_1 \), \( 0 \leq l_2 \leq k_2 \): if \( a_{\text{HP}}(\mu, l_1, l_2) = 1 \), then there exists a witnessing coloring \( \Upsilon : V(H) \rightarrow [k]_0 \) that is \((q, k)\)-unbreakable.
An instance $HP(k_1, k_2, b, d, q, H, \{f_F\}_{F \in \mathcal{E}(H)})$ is called a favorable instance of $HP$ if it is locally unbreakable, connected and globally unbreakable. For such instances we have the following theorem.

**Theorem 5.** $HP$ on favorable instances is solvable in time $2^{O(\min(k,q) \log(k+q))} d^{O(k^2)} m^{O(1)}$.

The proof of this theorem is very technical, involving non-trivial analysis of a very "messy" picture obtained by guessing part of a hypothetical witnessing coloring via the method of color coding. We defer the proof of Theorem 5 to Section 6.

From now onwards, to simplify the presentation of arguments ahead with respect to ABC-BJB, we would abuse notation and directly define a witnessing coloring as a function rather than a partition. More precisely, the term witnessing coloring for $aJP[\mu, l_1, l_2] = 1$ would refer to a function $col : V(G) \rightarrow \{V_1, V_2\}$ such that $A \subseteq V_1, B \subseteq V_2, -V_1 = \mu$ and for $i \in \{1, 2\}$, $|E(G[V_i])| \leq l_i$. To proceed to our proof of Theorem 5, we first need to introduce an additional notation. Roughly speaking, this notation translates a coloring $\Upsilon$ of the form that witnesses some $aJP[\mu, l_1, l_2] = 1$ to a coloring of the form that witnesses $aJP[\mu, l_1, l_2] = 1$ via some tuple $v \in \{0, 1\}^{k+1}$. Formally,

**Definition 7.** For a tuple $v \in \{0, 1\}^{k+1}$, bipartite graph $G$ with bipartition $(P, Q)$, $A \subseteq V(G)$ and $\Upsilon : A \rightarrow [k]_0$, define $\hat{\Upsilon}_v : A \rightarrow \{V_1, V_2\}$ as follows.

- For all $v \in P$, $\hat{\Upsilon}_v(v) = V_1$ if and only if $v[\Upsilon(v)] = 0$.
- For all $v \in Q$, $\hat{\Upsilon}_v(v) = V_1$ if and only if $v[\Upsilon(v)] = 1$.

Suppose we are given an instance ABC-BJB($G, A, B, k_1, k_2$). Fix some bipartition $(P, Q)$ of $G$. Let $(T, \beta)$ be the highly connected tree decomposition computed by the algorithm of Theorem 4 and let $r$ be the root of $T$. In what follows, $\eta = 2^{O(k)}$ as in Theorem 4 and $q = (\eta + k)k$. We now proceed to define a binary variable that is supposed to represent the answer we would like to compute when we process the bag of a specific node of the tree. Hence, one of the arguments is a node $t$, and three additional arguments are $\mu \in [n]_0, l_1 \in [k_1]_0$ and $l_2 \in [k_2]_0$. However, we cannot be satisfied with one answer, but need an answer for every possible “interaction” between the bag of $t$ and the bag of its parent $t'$. Thus, the definition also includes a coloring of $\sigma(t)$. The tuple $v \in \{0, 1\}^{k+1}$ is necessary for the translation process described in Definition 7 (the way in which we shall obtain such a “right” tuple later in the proof would essentially rely on brute-force).

**Definition 8.** Given $t \in V(T)$, a $(3k^2, k)$-unbreakable function $\Upsilon^\sigma : \sigma(t) \rightarrow [k]_0$, a tuple $v \in \{0, 1\}^{k+1}$, and integers $\mu \in [n]_0, l_1 \in [k_1]_0$ and $l_2 \in [k_2]_0$, the binary variable $y[t, \Upsilon^\sigma, v, \mu, l_1, l_2]$ is 1 if and only if there exists $\Upsilon : \gamma(t) \rightarrow [k]_0$ extending $\Upsilon^\sigma$ such that

1. The translation $\hat{\Upsilon}_v$ maps to $V_1$ exactly $\mu$ vertices, that is, $|\hat{\Upsilon}_v^{-1}(V_1)| = \mu$.
2. The translation $\hat{\Upsilon}_v$ maps $A \cap \gamma(t)$ to $V_1$ and $B \cap \gamma(t)$ to $V_2$, that is, $A \cap \gamma(t) \subseteq \hat{\Upsilon}_v^{-1}(V_1)$ and $B \cap \gamma(t) \subseteq \hat{\Upsilon}_v^{-1}(V_2)$.
3. For all $i \in \{1, 2\}$, it holds that $|E(G[\hat{\Upsilon}_v^{-1}(V_i)])| \leq l_i$.
4. The set of edges between vertices receiving different colors by $\Upsilon$ is exactly the set of edges between vertices that are mapped to the same side by the translation $\hat{\Upsilon}_v$, that is,

$$
\bigcup_{i,j \in [k]_0, i \neq j} E(\Upsilon^{-1}(i), \Upsilon^{-1}(j)) = E(G[\hat{\Upsilon}_v^{-1}(V_1)]) \cup E(G[\hat{\Upsilon}_v^{-1}(V_2)]).
$$

A function $\Upsilon$ as above is called a witnessing coloring for $y[t, \Upsilon^\sigma, v, \mu, l_1, l_2]$.

**Lemma 2.** For any $\mu \in [n]_0, l_1 \in [k_1]_0$ and $l_2 \in [k_2]_0$, $aJP[\mu, l_1, l_2] = 1$ if and only if there exists $v \in \{0, 1\}^{k+1}$ such that $y[r, \phi, v, \mu, l_1, l_2] = 1$. 

10
Proof. Let us prove the backward direction first. Let $v \in \{0, 1\}^{k+1}$ be such that $y[r, \emptyset, v, \mu, l_1, l_2] = 1$ and let $\Upsilon : V(G) \to [k]$ be one of its witnessing coloring. Then, Definition 3 directly implies that $\hat{\Upsilon}_v$ is a witnessing coloring for $aP(\mu, l_1, l_2) = 1$.

For the forward direction, let $col : V(G) \to \{V_1, V_2\}$ be a witnessing coloring for $aP(\mu, l_1, l_2)$. Let $X = E(G[\text{col}^{-1}(V_1)]) \cup E(G[\text{col}^{-1}(V_2)])$. Let $C_0, \ldots, C_r$ be the connected components of $G \setminus X$. Since $X \subseteq E(G)$ and $|X| \leq l_1 + l_2 \leq k_1 + k_2 = k$, we have that the number of connected components $r$ is upper bounded by $k$. For any $i \in [r]_0$, let $(P_i = (P \cap C_i), Q_i = (Q \cap C_i))$ be a bipartition of $C_i$ (recall that $G$ is a connected bipartite graph with fixed bipartition $(P, Q)$).

**Claim 3.** For any $i \in [r]_0$, either both $P_i \subseteq \text{col}^{-1}(V_1)$ and $Q_i \subseteq \text{col}^{-1}(V_2)$ or both $P_i \subseteq \text{col}^{-1}(V_2)$ and $Q_i \subseteq \text{col}^{-1}(V_1)$.

**Proof.** Consider a bipartition $(P'_i, Q'_i)$ of $C_i$, where $P'_i = \text{col}^{-1}(V_1)$ and $Q'_i = \text{col}^{-1}(V_2)$. Since $C_i$ is connected, from Proposition 4 either $P_i \subseteq P'_i$ and $Q_i \subseteq Q'_i$, or $P_i \subseteq Q'_i$ and $Q_i \subseteq P'_i$. Hence the claim follows. \(\Diamond\)

Let us now construct a $k$-length binary string, $v$, as follows. For any $i \in [r]_0$, $v[i] = 0$ if and only if $P_i \subseteq \text{col}^{-1}(V_1)$ and $Q_i \subseteq \text{col}^{-1}(V_2)$. For $i \in \{r + 1, \ldots, k\}$, $v[i] = 0$.

Define $\Upsilon : V(G) \to [k]_0$ as follows. For any $v \in V(G)$, $\Upsilon(v) = i$ if and only if $v \in C_i$.

**Claim 4.** $\hat{\Upsilon}_v = col$.

**Proof.** Consider some vertex $v \in V(G)$. Denote $V_j = \text{col}(v)$, $i = \Upsilon(v)$ and $b = v[i]$, and note that $j \in \{1, 2\}$, $i \in [k]_0$ and $b \in \{0, 1\}$. We divide the argument into two cases corresponding to whether $v \in P_i$ or $v \in Q_i$. Since $v \in \text{col}^{-1}(V_j)$, if $v \in P_i$, then by Claim 3, $P_i \subseteq \text{col}^{-1}(V_j)$ and $Q_i \subseteq \text{col}^{-1}(V_{3-j})$. Thus, by the construction of $v$, $b = j - 1$. Hence, by the definition of $\hat{\Upsilon}_v$, $\hat{\Upsilon}_v(v) = v_j$. Similarly, if $v \in Q_i$, then by Claim 3, $Q_i \subseteq \text{col}^{-1}(V_j)$ and $P_i \subseteq \text{col}^{-1}(V_{3-j})$. Thus, by the construction of $v$, $b = 2 - j$. Hence, by the definition of $\hat{\Upsilon}_v$, $\hat{\Upsilon}_v(v) = v_j$.

Since the choice of $v$ was arbitrary, by the definition of $\hat{\Upsilon}_v$, we have that $\hat{\Upsilon}_v(v) = v_j$. \(\Diamond\)

**Claim 5.** For the binary string $v$ constructed as above, the function $\Upsilon$ constructed above is a witnessing coloring for $y[r, \emptyset, v, \mu, l_1, l_2] = 1$.

**Proof.** Since $\hat{\Upsilon}_v = col$, from the definition of $col$, we have that $|\hat{\Upsilon}_v^{-1}(V_1)| = \mu$, $A \subseteq \hat{\Upsilon}_v^{-1}(V_1)$, $B \subseteq \hat{\Upsilon}_v^{-1}(V_2)$, and for all $i \in \{1, 2\}$, $|E(G[\hat{\Upsilon}_v^{-1}(V_i)])| \leq l_i$. Observe that $\bigcup_{i,j \in [k]_0, i \neq j} E(\Upsilon_i^{-1}(i), \Upsilon_i^{-1}(j)) = X$. Therefore, $\bigcup_{i \in [k]_0, i \neq j} E(\Upsilon_i^{-1}(i), \Upsilon_i^{-1}(j)) = E(G[\hat{\Upsilon}_v^{-1}(V_1)]) \cup E(G[\hat{\Upsilon}_v^{-1}(V_2)])$. Thus, $\Upsilon$ is a witnessing coloring for $y[r, \emptyset, v, \mu, l_1, l_2] = 1$. \(\Diamond\)

This concludes the proof of the lemma. \(\square\)

By Lemma 2 it is sufficient to compute $y[t, \Upsilon', v, \mu, l_1, l_2]$ for all $\mu \in [n], l_1 \in [k]_0$ and $l_2 \in [k]_0$. To this end, we need to compute $y[t, \Upsilon', v, \mu, l_1, l_2]$ for every node $t \in V(T)$, function $\Upsilon' : \sigma(t) \to [k]$ that is $(3k^2, k)$-unbreakable, tuple $v \in \{0, 1\}^{k+1}$, and integers $\mu \in [n], l_1 \in [k]_0$ and $l_2 \in [k]_0$. Here, we employ bottom-up dynamic programming over the tree decomposition $(T, \beta)$. Let us now zoom into the computation of $y[t, \Upsilon', v, \mu, l_1, l_2]$ for all $\mu \in [n], l_1 \in [k]_0$ and $l_2 \in [k]_0$, for some specific $t, \Upsilon'$ and $v$. Note that we now assume that values corresponding to the children of $t$ (if such children exist) have been already computed correctly. Moreover, note that $|\sigma(t)| \leq \eta$, the number of $(3k^2, k)$-unbreakable functions $\Upsilon' : \sigma(t) \to [k]$ is at most $\eta^{O(k)} = 2k^{O(k)}$ (by Lemma 3), and the number of binary vectors of size $k + 1$ is at most $2^{k+1}$. Thus, the total running time would consist of the computation time of $(T, \beta)$, and $n \cdot \eta^{O(k)} \cdot 2^{k+1}$ times the computation time for a set of values as the one we examine now. Hence, it remains to show how to compute the current set of values in time $2^{k^{O(1)}}$.

