Fixed-Parameter Algorithms for Fair Hitting Set Problems

Tanmay Inamdar
University of Bergen, Norway

Lawqueen Kanesh
Indian Institute of Technology, Jodhpur, India

Madhumita Kundu
University of Bergen, Norway

Nidhi Purohit
University of Bergen, Norway

Saket Saurabh
Institute of Mathematical Sciences, Chennai, India
University of Bergen, Norway

Abstract

Selection of a group of representatives satisfying certain fairness constraints, is a commonly occurring scenario. Motivated by this, we initiate a systematic algorithmic study of a *fair* version of Hitting Set. In the classical Hitting Set problem, the input is a universe $\mathcal{U}$, a family $\mathcal{F}$ of subsets of $\mathcal{U}$, and a non-negative integer k. The goal is to determine whether there exists a subset $S \subseteq \mathcal{U}$ of size k that *hits* (i.e., intersects) every set in $\mathcal{F}$. Inspired by several recent works, we formulate a fair version of this problem, as follows. The input additionally contains a family $\mathcal{B}$ of subsets of $\mathcal{U}$, where each subset in $\mathcal{B}$ can be thought of as the group of elements of the same *type*. We want to find a set $S \subseteq \mathcal{U}$ of size k that (i) hits all sets of $\mathcal{F}$, and (ii) does not contain *too many* elements of each type. We call this problem Fair Hitting Set, and chart out its tractability boundary from both classical as well as multivariate perspective. Our results use a multitude of techniques from parameterized complexity including classical to advanced tools, such as, methods of representative sets for matroids, FO model checking, and a generalization of best known kernels for Hitting Set.

2012 ACM Subject Classification Theory of computation → Fixed parameter tractability

Keywords and phrases Fairness, Parameterized Algorithms, Hitting Set

Digital Object Identifier 10.4230/LIPIcs.MFCS.2023.55

Related Version *Full Version*: https://arxiv.org/abs/2307.08854

Funding *Tanmay Inamdar*: Supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 819416).
Madhumita Kundu: Supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 819416).
Nidhi Purohit: Supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 819416).
Saket Saurabh: Supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 819416), and Swarnajayanti Fellowship (No. DST/SJF/MSA01/2017-18).

1 Introduction

Imagine a scenario of selecting a committee of size k from a group of people $\mathcal{U}$. We need a committee of people with some given attributes. These kinds of "attribute hitting" scenarios is modeled by a family $\mathcal{F}$ over $\mathcal{U}$, where for each attribute $\mathscr{A}$, we have a set $\mathcal{F}$ containing people in $\mathcal{U}$ who have the attribute $\mathscr{A}$. As is life, not always every set of people can work collectively. In particular, the committee cannot operate smoothly if we select more than the desired number of people from a set $B \subseteq \mathcal{U}$. These conflicts are modeled by another family, $\mathcal{B}$ over $\mathcal{U}$, and a function $f : \mathcal{B} \to \mathbb{N}$, which says that $f(B)$ is the maximum number of people from a set $B \in \mathcal{B}$ that can serve on the committee. Specifically, we want a committee that is a hitting set for attributes and has a set of people who are "conflict free". This paper aims to undertake a systematic study of a generalization of Hitting Set, which models such scenarios, and study this problem in the realm of parameterized complexity.

Indeed, Hitting Set is one of the 21 problems proven to be NP-complete by [6]. Recall, in this problem, we are given a set system $(\mathcal{U}, \mathcal{F})$, and an integer k. Here, $\mathcal{U}$ is a finite set of elements known as *universe* and $\mathcal{F}$ is a family of subsets of $\mathcal{U}$. The objective is to determine whether there exists a subset $S \subseteq \mathcal{U}$ such that S *hits* all sets in $\mathcal{F}$, i.e., for every $F_i \in \mathcal{F}$, $S \cap F_i \neq \emptyset$. Hitting Set is closely related to the Set Cover problem. These two problems, along with a particularly interesting special case thereof, namely that of Vertex Cover, are some of the most extensively studied problems in the field of approximation algorithms and parameterized complexity. Hitting Set problem is of a particular interest, because many combinatorial problems can be modeled as instances of Hitting Set.

Motivated from real-life applications, there has been a growing interest on the fairness aspect of various problems and algorithms developed. This has led to the whole new field of algorithmic fairness. Depending on the specific application, there are numerous ways to define the notion of *fairness*. One of the earliest definitions of fairness comes from [8], who defined fair versions of edge deletion problems. This was motivated from the following scenario. Suppose the graph models a communication network, with each edge being a link between a pair of nodes. In order to achieve acyclicity in the network, some links need to be disconnected. However, from the perspective of each node, it is desirable that fewest possible links incident to it are disconnected. Thus, we wish to disconnect links in a fair or equitable manner for the nodes.

Subsequently, this notion was extended by [11, 7] to define fair versions of vertex deletion problems. In this model, we want to delete a subset of vertices in order to achieve a certain graph property, such that each vertex has fewest possible neighbors deleted. As a concrete example, in a fair version of Vertex Cover in this model, we want to find a vertex cover S, such that each vertex outside S has fewest neighbors in S. Recently, [1] studied a generalization of this, called Sparse Hitting Set. The input to Sparse Hitting Set consists of $(\mathcal{U}, \mathcal{F}, \mathcal{B})$, where $\mathcal{U}$ is the universe, and $\mathcal{F}$ and $\mathcal{B}$ are two families