To compute our current set of values, let us construct an instance $HP(k_1, k_2, n, \eta, q, H, \{f_F\}_{F \in E(H)})$ of HP where $V(H) = \beta(t)$, and $E(H)$ and $\{f_F\}_{F \in E(H)}$ are defined as follows.
1. **Type-1 Hyperedges.** For all \( v \in \beta(t) \), insert \( F = \{v\} \) into \( E(H) \). Define \( f_F : [k_0]^2 \times [n_0] \times [k_1]_0 \times [k_2]_0 \to \{0, 1\} \) as

\[
f_F(\Gamma, \mu, l_1, l_2) = \begin{cases} 
0, & \text{if } v \in \sigma(t) \text{ and } \Gamma(v) \neq \Upsilon^\sigma(v) \\
1, & \text{if } v \in A, \hat{\Gamma}_v(F) = V_1, l_1 = l_2 = 0 \text{ and } \mu = 1 \\
1, & \text{if } v \in B, \hat{\Gamma}_v(F) = V_2, l_1 = l_2 = 0 \text{ and } \mu = 0 \\
1, & \text{if } v \not\in A \cup B, l_1 = l_2 = 0 \text{ and } \mu = |\hat{\Gamma}_v(F) = V_1| \\
0, & \text{otherwise}
\end{cases}
\]

2. **Type-2 Hyperedges.** For all \( (u, v) \in E(G[\beta(t)]) \), add \( F = \{u, v\} \) in \( E(H) \). Define \( f_F : [k_0]^2 \times [n_0] \times [k_1]_0 \times [k_2]_0 \to \{0, 1\} \) as

\[
f_F(\Gamma, \mu, l_1, l_2) = \begin{cases} 
0, & \text{if } \mu \neq 0 \\
1, & \text{if } \hat{\Gamma}_v(u) \neq \hat{\Gamma}_v(v) \text{ and } \Gamma(u) = \Gamma(v) \\
1, & \text{if } \hat{\Gamma}_v(u) = \hat{\Gamma}_v(v) = V_1 \text{ and } l_1 \geq 1 \\
1, & \text{if } \hat{\Gamma}_v(u) = \hat{\Gamma}_v(v) = V_2 \text{ and } l_2 \geq 1 \\
0, & \text{otherwise}
\end{cases}
\]

3. **Type-3 Hyperedges.** For all \( i \in V(T) \) that is a child of \( t \) in the tree \( T \), insert \( F = \sigma(i) \) into \( E(H) \). Define \( f_F : [k_0]^2 \times [n_0] \times [k_1]_0 \times [k_2]_0 \to \{0, 1\} \) as

\[
f_F(\Gamma, \mu, l_1, l_2) = \begin{cases} 
0, & \text{if } \Gamma \text{ is not } (3k^2, k)\text{-unbreakable or } y[\hat{\Gamma}, \Gamma, \mu, l_1, l_1', l_2' + l_2] = 0 \\
1, & \text{otherwise}
\end{cases}
\]

where \( \mu' = |\hat{\Gamma}_v^{-1}(V_1)| \) and \( l_i' = |\{(u, v) \in E(G[\sigma(i)]) : \hat{\Gamma}_v(u) = \hat{\Gamma}_v(v) = V_i\}| \) for \( i \in [2] \).

Let us first claim that witnessing colorings related to \( \text{HP}(k_1, k_2, n, \eta, q, H, \{f_F\}_{F \in E(H)}) \) are useful in the sense that they can be extended to witnessing colorings for the binary values in which we are interested.

**Lemma 3.** For all \( \mu \in [n], l_1 \in [k_1]_0, l_2 \in [k_2]_0, \text{ if } \text{aHP}[\mu, l_1, l_2] = 1, \text{ then } y[t, \Upsilon^\sigma, v, \mu, l_1, l_2] = 1. \) In fact, for any witness \( \Upsilon : \beta(t) \to [k_0] \) of \( \text{aHP}[\mu, l_1, l_2] = 1 \), there exists a function \( \Upsilon' : \gamma(t) \to [k_0] \) that extends \( \Upsilon \) and witnesses \( y[t, \Upsilon^\sigma, v, \mu, l_1, l_2] = 1. \)

**Proof.** If \( \text{aHP}[\mu, l_1, l_2] = 1, \text{ let } \Upsilon : \beta(t) \to [k_0] \text{ be a witnessing coloring for } \text{aHP}[\mu, l_1, l_2] = 1. \) Then, there exist \( \mu = \sum_{F \in E(H)} \mu^F \), \( \sum_{F \in E(H)} l_1^F \leq l_1 \) and \( \sum_{F \in E(H)} l_2^F \leq l_2 \), such that for all \( F \in E(H), f_F(\Upsilon|_F, \mu^F, l_1^F, l_2^F) = 1. \) In particular, the following conditions hold.

1. Since for any type-1 hyperedge \( F \), it holds that \( f_F(\Upsilon|_F, \mu^F, l_1^F, l_2^F) = 1 \), we overall have that \( \Upsilon^\sigma \subseteq \Upsilon, A \cap \beta(t) \subseteq \hat{\Upsilon}_v^{-1}(V_1), B \cap \beta(t) \subseteq \hat{\Upsilon}_v^{-1}(V_2) \) and

\[
\sum_{F \text{ is a type-1 hyperedge}} \mu^F = |\hat{\Upsilon}_v^{-1}(V_1) \cap \beta(t)|.
\]

2. Since for any type-2 hyperedge \( F \) and \( i \in \{1, 2\} \), it holds that \( f_F(\Upsilon|_F, \mu^F, l_1^F, l_2^F) = 1 \), we overall have that

\[
|E(G[\hat{\Upsilon}_v^{-1}(V_i) \cap \beta(t)])| \leq \sum_{F \text{ is a type-2 hyperedge}} l_i^F.
\]

3. For any type-3 hyperedge \( F = \sigma(t_i) \), since \( f_F(\Upsilon|_F, \mu^F, l_1^F, l_2^F) = 1 \), we have that \( \Upsilon|_F \) is \((3k^2, k)\)-unbreakable and \( y[t_i, \Upsilon|_F, \mu' + \mu, l_1^F + l_1', l_2^F + l_2'] = 1 \), where \( \mu' = |\hat{\Upsilon}_v^{-1}(V_1) \cap F|, l_1' = |\{(u, v) \in E(G[\sigma(t_i)]) : \hat{\Upsilon}_v(u) = \hat{\Upsilon}_v(v) = V_1\}| \) and \( l_2' = |\{(u, v) \in E(G[\sigma(t_i)]) : \hat{\Upsilon}_v(u) = \hat{\Upsilon}_v(v) = V_2\}| \). We thus derive that there exists a witnessing coloring \( \Upsilon' : \gamma(t_i) \to [k_0] \) for the condition \( y[t_i, \Upsilon|_F, \mu' + \mu, l_1^F + l_1', l_2^F + l_2'] = 1 \). Specifically, the following conditions are satisfied.
of the other conditions specified in Definition 8 are satisfied. The aforementioned Claim 6.

For the sake of clarity, let us extract the required argument to the proof of a separate claim.

Keeping the above items in mind, we proceed to identify a witnessing coloring for \( y[t, \Upsilon^*] \). We construct such a coloring \( \Upsilon' : \gamma(t) \to [k]_0 \) as follows. For all \( v \in \gamma(t) \), if \( v \in \beta(t) \), then define \( \Upsilon'(v) = \Upsilon(v) \), and otherwise there exists a unique child \( t_i \) of \( t \) such that \( v \in \gamma(t_i) \), in which case we define \( \Upsilon'(v) = \Upsilon(t) \). For the sake of clarity, let us extract the required argument to the proof of a separate claim.

Claim 6. The aforementioned \( \Upsilon' \) is a witnessing coloring for \( y[t, \Upsilon^*, \nu, \mu, l_1, l_2] = 1 \).

Proof. First, note that by Item 1, we have that \( \Upsilon \subseteq \Upsilon \) and therefore \( \Upsilon \subseteq \Upsilon ' \). Let us now verify that all of the other conditions specified in Definition 5 are satisfied.

- Let us first prove Condition 1. To this end, we observe that by Items 1, 3a and 3b, we have that the three following equalities hold:
  
  \[
  |\Upsilon^{-1}_v(V_1)| = |\Upsilon^{-1}_v(V_1) \cap \beta(t)| + \sum_{t_i \text{ is a child of } t \in T} |\Upsilon^{-1}_v(V_1) \cap (\gamma(t_i) \setminus \sigma(t_i))|.
  \]

  - For every child \( t_i \) of \( t \), \( |\Upsilon^{-1}_v(V_1) \cap (\gamma(t_i) \setminus \sigma(t_i))| = \mu_F \), where \( F = \sigma(t_i) \).

  Thus, since \( \sum_{F \text{ is a type-2 hyperedge}} \mu_F = 0 \), we conclude that \( |\Upsilon^{-1}_v(V_1)| = \sum_{F \in E(H)} \mu_F = \mu \).

- Next, we prove Condition 2. However, by Items 1 and 3c, we directly deduce that both \( A \cap \gamma(t) \subseteq \Upsilon^{-1}_v(V_1) \) and \( B \cap \gamma(t) \subseteq \Upsilon^{-1}_v(V_2) \) as required.

- We now turn to prove Condition 3. In light of Item 3a, note that

  \[
  |E(G[\Upsilon^{-1}_v(V_1)])| = |E(G[\Upsilon^{-1}_v(V_1) \cap \beta(t)])| + \sum_{t_i \text{ is a child of } t \in T} |E(G[\Upsilon^{-1}_v(V_1) \cap (\gamma(t_i) \setminus \sigma(t_i))])|
  \]

  Now, observe that by Items 2, 3a and 3d, the two following equations hold.

  - For every child \( t_i \) of \( t \), \( |E(G[\Upsilon^{-1}_v(V_1) \cap (\gamma(t_i) \setminus \sigma(t_i))])| = l_i^F + |E(G[\Upsilon^{-1}_v(V_1) \cap \sigma(t_i)])| \), where \( F = \sigma(t_i) \).

  Since \( \sum_{F \text{ is a type-1 hyperedge}} l_i^F = 0 \), we conclude that

  \[
  |E(G[\Upsilon^{-1}_v(V_1)])| \leq \sum_{F \in E(H)} l_i^F \leq l_1.
  \]

  Similarly, we derive that \( |E(G[\Upsilon^{-1}_v(V_2)])| \leq \sum_{F \in E(H)} l_i^F \leq l_2 \).
Finally, we prove Condition 4. In the first direction, consider some edge \( e \in E(G[\hat{\Upsilon}^{-1}(V_1)]) \cup E(G[\hat{\Upsilon}^{-1}(V_2)]) \). Let us denote \( e = \{u, v\} \), and observe that \( \hat{\Upsilon}'(v) = \hat{\Upsilon}'(u) \). If \( u, v \in \gamma(t_i) \) for some child \( t_i \) of \( t \), then by Item 3c we have that \( e \in \bigcup_{i,j \in [k]_0, i \neq j} E(\Upsilon^{-1}(i), \Upsilon^{-1}(j)) \). Otherwise, \( u, v \in \beta(t) \), and thus \( e \) is some type-2 hyperedge \( F \). Since \( f_F(\Upsilon | F, \mu^F, t_1^F, t_2^F) = 1 \), the definition of \( f_F(\Upsilon | F, \mu^F, t_1^F, t_2^F) \) directly implies that \( \Upsilon(v) \neq \Upsilon(v) \), and therefore again \( e \in \bigcup_{i,j \in [k]_0, i \neq j} E(\Upsilon^{-1}(i), \Upsilon^{-1}(j)) \).

In the other direction, consider some edge \( e \in \bigcup_{i,j \in [k]_0, i \neq j} E(\Upsilon^{-1}(i), \Upsilon^{-1}(j)) \). Let us denote \( e = \{u, v\} \), and observe that \( \Upsilon'(v) \neq \Upsilon'(u) \). If \( u, v \in \gamma(t_i) \) for some child \( t_i \) of \( t \), then by Item 3c we have that \( e \in E(G[\hat{\Upsilon}^{-1}(V_1)]) \cup E(G[\hat{\Upsilon}^{-1}(V_2)]) \). Otherwise, \( u, v \in \beta(t) \), and thus \( e \) is some type-2 hyperedge \( F \). Since \( f_F(\Upsilon | F, \mu^F, t_1^F, t_2^F) = 1 \), the definition of \( f_F(\Upsilon | F, \mu^F, t_1^F, t_2^F) \) directly implies that \( \Upsilon'(v) = \Upsilon'(u) \), and therefore again \( e \in E(G[\hat{\Upsilon}^{-1}(V_1)]) \cup E(G[\hat{\Upsilon}^{-1}(V_2)]) \).

Thus, we have proved that \( \Upsilon' \) is a witnessing coloring for \( y[t, \Upsilon^*, v, \mu, l_1, l_2] \). Moreover, \( \Upsilon' \), which extends \( \Upsilon \), is the desired function for the second part of the lemma.

This concludes the proof of the lemma.

In light of Lemma 3, we now turn to verify that \( \text{HP}(k_1, k_2, n, \eta, q, H, \{f_F\}_{F \in E(H)}) \) is of the form that we are actually able to solve.

**Lemma 4.** \( \text{HP}(k_1, k_2, n, \eta, q, H, \{f_F\}_{F \in E(H)}) \) is a favorable instance of \( \text{HP} \).

**Proof.** Let us verify that each of the three properties of a favorable instance is satisfied.

- **Local Unbreakability:** Let us choose an arbitrary \( F \in E(H) \). If \( F \) is a type-1 or a type-2 hyperedge, then since \( |F| \leq 2 \), we have that local unbreakability is trivially satisfied. Otherwise, if \( F \) is a type-3 hyperedge, then the satisfaction of local unbreakability directly follows from the construction of \( f_F \).

- **Connectivity:** Choose an arbitrary \( F \in E(H) \) along with a tuple \( (\Gamma, \mu, l_1, l_2) \) in the domain of \( f_F \) such that \( f_F(\Gamma, \mu, l_1, l_2) = 1 \). If \( F \) is a type-1 hyperedge, then connectivity trivially holds. If \( F \) is a type-2 hyperedge, then connectivity follows from the construction of \( f_F \). Indeed, to see this, let us denote \( F = \{u, v\} \). Then, if \( \Gamma(u) \neq \Gamma(v) \), by the second and last cases in the definition of \( f_F \), we deduce that \( \hat{\Upsilon}(u) = \hat{\Upsilon}(v) \), and otherwise we contradict the supposition that \( f_F(\Gamma, \mu, l_1, l_2) = 1 \). Then, connectivity directly follows from the third and fourth cases.

Now, suppose that \( F = \sigma(\hat{t}) \) is a type-3 hyperedge, and say \( \Gamma : F \to [k]_0 \) is such that there exist \( i, j \in [k]_0 \), \( i \neq j \), satisfying \( |\Gamma^{-1}(i)| > 0 \) and \( |\Gamma^{-1}(j)| > 0 \). We need to show that \( l_1 + l_2 \geq 1 \). Since \( f_F(\Gamma, \mu, l_1, l_2) = 1 \), it holds that \( y[t, \Gamma, \mu + \mu', l_1 + l_1', l_2 + l_2'] = 1 \), where \( \mu', l_1' \) and \( l_2' \) are as defined at the construction of \( f_F \). Let \( \hat{\Upsilon} : \gamma(\hat{t}) \to [k]_0 \) denote some witnessing coloring for this condition. Since \( (T, \beta) \) is a highly connected tree decomposition, the Property 1 of such a decomposition implies that \( G^* = G[\gamma(\hat{t})] \setminus E(G[\sigma(\hat{t})]) \) is connected and that every vertex in \( \sigma(\hat{t}) \) has at least one vertex in \( V(G^*) \) that is its neighbor in \( G[\gamma(\hat{t})] \).

In particular, every two vertices in any \( \sigma(\hat{t}) \) are connected by a path in \( G^* \) (observe that \( V(G^*) = \gamma(\hat{t}) \) as we have only edges are discarded when \( G[\gamma(\hat{t})] \) is modified to be \( G^* \)). Let \( u \in \Gamma^{-1}(i) \) and \( v \in \Gamma^{-1}(j) \). Note that \( u \neq v \) and \( i \neq j \). Since \( u \) and \( v \) are connected by a path in \( G^* \), we derive that \( G^* \) has an edge \( e \) such that

\[
e \in \bigcup_{c, d \in [k]_0, c \neq d} E(\Upsilon^{-1}(c), \Upsilon^{-1}(d)) \setminus E(\sigma(\hat{t})).\]
Recall that \(\bigcup_{c,d\in[k]\times[c,d\neq d]} E(T^{-1}(c), T^{-1}(d)) = E(G[\hat{T}_v^{-1}(V_1)]) \cup E(G[\hat{T}_v^{-1}(V_2)])\). Therefore, we have that 
\(e \in (E(G[\hat{T}_v^{-1}(V_1)]) \cup E(G[\hat{T}_v^{-1}(V_2)])) \setminus E(G[\sigma(0)])\). Thus, by the inductive hypothesis, \(l_1 + l_2 \geq 1\).

- **Global Unbreakability:** Suppose that \(\alpha_P[\mu, l_1, l_2] = 1\). Then, by Lemma 5 there exists \(Y' : \gamma(t) \to [k]_0\) satisfying the properties listed in that lemma. From here, we get that 
\(\sum_{i,j\in[k]_0,i\neq j} |E(Y^{-1}(i), Y^{-1}(j))| \leq l_1 + l_2 = 0\). We argue that \(Y'|_{\beta(t)}\) is a witnessing coloring for global unbreakability, that is, this function is \((q,k)\)-unbreakable. In this context, we remind that \(q = (\eta + k)k\). To prove our argument, we first prove the following claim.

**Claim 7.** Suppose that there exists \(i \in [k]_0\) such that \(|Y^{-1}(i) \cap \beta(t)| > \eta + k\). Then, \(\sum_{j\in[k]_0,i\neq j} |Y^{-1}(j) \cap \beta(t)| \leq \eta + k\).

**Proof.** Suppose that the claim is false. Then, both \(|Y^{-1}(i) \cap \beta(t)| > \eta + k\) and \(\sum_{j\in[k]_0,i\neq j} |Y^{-1}(j) \cap \beta(t)| > \eta + k\). Thus,
\[
X = Y^{-1}(i) \cap \beta(t) \quad \text{and} \quad Y = \left( \bigcup_{j\in[k]_0,i\neq j} Y^{-1}(j) \cap \beta(t) \right) \cup \delta(Y^{-1}(i) \cap \beta(t))
\]
is a separation of order at most \(k\) of \(G[\gamma(t)]\) as we have already shown that
\[
\sum_{i,j\in[k]_0,i\leq j} |E(Y^{-1}(i), Y^{-1}(j))| \leq l_1 + l_2 \leq k_1 + k_2 \leq k.
\]
Moreover, \(|X \setminus Y) \cap \beta(t)| > \eta\) and \(|Y \setminus X) \cap \beta(t)| > \eta\), which contradicts that \(\beta(t)\) is \((\eta,k)\)-unbreakable in \(G[\gamma(t)]\).

Thus, if there exist \(i \in [k]_0\) as defined in Claim 7 then we are done. That is, we conclude that \(Y'|_{\beta(t)}\) is a \((q,k)\)-unbreakable. Otherwise, for all \(i \in [k]_0\), it holds that \(|Y^{-1}(i)| \leq \eta + k\). In particular, for any \(i \in [k]_0\), \(\sum_{j\in[k]_0,i\neq j} |Y^{-1}(j)| \leq (\eta + k)k = q\). Thus, we again conclude that \(Y'|_{\beta(t)}\) is \((q,k)\)-unbreakable.

Finally, we turn to address the statement complementary to the one of Lemma 8.

**Lemma 5.** For all \(\mu \in [n], l_1 \in [k]_0\) and \(l_2 \in [k]_0\), if \(y[t, Y^0, \nu, \mu, l_1, l_2] = 1\), then \(\alpha_P[\mu, l_1, l_2] = 1\).

**Proof.** Fix some \(\mu \in [n], l_1 \in [k]_0\) and \(l_2 \in [k]_0\) such that \(y[t, Y^0, \nu, \mu, l_1, l_2] = 1\). Our objective is to show that \(\alpha_P[\mu, l_1, l_2] = 1\). To this end, let \(Y\) be a witnessing coloring for \(y[t, Y^0, \nu, \mu, l_1, l_2] = 1\). We would like to prove that \(Y|_{\beta(t)}\) is a witnessing coloring for \(\alpha_P[\mu, l_1, l_2] = 1\), which would complete the proof of the lemma. To do so, we proceed as follows.

First, for any hyperedge \(F \in E(H)\), let us define \(\mu^F, l_1^F\) and \(l_2^F\) as follows.

- **If \(F\) is a type-1 hyperedge:** Set \(\mu^F = 1\) if \(\hat{T}_v(F) = V_1\), and \(\mu^F = 0\) otherwise. Set \(l_1^F = 0\) and \(l_2^F = 0\).

- **If \(F\) is a type-2 hyperedge:** Set \(\mu^F = 0\). If \(\hat{T}_v(u) \neq \hat{T}_v(v)\) and \(Y(u) = Y(v)\), set \(l_1^F = l_2^F = 0\). Otherwise, if \(\hat{T}_v(u) = \hat{T}_v(v)\), set \(l_1^F = 1\) and \(l_2^F = 0\), and if \(\hat{T}_v(u) = \hat{T}_v(v)\), set \(l_1^F = 0\) and \(l_2^F = 1\). The case where \(\hat{T}_v(u) \neq \hat{T}_v(v)\) and \(Y(u) \neq Y(v)\) cannot arise. Indeed, since \(Y\) is a witnessing coloring for \(y[t, Y^0, \nu, \mu, l_1, l_2] = 1\), we have that
\[
\bigcup_{i,j\in[k]_0,i\neq j} E(Y^{-1}(i), Y^{-1}(j)) = E(G[\hat{T}_v^{-1}(V_1)]) \cup E(G[\hat{T}_v^{-1}(V_2)])
\]
• **If** $F$ **is a type-3 hyperedge:** Denote $F = \sigma(\hat{t})$, where $\hat{t}$ is a child of $t$ in $T$. Set $\mu^F = |\hat{\mathcal{Y}}^{-1}_F(V_1) \cap (\gamma(\hat{t}) \cap \sigma(\hat{t}))|$, $l_1^F = |E(G[\hat{\mathcal{Y}}^{-1}_F(V_1) \cap (\gamma(\hat{t}))])| - |E(G[\hat{\mathcal{Y}}^{-1}_F(V_1) \cap \sigma(\hat{t}))]|$ and $l_2^F = |E(G[\hat{\mathcal{Y}}^{-1}_F(V_2) \cap \sigma(\hat{t}))]| - |E(G[\hat{\mathcal{Y}}^{-1}_F(V_2) \cap \sigma(\hat{t}))]|$.

Let us proceed by proving three claims that would together imply that $\mathcal{Y}|_{\beta(t)}$ is a witnessing coloring for $\text{aHP}[\mu, l_1, l_2] = 1$.

**Claim 8.** Let $\hat{t}$ be a child of $t$ in $T$, and let $i \in [k]_0$ be such that $|\mathcal{Y}^{-1}(i) \cap \sigma(\hat{t})| > 3k$. Then, $\sum_{j \in [k]_0, i \neq j} |\mathcal{Y}^{-1}(j) \cap \sigma(\hat{t})| \leq 3k$.

**Proof.** Suppose, by way of contradiction, that the claim is false. That is, we have that both $|\mathcal{Y}^{-1}(i) \cap \sigma(\hat{t})| > 3k$ and $\sum_{j \in [k]_0, i \neq j} |\mathcal{Y}^{-1}(j) \cap \sigma(\hat{t})| > 3k$. Consider the separation $(X, Y)$ of $G[\gamma(t)]$, where $X = \mathcal{Y}^{-1}(i)$ and $Y = (\gamma(t) \setminus \mathcal{Y}^{-1}(i)) \cup \delta(\mathcal{Y}^{-1}(i))$. Observe that $X \cap Y = \delta(\mathcal{Y}^{-1}(i))$. Since $\mathcal{Y}$ is a witnessing coloring for $y[t, \mathcal{Y}^\sigma, v, \mu, l_1, l_2]$, we have that

$$\bigcup_{i, j \in [k]_0, i \neq j} E(\mathcal{Y}^{-1}(i), \mathcal{Y}^{-1}(j)) = E(G[\hat{\mathcal{Y}}^{-1}_F(V_1)]) \cup E(G[\hat{\mathcal{Y}}^{-1}_F(V_2)])$$

and $|E(G[\hat{\mathcal{Y}}^{-1}_F(V_1)]) \cup E(G[\hat{\mathcal{Y}}^{-1}_F(V_2)])| \leq l_1 + l_2 \leq k + k \leq k$. Therefore, $|\delta(\mathcal{Y}^{-1}(i))| \leq k$, and thus the order of separation $(X, Y)$ is at most $k$. Moreover, since $|\mathcal{Y}^{-1}(i) \cap \sigma(\hat{t})| > 3k$, we have that $|(X \cap Y) \cap \sigma(\hat{t})| > 3k - k = 2k$, and since $\sum_{j \in [k]_0, i \neq j} |\mathcal{Y}^{-1}(j) \cap \sigma(\hat{t})| > 3k$, we also have that $|(Y \setminus X) \cap \sigma(\hat{t})| > 3k$. This implies that $\sigma(\hat{t})$ is not $(2k, k)$-unbreakable in $G[\gamma(t)]$, which mean that $\sigma(\hat{t})$ is not $(2k, k)$-unbreakable in $G[\gamma(\text{parent}(\hat{t}))]$. This is a contradiction to the fact that $(T, \beta)$ is a highly connected tree decomposition—specifically, it should satisfy Property 3 in Theorem 4.

Having Claim 8 at hand, we now verify that each function $f_F$ assigns 1 to the required tuple.

**Claim 9.** For any $F \in E(H)$, $f_F(\mathcal{Y}|_F, \mu^F, l_1^F, l_2^F) = 1$.

**Proof.** First, note that since $\mathcal{Y}$ is a witnessing coloring for $y[t, \mathcal{Y}^\sigma, v, \mu, l_1, l_2] = 1$, we have that $\mathcal{Y} \subseteq \mathcal{Y}_\sigma$. $A \cap \gamma(t) \subseteq \hat{\mathcal{Y}}^{-1}_F(V_1)$ and $B \cap \gamma(t) \subseteq \hat{\mathcal{Y}}^{-1}_F(V_2)$. Thus, from the construction of a type-1 hyperedge $F$ and the corresponding function $f_F$ with respect to $HP(k_1, k_2, n, \eta, q, H, (f_F)_{F \in E(H)})$, it is clear that $f_F(\mathcal{Y}|_F, \mu^F, l_1^F, l_2^F) = 1$. Second, suppose $F$ is a type-2 hyperedge. The specifications of $f_F$, together with our definition of $\mu^F, l_1^F, l_2^F$ directly implies that $f_F(\mathcal{Y}|_F, \mu^F, l_1^F, l_2^F) = 1$.

Third, suppose that $F$ is a type-3 hyperedge, and denote $F = \sigma(t_i)$ for some $t_i$ that is a child of $t$ in $T$. Note that $y[t, \mathcal{Y}|_F, v, \mu^F, \mu', l_1^F, l_2^F, l_1^F, l_2^F] = 1$ because $\mathcal{Y}|_{\gamma(t_i)}$ is a witnessing coloring for this equality, where $\mu' = |\hat{\mathcal{Y}}^{-1}_F(V_1) \cap \sigma(\hat{t_i})|$, $l_1' = |E(G[\hat{\mathcal{Y}}^{-1}_F(V_1) \cap \sigma(\hat{t_i})]|$ and $l_2' = |E(G[\hat{\mathcal{Y}}^{-1}_F(V_2) \cap \sigma(\hat{t_i})]|$. We now need to show that $\mathcal{Y}|_F$ is $(3k^2, k)$-unbreakable, as then we would be able to conclude that $f_F(\mathcal{Y}|_F, \mu^F, l_1^F, l_2^F) = 1$. By Claim 8 if there exists $i \in [k]_0$ such that $|\mathcal{Y}^{-1}(i) \cap \sigma(\hat{t_i})| > 3k$, then we deduce that $\mathcal{Y}|_{\sigma(\hat{t_i})}$ is $(3k^2, k)$-unbreakable. Otherwise, for all $i \in [k]_0$, $|\mathcal{Y}^{-1}(i) \cap \sigma(\hat{t_i})| \leq 3k$. Hence, for any $i \in [k]_0$, $\sum_{j \in [k]_0, i \neq j} |\mathcal{Y}^{-1}(j) \cap \sigma(\hat{t_i})| \leq 3k^2$. Thus, we have proved that $\mathcal{Y}|_F$ is $(3k^2, k)$-unbreakable.

Finally, we present our third claim.

**Claim 10.** $\mu = \sum_{F \in E(H)} \mu^F, \sum_{F \in E(H)} l_1^F \leq l_1$ and $\sum_{F \in E(H)} l_2^F \leq l_2$.

**Proof.** By the property of $(T, \beta)$ being a tree decomposition, for any two children $t_i$ and $t_j$ of $t$ in $T$, $\gamma(t_i) \cap \gamma(t_j) \subseteq \beta(t)$, and also from definition, $\sigma(t_i) \subseteq \beta(t)$ for any child $t_i$ of $t$. Now, note that $\mu = |\hat{\mathcal{Y}}^{-1}_F(V_1)|$. Thus, to show that $\mu = \sum_{F \in E(H)} \mu^F$, it is sufficient to show that $|\hat{\mathcal{Y}}^{-1}_F(V_1)| = \sum_{F \in E(H)} \mu^F$. However, keeping the above argument in mind, the claim that $|\hat{\mathcal{Y}}^{-1}_F(V_1)| = \sum_{F \in E(H)} \mu^F$ directly follows from the
satisfaction of the three following conditions. We remark that the satisfaction of these conditions is a direct consequence of the supposition that $\hat{\Upsilon}$ be a witnessing coloring for $y[t, \Upsilon^0, v, \mu, l_1, l_2] = 1$, together with our definition of the values $\mu^E, l_1^F$ and $l_2^F$.

1. For any type-1 hyperedge $F$, we have that $\mu^E = 1$ only if $\hat{\Upsilon}_v(F) = V_1$. In particular, $\sum_{F \in E(H)} \mu^E = |\hat{\Upsilon}_v^{-1}(V_1) \cap \beta(t)|$.

2. For any type-2 hyperedge $F$, $\mu^E = 0$. Thus, $\sum_{F \in E(H)} \mu^E = 0$.

3. For any type-3 hyperedge $F$, $|\hat{\Upsilon}_v^{-1}(V_1) \cap (\gamma(t_i) \setminus \sigma(t_i))|$.

Similarly, let us observe that $|E(G[\hat{\Upsilon}_v^{-1}(V_1)])| \leq l_1$. Thus, to show that $\sum_{F \in E(H)} l_1^F \leq l_1$, it is sufficient to show that $\sum_{F \in E(H)} l_1^F \leq |E(G[\hat{\Upsilon}_v^{-1}(V_1)])|$. However, the latter inequality that directly follows from the satisfaction of all of the following conditions.

1. For any type-1 hyperedge $F$, $l_1^F = 0$. Thus, $\sum_{F \in E(H)} l_1^F = 0$.

2. For any type-2 hyperedge $F = \{u, v\}$, $l_1^F = 1$ only if $\hat{\Upsilon}_v(u) = \hat{\Upsilon}_v(v) = V_1$. In particular, $\sum_{F \in E(H)} l_1^F = |E(G[\hat{\Upsilon}_v^{-1}(V_1)]) \cap E(G[\beta(t)])|$.

3. For any type-3 hyperedge $F$, $|E(G[\hat{\Upsilon}_v^{-1}(V_1) \cap (\gamma(t_i) \setminus \sigma(t_i))]| \leq l_1^F$.

Symmetrically, $\sum_{F \in E(H)} l_2^F \leq l_2$. This concludes the proof of the claim. □

As we have proved Claims 9 and 10, we derive that $\Upsilon|_{\beta(t)}$ is a witnessing coloring for $a_{\text{HP}}[\mu, l_1, l_2] = 1$. This concludes the proof of the lemma.

Recall that we have argued that to prove Theorem 5 it is sufficient to show that the current set of values $y[t, \Upsilon^0, v, \mu, l_1, l_2]$ can be computed in time $2^{O(1)n^{O(1)}}$. Here, $n$ refers to $|V(G)|$. By Lemmas 3 and 5, this set of values can be derived from the solution of HP($k_1, k_2, n, \eta, q, H, \{f_F\}|_{F \in E(H)}$). Since HP($k_1, k_2, n, \eta, q, H, \{f_F\}|_{F \in E(H)}$) is a favorable instance of HP (by Lemma 1), the algorithm given by Theorem 5 solves it in time $2^{O((\min(k,q) \log(k+q))d^{O(k^2)}|E(H)|^{O(1)})} = 2^{k^{O(1)}n^{O(1)}}$.

6 Solving Favorable Instances of HP

Recall the problem statement of Hypergraph Painting (HP) and the definition of a favorable instance of HP from Section 6. In this section, we prove Theorem 5. For this purpose, let $HP(k_1, k_2, b, d, q, H, \{f_F\}|_{F \in E(H)})$ be a favorable instance of HP. We aim to show how to compute $a_{\text{HP}}[\mu, l_1, l_2]$ in time $2^{O((\min(k,q) \log(k+q))d^{O(k^2)}|E(H)|^{O(1)})} = 2^{k^{O(1)}n^{O(1)}}$ for an arbitrarily fixed choice of $0 \leq \mu \leq b$, $0 \leq l_1 \leq k_1$ and $0 \leq l_2 \leq k_2$. Since there are only $(b+1)(k_1+1)(k_2+1)$ choices for such $\mu, l_1$ and $l_2$, we would thus indeed verify the correctness of Theorem 5.

6.1 Classifying Hyperedges

We begin by analyzing the structure of the input instance $HP(k_1, k_2, b, d, q, H, \{f_F\}|_{F \in E(H)})$ under the assumption that $a_{\text{HP}}[\mu, l_1, l_2] = 1$. Recall that $k = k_1 + k_2$. Then, by the property of global unbreakability, there exists a witnessing coloring $\Upsilon : V(H) \to [k_0]$ such that $\sum_{j \in [k_0]} |\Upsilon^{-1}(j)| \leq q$ for some index $i \in [k_0]$.

Without loss of generality, suppose that $i = 0$ is such an index, that is, $\sum_{j \in [k]} |\Upsilon^{-1}(j)| \leq q$. In the forthcoming arguments, we aim to elucidate the behavior of the function that is the restriction of the witnessing coloring.
Lemma 9. \( \sum \Upsilon \) efficiently. In addition to this, as we shall see in Lemma 8, the total number of possible restrictions \( \sum \Upsilon \) witnesses the global unbreakability property, we assumed w.l.o.g. that \( \Upsilon \) with respect to each hyperedge of the hypergraph \( H \). To achieve such a property, we perform several phases of color coding of the hypergraph (here, these phases of color coding are hidden under a layer of derandomization tools). These phases would exploit the properties of a favourable instance, and eventually highlight a “nice” structure that would help us achieve our goal.

To proceed with the implementation of the above-mentioned idea, we first categorize the hyperedges of \( H \) into the following types, based on the witnessing coloring \( \Upsilon \). In this context, we remind that the notation \( \sum \Upsilon \) indicates that for all \( a \in A' \), it holds that \( f(a) = b \) (see Section 2).

- Let \( E_b = \{ F \in E(H) : \Upsilon(F) = 0 \} \). Here, ‘\( b \)’ stand for big.
- For each \( i \in [k] \), let \( E_{s_i} = \{ F \in E(H) : \Upsilon(F) = i \} \). Here, ‘\( s \)’ stands for small.
- Let \( E_m = \{ F \in E(H) : \text{there exist } u,v \in F \text{ such that } \Upsilon(u) \neq \Upsilon(v) \} \). Here, ‘\( m \)’ stands for multichromatic.

Observe that each hyperedge \( F \in E(H) \) belongs to exactly one of the sets \( E_b, E_m, E_{s_1}, \ldots, E_{s_k} \). Furthermore, let \( E'_{s_i} \) denote the edge set of some arbitrary spanning forest of the hypergraph on the vertex set \( V(H) \) and the edge set \( E_s \). Let \( E_s = \bigcup_{i \in [k]} E'_{s_i} \) denote the union of these edge sets. Since we are working with a favourable instance of \( HP \), we would see (in Lemmas 6 and 7) that the sizes of the sets \( E_s \) and \( E_m \) can be upper bounded by \( q \) and \( k \), respectively. We exploit these bounds to highlight the hyperedges in \( E_m \) and \( E_s \) (Lemma 9) efficiently. In addition to this, as we shall see in Lemma 8 the total number of possible restrictions of \( \Upsilon \) on any hyperedge can also bounded effectively. Thus, we can not only highlight the hyperedges in \( E_m \) and \( E_s \), but we can also guess the restrictions of \( \Upsilon \) to these hyperedges. We remark that since we aim to solve a favourable instance of \( HP \) in time that is proportional to a single exponential function of \( q \), we do not guess the restriction of \( \Upsilon \) to the hyperedges of \( E_s \) straightaway (as \( |E_s| \leq q \) from Lemma 6). The proof of Lemma 9 would capture the idea of the performance of highlighting and guessing. As one would expect, this highlighting does concludes our arguments, as it does not just highlight the hyperedges in \( E_m \) and \( E_s \), but also some hyperedges from \( E_b \). We deal with the inherent challenges of handling such a “messy picture” later in our proof.

Lemma 6. \( |E_s| \leq q \).

Proof. Recall that for each \( i \in [k] \), we defined \( E'_{s_i} \) as the edge set of a spanning forest of the hypergraph with the vertex set \( V(H) \) and the edge set \( E_{s_i} \). Hence, by this definition, \( |E'_{s_i}| \leq |\Upsilon^{-1}(i)| \). Now, recall that since \( \Upsilon \) witnesses the global unbreakability property, we assumed w.l.o.g. that \( \sum_{i \in [k]} |\Upsilon^{-1}(i)| \leq q \). We thus have that \( \sum_{i \in [k]} |E'_{s_i}| \leq q \). Therefore, \( |E_s| \leq q \). \( \square \)

Lemma 7. \( |E_m| \leq k \).

Proof. Since \( aHP[\mu, l_1, l_2] = 1 \), for all \( F \in E(H) \), there exist \( \mu^F \), \( l_1^F \) and \( l_2^F \) such that \( f_F(\Upsilon_{|F}^F, \mu^F, l_1^F, l_2^F) = 1 \). Hence, the connectivity property implies that for each \( F \in E_m \), we have that \( l_1^F + l_2^F \geq 1 \). However, \( \sum_{F \in E(H)} l_1^F + l_2^F \leq l_1 + l_2 \leq k_1 + k_2 = k \). Thus, \( |E_m| \leq k \). \( \square \)
6.2 Introducing Good Assignments

Let us first note that by Lemma [1] for any hyperedge \( F \in E(H) \), the number of \((3k^2, k)\)-unbreakable functions (that we call \((3k^2, k)\)-unbreakable colorings) from \( F \) to \([k]_0\) is at most \( \alpha = \sum_{i=1}^{k^2} (d^i) \cdot (3k^2)^k \cdot (k + 1) = d^{O(k^2)} \).

For each hyperedge \( F \), let us arbitrarily order all possible \((3k^2, k)\)-unbreakable colorings. For each \( i \in [\alpha] \), let \( \lambda_{F,i} \) denote the \( i \)-th such coloring. If for an hyperedge \( F \), the number of such colorings is strictly smaller than \( \alpha \), then we extend its list of possible colorings to be of size \( \alpha \) by letting some colorings be present multiple times. Thus, for each \( F \in E(H) \) and \( i \in [\alpha] \), we ensure \( \lambda_{F,i} \) is well-defined.

**Lemma 8.** For any \( F \in E(H) \), there exists \( i \in [\alpha] \) such that \( \Upsilon|_F = \lambda_{F,i} \).

**Proof.** Since \( \Upsilon \) is a witnessing coloring for \( \text{AHF}[\mu, l_1, l_2] = 1 \), we have that for any \( F \in E(H) \) there exist \( \mu^F \), \( \ell_1^F \) and \( \ell_2^F \) such that \( f_F(\Upsilon|_F, \mu^F, \ell_1^F, \ell_2^F) = 1 \). Since the property of local unbreakability is then enforced by the definition of \( f_F \), we have that \( \Upsilon|_F \) is a \((3k^2, k)\)-unbreakable coloring. Since \( \{\lambda_{F,1}, \ldots, \lambda_{F,\alpha}\} \) contains all possible \((3k^2, k)\)-unbreakable colorings from \( F \) to \([\alpha] \), there exists \( i \in [\alpha] \) such that \( \Upsilon|_F = \lambda_{F,i} \). \( \square \)

Here, we are interested in assignments that are functions associating each hyperedge \( F \in E(H) \) with a coloring \( \lambda_{F,i} \). Let us proceed by defining which assignments would be useful for us to have at hand.

**Definition 9.** An assignment \( p : E(H) \to [\alpha]_0 \) is said to be a good assignment if the following conditions hold.

1. For all \( F \in E_s \), \( p(F) = 0 \).
2. For all \( F \in E_m \), \( p(F) = i > 0 \) and \( \Upsilon|_F = \lambda_{F,i} \).

To employ coloring coding, we first mention the required derandomization tools.

**Proposition 4 (Lemma 1.1, [1]).** Given a set \( U \) of size \( n \) and \( y, z \in [n]_0 \), we can construct in time \( O(2^{O(\min(y,z) \log(y+z))} n \log n) \) a family \( F \) of at most \( O(2^{O(\min(y,z) \log(y+z))} \log n) \) subsets of \( U \), such that the following holds: for all sets \( Y, Z \subseteq U \) such that \( Y \cap Z = \emptyset, |Y| \leq y \) and \( |Z| \leq z \), there exists a set \( S \in F \) with \( Y \subseteq S \) and \( Z \cap S = \emptyset \).

**Definition 10 ((\(N, r\))-perfect family).** An \((N, r)\)-perfect family is a family of functions from \([N] \) to \([r] \), such that for any subset \( X \subseteq [N] \) of size \( r \), there exists a function in the family that is injective on \( X \).

**Proposition 5 ([3]).** An \((N, r)\)-perfect family of size \( O(e^r O(\log r) \log N) \) can be computed in time \( O(e^r O(\log r) N \log N) \).

We are now ready to present our color coding phases.

**Lemma 9.** There exists a set \( \mathcal{A} \) of assignments from \( E(H) \) to \([\alpha]_0 \), such that \( |\mathcal{A}| \leq 2^{O(\min(k, q) \log(k+q))} \cdot \alpha d^{O(k^2)} \cdot \log |E(H)| \) and there exists a good assignment in \( \mathcal{A} \). Moreover, such a set \( \mathcal{A} \) is computable in time \( O(2^{O(\min(k, q) \log(k+q))} \cdot \alpha d^{O(k^2)} \cdot |E(H)|^{O(1)}) \).

**Proof.** We start by defining three several families, which would guide us through the construction of \( \mathcal{A} \). For \( U = E(H) \), \( y = k \) and \( z = q \), let \( \mathcal{F} = \{S_1, \ldots, S_p\} \) be the family of size \( \nu = 2^{O(\min(k, q) \log(k+q))} \log |E(H)| \) obtained by calling the algorithm of Proposition [1]. For each \( j \in [\nu] \), let \( \mathcal{P}_j \) be a \((|S_j|, k)-\)perfect family of size at most \( \zeta = e^k \cdot k^{O(k^2)} \log |S_j| \cdot k^{O(k^2)} \cdot |E(H)| \) computed by the algorithm of Proposition [5]. Let \( \mathcal{Q} \) be the family of all possible functions from \([k] \) to \([\alpha] \). Observe that \( |\mathcal{Q}| = \alpha^k \).

For each set \( S_j \in \mathcal{F} \), function \( \kappa \in \mathcal{P}_j \) and function \( \kappa_0 \in \mathcal{Q} \), let \( p[S_j, \kappa, \kappa_0] : E(H) \to [\alpha]_0 \) be defined as follows.

\[
p[S_j, \kappa, \kappa_0](F) = \begin{cases} 0, & \text{if } F \in S_j \\ \kappa_0(\kappa(F)), & \text{otherwise} \end{cases}
\]
Let $\mathcal{A} = \{p[S_j, \kappa, \kappa_0] : S_j \in \mathcal{F}, \kappa \in \mathcal{P}_j, \kappa_0 \in \mathcal{Q}\}$. We claim that there exists a good assignment in $\mathcal{A}$. Since $|E_m| \leq k$ (from Lemma 7) and $|E_s| \leq q$ (from Lemma 8), from Proposition 4 there exists $S_j \in \mathcal{F}$ such that $E_m \subseteq S_j$ and $E_s \cap S_j = \emptyset$. By Proposition 5, there exists a function $\kappa \in \mathcal{P}_j$ which is injective on $E_m$. Let $E_m = \{F_1, \ldots, F_r\}$ where $c \leq k$. Without loss of generality, $\kappa(E_p) = y$ for all $y \in [c]$. Since $\mathcal{Q}$ contains all possible functions from $[k]$ to $[\alpha]$, and for each $F \in E_m$ there exists $i \in [\alpha]$ such that $\Upsilon|_F = \lambda_{F,i}$ (from Lemma 5), there exists $\kappa_0 \in \mathcal{Q}$ such that for each $F \in E_m$, $\Upsilon|_F = \lambda_{F,\kappa_0(\kappa(F))}$. Moreover, since $E_s \cap S_j = \emptyset$, we have that $p[S_j, \kappa, \kappa_0] = 0$. Thus, $p[S_j, \kappa, \kappa_0] \in \mathcal{A}$ is a good assignment.

Recall that $\alpha = d^{O(k^2)}$. Now, as we have upper bounded $\nu$ and $\zeta$, we observe that $|\mathcal{A}| \leq \nu\zeta\alpha^k = 2^{O(\min(k,q)\log(k+q))}k^k d^{O(k^2)} \log^2|E(H)|$. Thus, the size of $\mathcal{A}$ is upper bounded by $2^{O(\min(k,q)\log(k+q))} \cdot d^{O(k^2)} \cdot \log^2|E(H)|$.

The time taken to compute $\mathcal{A}$ is proportional to the time taken to compute $\mathcal{F}, \mathcal{P}_j$ for each $j \in \{\nu\}$ and $\mathcal{Q}$. By Propositions 4 and 5 we thus derive that the running time is upper bounded $O(2^{O(\min(k,q)\log(k+q))} \cdot d^{O(k^2)} \cdot |E(H)|^{O(1)})$.

The algorithm we design to compute $\text{aHP}[\mu, l_1, l_2]$ first constructs the set $\mathcal{A}$ of Lemma 9. Observe that this computation can be done regardless of whether $\text{aHP}[\mu, l_1, l_2] = 1$ or $\text{aHP}[\mu, l_1, l_2] = 0$. (We only use the supposition that $\text{aHP}[\mu, l_1, l_2] = 1$ to analyze structural properties of an input instance satisfying this condition.) Next, the algorithm branches on all possible assignments in $\mathcal{A}$. By Lemma 9 assuming that $\text{aHP}[\mu, l_1, l_2] = 1$, we know that there exists at least one assignment from $E(H)$ to $[\alpha]_0$ that is good. Henceforth, we assume that we currently consider a branch that corresponds to a good assignment, denoted by $p : E(H) \to [\alpha]_0$. Thus, we would like to show that we correctly determine at the current branch that $\text{aHP}[\mu, l_1, l_2] = 1$. (If it were the case that $\text{aHP}[\mu, l_1, l_2] = 0$, it would also be clear from our arguments that we would not determine that $\text{aHP}[\mu, l_1, l_2] = 1$, which would overall imply that no branch determines that this condition holds, and hence we would eventually decide that $\text{aHP}[\mu, l_1, l_2] = 0$.)

### 6.3 Associating the Graph $L_p$ with an Assignment $p$

For our assignment $p : E(H) \to [\alpha]_0$, let us now construct an undirected simple graph $L_p$ with $V(L_p) = V(H)$. For each $F \in E(H)$ such that $p(F) = 0$, make $F$ a clique in $L_p$. We say that the edges of this clique are the edges that correspond to the hyperedge $F$. For any $F \in E(H)$ such that $p(F) = 1 > 0$, for each $j \in [k]_0$, make the set $\lambda_{F,i}^{-1}(j)$ a clique in $L_p$. We say that the edges of all such cliques are the edges that correspond to the hyperedge $F$. Since we want $L_p$ to be a simple graph, between any two vertices of $L_p$ we retain at most one copy of the edge between them (if one exists). If a deleted copies of some edge $e$ in $L_p$ corresponds to a hyperedge $F$, then in the simple graph the retained copy of that edge $e$ is the one that is said to correspond to that hyperedge $F$ (even if we originally added the retained copy of $e$ due to a different hyperedge). Note that it may thus be the case that one edge in $L_p$ corresponds to several hyperedges in $E(H)$.

We proceed by analyzing the connected components of $L_p$. Informally, we first argue that every connected component behaves as a single unit with respect to $\Upsilon$.

**Lemma 10.** Let $D$ be any connected component of $L_p$. Then, $\Upsilon(D) = i$ for some $i \in [k]_0$, that is, all the vertices in $D$ are assigned the same color by $\Upsilon$.

**Proof.** For any $\mathcal{F} \subseteq E(H)$, let $L_p[\mathcal{F}]$ be the simple graph on the same vertex set as $L_p$, whose edge set contains only those edges of $L_p$ that correspond to some hyperedge in $\mathcal{F}$. Observe that $L_p[E(H)] = L_p$. Moreover, observe that if a set of vertices is connected in $L_p[\mathcal{F}]$ then it is also connected in $L_p[\mathcal{F}']$ for any $\mathcal{F} \supseteq \mathcal{F}'$.

Let $E(H) = \{F_1, \ldots, F_r\}$. Moreover, for any $j \in [r]$, denote $\mathcal{F}_j = \bigcup_{c=1}^r F_c$. Let us prove by induction on $j$ that for each component $D$ of $L_p[\mathcal{F}_j]$, we have that $\Upsilon(D) = i$ for some $i \in [k]_0$. The proof of this claim would conclude the proof of the lemma, as by setting $j = r$, we thus derive that for each component $D$ of $L_p[\mathcal{F}_r] = L_p$, we have that $\Upsilon(D) = i$ for some $i \in [k]_0$. Hence, we next focus only on the proof of the claim.

To prove the base case, where $j = 1$, consider the graph $L_p[\mathcal{F}_1]$. If $F_1 \notin E_m$, then $\Upsilon(F_1) = i$ for some $i \in [k]_0$ (by the definition of $E_m$). Hence, for each connected component $D$ of $L_p[\mathcal{F}_1]$, $\Upsilon(D) = i$ for some
Let us now prove that if \( p \) is a good assignment, then \( \lambda_{F_i,s} = \Upsilon | F_i \). Since each component \( D \) of \( L_{p}[F_{j-1}] \) is either an isolated vertex or \( \lambda_{F_i,s}^{-1}(i) \) for some \( i \in [k]_0 \), we conclude that \( \Upsilon(D) = i \) for some \( i \in [k]_0 \).

We now suppose that \( j \geq 2 \). By induction hypothesis, for each connected component \( D \) of \( L_p[F_{j-1}] \), we have that \( \Upsilon(D) = i \) for some \( i \in [k]_0 \). Let us now examine the graph \( L_p[F_j] \) and the hyperedge \( F_j \). Note that \( F_j = F_{j-1} \setminus F_{j-1}. \) If \( F_j \notin E_m \), then \( \Upsilon(F_j) = i \) for some \( i \in [k]_0 \) (from the definition of \( E_m \)). Let \( D \) be the collection of all connected components of \( L_p[F_{j-1}] \) which intersects \( F_j \). Then, the definition of \( L_p \) and the inductive hypothesis directly imply that \( \Upsilon(D) = i \) for some \( i \in [k]_0 \). Thus, by the inductive hypothesis, for each connected component \( D \) of \( L_p[F_j] \), we have that \( \Upsilon(D) = i \) for some \( i \in [k]_0 \). Otherwise, \( F_j \in E_m \). Then, denoted \( p(F_j) = s > 0 \). Since \( p \) is a good assignment, \( \lambda_{F_1,s} = \Upsilon | F_1 \). For each \( i \in [k]_0 \), let \( D_i \) be the collection of all connected components of \( L_p[F_{j-1}] \) that intersect \( \lambda_{F_1,s}^{-1}(i) \). Then, the definition of \( L_p \) and the inductive hypothesis directly imply \( \Upsilon(D_i) = i \). Hence, by the inductive hypothesis, for each connected component \( D \) of \( L_p[F_{j-1}] \), we have that \( \Upsilon(D) = i \) for some \( i \in [k]_0 \).

Roughly speaking, we now argue that hyperedges crossing several different components, where to at least one of them \( \Upsilon \) assigns some \( i > 0 \), should belong to \( E_m \).

**Lemma 11.** Let \( D \) be any connected component of \( L_p \) such that \( \Upsilon(D) = i > 0 \) for some \( i \in [k] \). For any \( F \in E(H) \) such that \( F \cap D \neq \emptyset \) and \( F \setminus D \neq \emptyset \), then \( F \notin E_m \).

**Proof.** Suppose that the statement is false, that is, there exists \( F \in E(H) \setminus E_m \) such that \( F \cap D \neq \emptyset \) and \( F \setminus D \neq \emptyset \). Since \( F \notin E_m \), \( F \cap D \neq \emptyset \) and \( \Upsilon(D) = i \), we have that \( j = i \), that is, \( F \in E_s \). Consider any spanning forest \( E_s' \) of the hypergraph with vertex set \( V(H) \) and edge set \( E_s \). Observe that by the definition of \( L_p \), any spanning forest \( E_s \) of \( F \) is present in some single tree of that spanning forest. Therefore, there exists some \( F' \in E_s \) where \( F' \) could be the same as the hyperedge \( F \), such that the vertices of \( F' \) form a superset of \( F \). Since \( p \) is a good assignment, \( p(F') = 0 \). Thus, the definition of \( L_p \) implies that all the vertices of \( F \) belong to the same connected component, which contradicts that \( F \setminus D \neq \emptyset \).

### 6.4 Rules to Modify a Good Assignment

We first modify the good assignment \( p \) by applying the following two rules exhaustively, prioritizing Rule 1 over Rule 2. Note that whenever we change \( p \), we update \( L_p \) accordingly.

**Rule 1:** If there exist a connected component \( D \) of \( L_p \) and a hyperedge \( F \in E(H) \) such that \( F \subseteq D \) and \( p(F) > 0 \), then update \( p(F) = 0 \).

**Rule 2:** If there exist a connected component \( D \) of \( L_p \), vertices \( v_1, v_2 \in D \) (\( v_1 \) could be equal to \( v_2 \)) and hyperedges \( F_1, F_2 \in E(H) \) (\( F_1 \) could be equal to \( F_2 \)) such that \( F_1 \cap D \neq \emptyset \), \( F_2 \cap D \neq \emptyset \), \( F_1 \setminus D \neq \emptyset \), \( F_2 \setminus D \neq \emptyset \), \( p(F_1) = i > 0 \), \( p(F_2) = j > 0 \), \( \lambda_{F_1,i}(v_1) \in [k] \) and \( \lambda_{F_2,j}(v_2) = 0 \), then update \( p(F_1) = 0 \).

**Lemma 12.** After any application of Rule 1 and Rule 2, \( p \) remains a good assignment.

**Proof.** Let us first prove that if \( p \) was a good assignment, then after the application of Rule 1, the modified \( p \) is still a good assignment. From Lemma 10, \( \Upsilon(D) = i \) for some \( i \in [k]_0 \). Thus, if \( F \subseteq D \), then \( F \notin E_m \). Hence, when we redefine \( p(F) = 0 \), \( p \) remains a good assignment.

Let us now prove that if \( p \) was a good assignment, then after the application of Rule 2, the modified \( p \) is still a good assignment. To prove this, it is enough to prove that \( F_1 \notin E_m \). Suppose, for the sake of contradiction, that \( F_1 \in E_m \). Since \( p \) is a good assignment, \( \lambda_{F_1,i} = \Upsilon | F \). Denote \( \lambda_{F_1,i}(v_1) = c \), where \( c \in [k] \). Since \( v_1 \in D \) and \( \lambda_{F_1,i}(v_1) = c > 0 \), from Lemma 10 \( \Upsilon(D) = c > 0 \). From Lemma 11 \( F_2 \in E_m \). Again, since
\[ p \] is a good assignment, \( \lambda_{F,j} = \Upsilon|_F \). Since \( \lambda_{F,j}(v_2) = 0 \) and \( v_2 \in D \), this implies that \( \Upsilon(D) = 0 \), which is a contradiction.

For each connected component \( D \) of \( L_p \), let us now define a label set \( L(D) \subseteq [k]_0 \) as follows. For any \( i \in [k]_0 \), we insert \( i \) into \( L(D) \) if and only if there exists \( F \in E(H) \) such that \( F \cap D \neq \emptyset \), \( p(F) = j > 0 \) and \( \lambda_{F,j}(F \cap D) = i \). Observe that \( L(D) \) could be empty.

Let us now turn to analyze the labels sets we have just defined.

**Lemma 13.** Let \( D \) be a connected component of \( L_p \) such that \( L(D) = \emptyset \). Then, for any \( F \in E(H) \) such that \( F \cap D \neq \emptyset \), \( F \setminus D = \emptyset \).

**Proof.** Observe that if there exists \( F \in E(H) \) such that \( p(F) > 0 \) and \( F \cap D \neq \emptyset \), then \( |L(D)| \geq 1 \). Therefore, if \( L(D) = \emptyset \), then for all \( F \in E(H) \) such that \( F \cap D \neq \emptyset \), we have that \( p(F) = 0 \). Thus, from the construction of \( L_p \), we have that \( F \setminus D = \emptyset \).

**Lemma 14.** For any connected component \( D \) of \( L_p \), if \( \Upsilon(D) = i > 0 \), then either \( L(D) = \emptyset \) or \( |L(D)| \geq 2 \).

**Proof.** Suppose that \( L(D) \neq \emptyset \). Then, there exists \( F \in E(H) \) such that \( F \cap D \neq \emptyset \) and \( p(F) = j > 0 \). Let \( \lambda_{F,j}(F \cap D) = s \). We will now show that \( s = i \). First of all, let us argue that \( F \setminus D = \emptyset \). Indeed, if \( F \setminus D \neq \emptyset \), then \( F \subset D \). In this case, since \( p \) is a good assignment, where Rule 1 has been exhaustively applied, \( p(F) \) should be equal to 0, which is a contradiction. Thus, \( \Upsilon(D) = i > 0 \), \( F \cap D \neq \emptyset \) and \( F \setminus D = \emptyset \), from Lemma 11 we have that \( F \in E_m \). Then, since \( p \) is a good assignment, \( \lambda_{F,j}(F \cap D) = \Upsilon|_F \). Since \( \Upsilon(D) = i \), we derive that indeed \( \lambda_{F,j}(F \cap D) = i \). Thus, \( L(D) = \{i\} \).

By Lemma 14 we have that if for a connected component \( D \) of \( L_p \), either \( L(D) = \emptyset \) or \( |L(D)| \geq 2 \), then \( \Upsilon(D) = 0 \).

**Lemma 15.** If \( D \) is a connected component of \( L_p \) such that \( L(D) = \{l_d\} \), then either \( \Upsilon(D) = l_d \) or \( \Upsilon(D) = 0 \).

**Proof.** Since \( L(D) = \{l_d\} \), there exists \( F \in E(H) \) such that \( p(F) = i > 0 \), \( F \cap D \neq \emptyset \) and \( \lambda_{F,i}(F \cap D) = l_d \). Denote \( \Upsilon(D) = j \), and suppose that \( j \neq 0 \), else we are done. Since \( j \neq 0 \), from Lemma 11 we have that \( F \in E_m \). Then, since \( p \) is a good assignment, \( \lambda_{F,i} = \Upsilon|_F \). Finally, since all the vertices of \( D \) are assigned the same color by \( \Upsilon \) (by Lemma 10), we have that \( \Upsilon(D) = l_d \).

For a connected component \( D \) of \( L_p \) such that \( |L(D)| \geq 2 \), let us redefine the label set of \( D \) to be \( L(D) = \emptyset \). Now, for any connected component \( D \) of \( L_p \), \( |L(D)| \leq 1 \). Moreover, if \( L(D) = \emptyset \), then \( \Upsilon(D) = 0 \) (by Lemma 14). We call a connected component \( D \) of \( L_p \) such that \( L(D) = \{0\} \) a 0-component.

Let us continue modifying the good assignment \( p \), now with the following rule. Again, whenever we modify \( p \), we update \( L_p \) accordingly.

**Rule 3:** If there exist \( F \in E(H) \) and two distinct 0-connected components of \( L_p \), \( D_1 \) and \( D_2 \), such that \( F \cap D_1 \neq \emptyset \) and \( F \cap D_2 \neq \emptyset \), then update \( p(F) = 0 \).

**Lemma 16.** The assignment resulting by applying Rule 3 to \( p \) is a good assignment.

**Proof.** To prove the lemma, it is sufficient to show that \( F \notin E_m \). Suppose that this claim is false, that is, \( F \in E_m \) and hence after the update, we obtain an assignment that is not good. Since (the original) \( p \) is a good assignment, we have that \( p(F) = i > 0 \) such that \( \lambda_{F,i} = \Upsilon|_F \). Since \( D_1 \) and \( D_2 \) are different connected components of \( L_p \), \( (F \cap D_1) \subseteq \lambda_{F,i}^{-1}(j_1) \), \( (F \cap D_2) \subseteq \lambda_{F,i}^{-1}(j_2) \) and \( j_1 \neq j_2 \). However, since \( D_1 \) and \( D_2 \) are 0-components of \( L_p \), \( \Upsilon(D_1) = 0 \) and \( \Upsilon(D_2) = 0 \). This contradicts that \( \lambda_{F,i} = \Upsilon|_F \). Therefore, \( F \notin E_m \).

To further analyze 0-components, define \( B \) as the set containing every vertex \( v \in V(H) \) such that \( \Upsilon(v) = 0 \) and there exists \( F \in E_m \) that is incident to \( v \).

**Lemma 17.** Let \( D \) be a connected component of \( L_p \) containing a vertex \( v \in B \). Then, \( D \) is a 0-component.
Proof. From the definition of the set $B$, there exists $F \in E_m$ such that $v \in F$. Since $p$ is a good assignment, $p(F) = i > 0$ such that $\lambda_{p,i} = \Upsilon|_F$. Since $\Upsilon(v) = 0$, $v \in F$ and $v \in D$, we have that $\lambda_{F,i}(F \cap D) = 0$. Hence, $0 \in L(D)$. Therefore, by Lemma \ref{lem:lem}, we conclude that $D$ is a 0-component of $L_p$. \hfill $\square$

### 6.5 Constructing a Supergraph $L_p^*$ of $L_p$

Let us now construct another simple undirected graph $L_p^*$, which is a supergraph of $L_p$ with the same vertex set as of $L_p$ and the following additional edges. If there exist $F \in E(H)$ and two distinct connected components of $L_p$, $D_1$ and $D_2$, such that $F \cap D_1 \neq \emptyset$, $F \cap D_2 \neq \emptyset$, $L(D_1) \neq \emptyset$, and $L(D_2) \neq \emptyset$, then insert an edge between some vertex of $D_1$ and some vertex of $D_2$ into $L_p^*$. Clearly, any connected component $D$ of $L_p$ is contained in some connected component of $L_p^*$. This leads us to the following definition.

**Definition 11.** Given a connected component $D^*$ of $L_p^*$, we say that a connected component $D$ of $L_p$ is a constituent of $D^*$ if $D \subseteq D^*$.

A component $D^*$ of $L_p^*$ is called a 0-component of $L_p^*$ if it has only one constituent component and that constituent component is a 0-component in $L_p$. We now proceed to analyze the new graph $L_p^*$.

**Lemma 18.** Let $D^*$ be some connected component of $L_p^*$ that has a constituent component $D$ such that $L(D) = \emptyset$. Then, $D$ is the only constituent component of $D^*$, that is, $D^* = D$.

**Proof.** By Lemma \ref{lem:lem} for any $F \in E(H)$ such that $F \cap D \neq \emptyset$, we have that $F \setminus D = \emptyset$. Thus, by the construction of $L_p^*$, it holds that $D^* = D$. \hfill $\square$

**Lemma 19.** For any $F \in E(H)$, either $F \subseteq D^*$ for some connected component $D^*$ of $L_p^*$, or $F$ intersects exactly two connected components of $L_p^*$, a 0-connected component $D^*_1$ of $L_p^*$ and a non 0-connected component $D^*_2$ of $L_p^*$.

**Proof.** Suppose that there exists $F \in E(H)$ such that for any connected component $D^*$ of $L_p^*$, $F \not\subseteq D^*$, else we are done. First, observe that in this case $p(F) > 0$, as otherwise $F \subseteq D$ for a connected component $D$ of $L_p$, which would imply that $F$ is contained in a connected component of $L_p^*$. We claim that $F$ intersects at most one 0-component of $L_p^*$. To show this, suppose by way of contradiction, that $F$ intersects at least two 0-components of $L_p^*$, which we denote by $D^*_1$ and $D^*_2$. From the definition of a 0-component in $L_p^*$, it follows that $D^*_1$ and $D^*_2$ are also different 0-components in $L_p^*$. Since $p$ is a good assignment and Rule 3 is no longer applicable, we have that $p(F)$ should be equal to 0, which is a contradiction. Therefore, $F$ can intersect at most one 0-component of $L_p^*$. From the construction of $L_p^*$, observe that $F$ cannot intersect more than one non 0-connected component of $L_p^*$. Hence, we conclude that if $F \not\subseteq D^*$, for some connected component $D^*$ of $L_p^*$, then $F$ intersects exactly one 0-component of $L_p^*$, say $D^*_1$, and exactly one non 0-component of $L_p^*$, say $D^*_2$. That is, $F = (F \cap D^*_1) \cup (F \cap D^*_2)$. This concludes the proof. \hfill $\square$

For any 0-connected component $D^*$ of $L_p^*$, let $E_{D^*} = \{F : F \subseteq D^*\}$. For any non 0-connected component $D^*$ of $L_p^*$, let $E_{D^*} = \{F : F \cap D^* \neq \emptyset\}$.

**Lemma 20.** $E(H) = \bigcup_{D^* \in L_p^*} E_{D^*}$.

**Proof.** The lemma follows from the definition of $E_{D^*}$ and Lemma \ref{lem:lem}. \hfill $\square$

For each connected component $D^*$ of $L_p^*$ and for each $i \in [k]_0$, let us define the coloring function $\Phi[D^*, i] : D^* \rightarrow [k]_0$ as follows. First, if $i = 0$, then define $\Phi[D^*, 0](D^*) = 0$, that is, for all $v \in D^*$, define $\Phi[D^*, 0] = 0$. Otherwise, if $i \in [k]$, then for each constituent component $D$ of $D^*$ such that $L(D) = \emptyset$, define $\Phi[D^*, i](D) = \lambda_0$, and for each constituent component $D$ of $D^*$ such that $L(D) \neq \emptyset$, define $\Phi[D^*, i](D) = i$.

We now prove that for any connected component $D^*$ of $L_p^*$, there exists $i \in [k]_0$ such that our above definition of $\Phi[D^*, i]$ precisely captures the way the hypothetical witnessing coloring $\Upsilon$ colors $D^*$.

**Lemma 21.** For any connected component $D^*$ of $L_p^*$, there exists $i \in [k]_0$ such that $\Upsilon|_{D^*} = \Phi[D^*, i]$. 23
Proof. Let \( D_1, \ldots, D_r \) be the constituent components of \( D^* \). First assume that \( r = 1 \), that is, there is only one constituent component of \( D^* \). From Lemma 10, since \( D^* = D_1 \), we have that \( \Upsilon(D^*) = i \) for some \( i \in [k] \). Hence, if \( L(D) = 0 \), then \( \Upsilon(D_1) = \Phi[D^*, i] \). Otherwise, let \( L(D) = \{ l_d \} \). In this case, from Lemma 15, either \( \Upsilon(D^*) = l_d \) or \( \Upsilon(D^*) = 0 \). Thus, either \( \Upsilon(D_1) = \Phi[D^*, 0] \) or \( \Upsilon(D_1) = \Phi[D^*, i] \) for any \( i \in [k] \).

Now, we need prove the claim for the case where \( r \geq 2 \). Then, by Lemma 18 for any constituent component \( D_i \) of \( D^* \), we have that \( L(D_i) \neq \emptyset \). For any \( i \in [r] \), let \( L(D_i) = \{ l_d \} \). From the construction of \( L_\mu^* \), for each \( i \in [r] \), we have that \( l_{d_i} \in [k] \), which in particular means that \( l_{d_i} \neq 0 \). From Lemma 15, either \( \Upsilon(D_i) = 0 \) or \( \Upsilon(D_i) = l_{d_i} \). We aim to prove that either \( \Upsilon(D_1) = \Phi[D^*, 0] \) or \( \Upsilon(D_1) = \Phi[D^*, i] \) for any \( i \in [k] \).

In other words, we next show that either all constituent components of \( D^* \) are colored 0 by \( \Upsilon \) or \( \Upsilon \) colors each constituent component with the color represented by the label of that constituent component. To this end, let \( D' \) be the collection of all constituent components of \( D^* \) such that for all \( D_i \in D' \), \( \Upsilon(D_i) = l_{d_i} \) and let \( D'' \) be the collection of all constituent components of \( D^* \) such that for all \( D_i \in D'' \), \( \Upsilon(D_i) = 0 \). We need to show that either \( D' = \emptyset \) or \( D'' = \emptyset \). Suppose not, that is, \( D' \neq \emptyset \) and \( D'' \neq \emptyset \). Then, there exist \( D_1 \in D' \) and \( D_2 \in D'' \). Since \( D^* \) is a connected component in \( L_\mu^* \), there exists \( F \in E(H) \) such that \( F \cap D_1 \neq \emptyset \) and \( F \cap D_2 \neq \emptyset \). Since \( \Upsilon(D_1) = l_{d_1} \neq 0 \), from Lemma 11, we have that \( F \in E_m \). Since \( \Upsilon(D_2) = 0 \), \( F \cap D_2 \neq \emptyset \) and \( F \in E_m \), we deduce that \( D \) contains a vertex from the set \( B \). Thus, by Lemma 17, we have that \( D_2 \) is a 0-component of \( L_\mu^* \). Since no constituent component of \( D^* \) (which contains at least two constituent components) can be a 0-component from the construction of \( L_\mu^* \), we have reached a contradiction.

6.6 Dynamic Programming

For the sake of clarity of presentation, for every hyperedge \( F \in E_{D^*} \), we denote \( h_F(\mu', l'_1, l'_2) = \bigvee_{i \in [k]} f_F(\Phi[D^*, i], F, \mu', l'_1, l'_2) \).

Moreover, we let \( D^*_1, \ldots, D^*_y \) denote the connected components of \( L_\mu^* \). For each \( i \in [y] \), denote \( E_{D^*_i} = \{ F_{1,i}, \ldots, F_{i,1} \} \) (recall that \( E_{D^*_i} \) was defined in Appendix 6.5). For all \( i \in [y] \), \( \mu' \in [\mu] \), \( l'_1 \in [l_1] \) and \( l'_2 \in [l_2] \), define \( \mathcal{H}[i, \mu', l'_1, l'_2] \) as follows.

\[
\mathcal{H}[i, \mu', l'_1, l'_2] = \bigvee_{(\mu'^{1,i}) \in [\mu], j \in [z_i]} \bigwedge_{(l'^{1,i}) \in [l_1], (l'^{2,i}) \in [l_2]} h_{F_{i,j}}(\mu'^{1,i}, l'^{1,i}, l'^{2,i}),
\]

where \( \mu' = \sum_{j \in [z_i]} \mu'^{1,i}, \sum_{j \in [z_i]} l'^{1,i} \leq l'_1, \sum_{j \in [z_i]} l'^{2,i} \leq l'_2 \), and each \( \mu'^{1,i}, l'^{1,i}, l'^{2,i} \) is a non-negative integer.

Lemma 22. \( \text{aHP}[\mu, l_1, l_2] = \bigvee_{(\mu'^{1}) \in [\mu], i \in [y]} \bigwedge_{(l'^{1}) \in [l_1], (l'^{2}) \in [l_2]} \mathcal{H}[i, \mu'^{1,i}, l'^{1,i}, l'^{2,i}], \)

where \( \mu = \sum_{j \in [z]} \mu'^{1,i}, \sum_{j \in [z]} l'^{1,i} \leq l_1, \sum_{j \in [z]} l'^{2,i} \leq l_2 \), and each \( \mu'^{1,i}, l'^{1,i}, l'^{2,i} \) is a non-negative integer.

Proof. By Lemma 20, we have the following equality.

\[
\text{aHP}[\mu, l_1, l_2] = \bigvee_{F \in E_{D^*}} \bigwedge_{(\mu'^{F}) \in [\mu], (l'^{1,F}) \in [l_1], (l'^{2,F}) \in [l_2]} f_F(\Upsilon[F], \mu'^{F}, l'^{1,F}, l'^{2,F}),
\]

where \( \mu = \sum_{F \in E(H)} \mu'^{F}, \sum_{F \in E(H)} l'^{1,F} \leq l_1, \sum_{F \in E(H)} l'^{2,F} \leq l_2 \), and for all \( F \in E(H) \), \( \mu'^{F}, l'^{1,F} \) and \( l'^{2,F} \) are non-negative integers.

From Lemma 21 we thus further have the following equality.
where $\mu = \sum_{F \in E(H)} \mu^F$, $\sum_{F \in E(H)} l_1^F \leq l_1$, $\sum_{F \in E(H)} l_2^F \leq l_2$, and for all $F \in F(H)$, $\mu^F$, $l_1^F$ and $l_2^F$ are non-negative integers.

Hence, by the definition of $H[i, \mu', l'_1, l'_2]$, we conclude that the equation in the statement of the lemma is correct.

\[ \text{Lemma 23.} \quad \text{Suppose that for all } i \in [y], F \in \mathcal{E}_{D'}, \mu' \leq \mu, l'_1 \leq l_1 \text{ and } l'_2 \leq l_2, \text{ it holds that } h_F(\mu', l'_1, l'_2) \text{ is computable in time } \tau. \quad \text{Then, for all } i \in [y], \mu' \leq \mu, l'_1 \leq l_1 \text{ and } l'_2 \leq l_2, \text{ it holds that } H[i, \mu', l'_1, l'_2] \text{ can be computed in time } O(\tau \cdot z_1 \cdot b^2 \cdot k_1^2 \cdot k_2^2). \]

\[ \text{Proof.} \quad \text{Arbitrarily choose some } i \in [y], \mu' \leq \mu, l'_1 \leq l_1 \text{ and } l'_2 \leq l_2. \text{ Under the given supposition, we would show that } H[i, \mu', l'_1, l'_2] \text{ can be computed in time } O(\tau \cdot z_1 \cdot b^2 \cdot k_1^2 \cdot k_2^2). \text{ To this end, for all } c \in [z_i], \mu' \leq \mu, l'_1 \leq l_1 \text{ and } l'_2 \leq l_2, \text{ define } H_c[i, \mu', l'_1, l'_2] \text{ as follows.} \]

\[ H_c[i, \mu', l'_1, l'_2] = \bigvee_{(\mu')_j \in [c], j \in [c]} \bigwedge_{l'_1 j \in [c]} h_{F_j}(\mu^j l'_1, l'_2), \]

where $\mu' = \sum_{j \in [c]} \mu^j$, $\sum_{j \in [c]} l'_1 \leq l'_1$, $\sum_{j \in [c]} l'_2 \leq l'_2$ and each $\mu^j, l'_1, l'_2$ is a non-negative integer.

Now, for all $\mu' \leq \mu, l'_1 \leq l_1$ and $l'_2 \leq l_2$, $H_z[i, \mu', l'_1, l'_2]$ can be computed in time $O(\tau \cdot z_1 \cdot b^2 \cdot k_1^2 \cdot k_2^2)$ using the following recurrences.

\[ H_1[i, \mu', l'_1, l'_2] = h_{F_1}(\mu' l'_1, l'_2), \]

For all $c \in \{2, \ldots, z_i\}, i \in [y], \mu' \leq \mu, l'_1 \leq l_1$ and $l'_2 \leq l_2$,

\[ H_c[i, \mu', l'_1, l'_2] = \bigvee_{\mu' = \mu'' + \mu^j \atop l'_1 = l'_1 + l'_1 j \atop l'_2 = l'_2 + l'_1 j} H_{c-1}[i, \mu', l'_1, l'_2] \land h_{F_1}(\mu'' l'_1, l'_2). \]

Observe that $H[i, \mu', l'_1, l'_2] = H_{z}[i, \mu', l'_1, l'_2]$. This concludes the proof.

\[ \text{Lemma 24.} \quad \text{Suppose that for all } i \in [y], \mu' \leq \mu, l'_1 \leq l_1 \text{ and } l'_2 \leq l_2, \text{ } H[i, \mu', l'_1, l'_2] \text{ can be computed in time } \psi. \quad \text{Then, } \text{aHP}[\mu, l_1, l_2] \text{ can be computed in time } O(\psi \cdot b^2 \cdot k_1^2 \cdot k_2^2). \]

\[ \text{Proof.} \quad \text{For all } i \in [y], \mu' \leq \mu, l'_1 \leq l_1 \text{ and } l'_2 \leq l_2, \text{ let us first define } \text{aHP}_{i}[\mu', l'_1, l'_2] \text{ as follows.} \]

\[ \text{aHP}_{i}[\mu', l'_1, l'_2] = \bigvee_{(\mu')_j \in [i], j \in [i]} \bigwedge_{l'_1 j \in [i]} \bigwedge_{l'_2 j \in [i]} H[j, \mu^j l'_1, l'_2], \]

where $\mu' = \sum_{j \in [c]} \mu^j$, $\sum_{j \in [c]} l'_1 \leq l'_1$, $\sum_{j \in [c]} l'_2 \leq l'_2$ and each $\mu^j, l'_1, l'_2$ is a non-negative integer.

Observe that $\text{aHP}[\mu, l_1, l_2] = \text{aHP}_{y}[\mu, l_1, l_2]$ from Lemma 22. Now, we can compute $\text{aHP}_{y}[\mu, l_1, l_2]$ using the following recurrences in time $O(\psi \cdot b^2 \cdot k_1^2 \cdot k_2^2)$. }
\[ \text{aHP}_i[\mu', l'_1, l'_2] = \mathcal{H}[1, \mu', l'_1, l'_2]. \]

For all \( i \in [2, \ldots, y] \), \( \mu' \leq \mu, l'_1 \leq l_1 \) and \( l'_2 \leq l_2 \),

\[ \text{aHP}_i[\mu', l'_1, l'_2] = \bigvee_{\mu' = \mu^1 + \mu^2, \ l'_1 \geq \frac{l'_1 + l'_2}{2}, \ l'_2 \geq \frac{l'_1 + l'_2}{2}} \text{aHP}_{i-1}[\mu^1, l'_1, l'_2] \wedge \mathcal{H}[i, \mu^2, l'_1, l'_2]. \]

This concludes the proof of the lemma.

Clearly, for all \( i \in [y] \), \( F \in E_{D^*_1} \), \( \mu' \leq \mu, l'_1 \leq l_1 \) and \( l'_2 \leq l_2 \), it holds that \( h_F(\mu', l'_1, l'_2) \) is computable in polynomial time. Thus, by Lemmas 23 and 24 we conclude the proof of Theorem 5.

**References**

[1] Noga Alon, Béla Bollobás, Michael Krivelevich, and Benny Sudakov. Maximum cuts and judicious partitions in graphs without short cycles. *Journal of Combinatorial Theory, Series B*, 88(2):329–346, 2003.

[2] Florian Barbero, Gregory Gutin, Mark Jones, and Bin Sheng. Parameterized and approximation algorithms for the load coloring problem. *Algorithmica*, pages 1–19, 2015.

[3] Béla Bollobás and Alex D Scott. Judicious partitions of graphs. *Periodica Mathematica Hungarica*, 26(2):125–137, 1993.

[4] Béla Bollobás and Alex D Scott. Judicious partitions of hypergraphs. *Journal of Combinatorial Theory, Series A*, 78(1):15–31, 1997.

[5] Béla Bollobás and Alex D Scott. Exact bounds for judicious partitions of graphs. *Combinatorica*, 19(4):473–486, 1999.

[6] Béla Bollobás and Alex D Scott. Judicious partitions of 3-uniform hypergraphs. *European Journal of Combinatorics*, 21(3):289–300, 2000.

[7] Béla Bollobás and Alex D Scott. Problems and results on judicious partitions. *Random Structures & Algorithms*, 21(3-4):414–430, 2002.

[8] Béla Bollobás and Alex D Scott. Judicious partitions of bounded-degree graphs. *Journal of Graph Theory*, 46(2):131–143, 2004.

[9] Béla Bollobás and Alex D Scott. Max \( k \)-cut and judicious \( k \)-partitions. *Discrete Mathematics*, 310(15):2126–2139, 2010.

[10] Edouard Bonnet, Bruno Escoffier, Vangelis Th. Paschos, and Emeric Tourniaire. Multi-parameter analysis for local graph partitioning problems: Using greediness for parameterization. *Algorithmica*, 71(3):566–580, 2015.

[11] Leizhen Cai. Parameterized complexity of cardinality constrained optimization problems. *Comput. J.*, 51(1):102–121, 2008.

[12] Jianer Chen, Yang Liu, and Songjian Lu. An improved parameterized algorithm for the minimum node multiway cut problem. *Algorithmica*, 55(1):1–13, 2009.

[13] Rajesh Chitnis, Marek Cygan, MohammadTaghi Hajiaghayi, Marcin Pilipczuk, and Michal Pilipczuk. Designing FPT algorithms for cut problems using randomized contractions. *SIAM J. Comput.*, 45(4):1171–1229, 2016.
[14] Rajesh Hemant Chitnis, Marek Cygan, MohammadTaghi Hajiaghayi, Marcin Pilipczuk, and Michal Pilipczuk. Designing FPT algorithms for cut problems using randomized contractions. In 53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, New Brunswick, NJ, USA, October 20-23, 2012, pages 460–469. IEEE Computer Society, 2012.

[15] Robert Crowston, Gregory Gutin, Mark Jones, and Gabriele Muciaccia. Maximum balanced subgraph problem parameterized above lower bound. Theor. Comput. Sci., 513:53–64, 2013.

[16] Robert Crowston, Mark Jones, and Matthias Mnich. Max-cut parameterized above the edwards-erdős bound. Algorithmica, 72(3):734–757, 2015.

[17] Marek Cygan, Daniel Lokshtanov, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. Minimum bisection is fixed parameter tractable. In Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014, pages 323–332, 2014.

[18] Christopher S Edwards. Some extremal properties of bipartite subgraphs. Canad. J. Math, 25(3):475–483, 1973.

[19] Christopher S Edwards. An improved lower bound for the number of edges in a largest bipartite subgraph. In Proc. 2nd Czechoslovak Symposium on Graph Theory, Prague, pages 167–181, 1975.

[20] Michael Etscheid and Matthias Mnich. Linear kernels and linear-time algorithms for finding large cuts. In 27th International Symposium on Algorithms and Computation, ISAAC 2016, December 12-14, 2016, Sydney, Australia, pages 31:1–31:13, 2016.

[21] Gregory Gutin and Mark Jones. Parameterized algorithms for load coloring problem. Information Processing Letters, 114(8):446–449, 2014.

[22] Gregory Gutin and Anders Yeo. Note on maximal bisection above tight lower bound. Information Processing Letters, 110(21):966–969, 2010.

[23] Gregory Gutin and Anders Yeo. Constraint satisfaction problems parameterized above or below tight bounds: A survey. In The Multivariate Algorithmic Revolution and Beyond - Essays Dedicated to Michael R. Fellows on the Occasion of His 60th Birthday, volume 7370 of Lecture Notes in Computer Science, pages 257–286. Springer, 2012.

[24] John Haslegrave. Judicious partitions of uniform hypergraphs. Combinatorica, 34(5):561–572, 2014.

[25] Ken-ichi Kawarabayashi and Mikkel Thorup. The minimum $k$-way cut of bounded size is fixed-parameter tractable. In IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011, Palm Springs, CA, USA, October 22-25, 2011, pages 160–169, 2011.

[26] Choongbum Lee, Po-Shen Loh, and Benny Sudakov. Judicious partitions of directed graphs. Random Structures & Algorithms, 48(1):147–170, 2016.

[27] Daniel Lokshtanov, N. S. Narayanaswamy, Venkatesh Raman, M. S. Ramanujan, and Saket Saurabh. Faster parameterized algorithms using linear programming. ACM Trans. Algorithms, 11(2):15:1–15:31, October 2014.

[28] Jie Ma and Xingxing Yu. On judicious bipartitions of graphs. Combinatorica, 36(5):537–556, 2016.

[29] Meena Mahajan and Venkatesh Raman. Parameterizing above guaranteed values: Maxsat and maxcut. Journal of Algorithms, 31(2):335–354, 1999.

[30] Meena Mahajan, Venkatesh Raman, and Somnath Sikdar. Parameterizing above or below guaranteed values. J. Comput. Syst. Sci., 75(2):137–153, 2009.

[31] Dániel Marx. Parameterized graph separation problems. Theor. Comput. Sci., 351(3):394–406, 2006.
[32] Dániel Marx and Igor Razgon. Fixed-parameter tractability of multicut parameterized by the size of the cutset. *SIAM J. Comput.*, 43(2):355–388, 2014.

[33] Matthias Mnich and Rico Zenklusen. Bisections above tight lower bounds. In *Graph-Theoretic Concepts in Computer Science - 38th International Workshop, WG 2012, Jerusalem, Israel, June 26-28, 2012, Revised Selected Papers*, volume 7551 of *Lecture Notes in Computer Science*, pages 184–193. Springer, 2012.

[34] Moni Naor, Leonard J. Schulman, and Aravind Srinivasan. Splitters and near-optimal derandomization. In *36th Annual Symposium on Foundations of Computer Science, Milwaukee, Wisconsin, 23-25 October 1995*, pages 182–191. IEEE Computer Society, 1995.

[35] Saket Saurabh and Meirav Zehavi. $(k, n - k)$-max-cut: An $O^*(2^k)$-time algorithm and a polynomial kernel. In *LATIN 2016: Theoretical Informatics - 12th Latin American Symposium, Ensenada, Mexico, April 11-15, 2016, Proceedings*, pages 686–699, 2016.

[36] Alex D Scott. Judicious partitions and related problems. *Surveys in combinatorics*, 327:95-117, 2005.

[37] Hadas Shachnai and Meirav Zehavi. Representative families: A unified tradeoff-based approach. *J. Comput. Syst. Sci.*, 82(3):488–502, 2016.

[38] Baogang Xu, Juan Yan, and Xingxing Yu. Balanced judicious bipartitions of graphs. *Journal of Graph Theory*, 63(3):210–225, 2010.

[39] Baogang Xu and Xingxing Yu. Judicious $k$-partitions of graphs. *Journal of Combinatorial Theory, Series B*, 99(2):324–337, 2009.

[40] Baogang Xu and Xingxing Yu. On judicious bisections of graphs. *J. Comb. Theory, Ser. B*, 106:30–69, 2014.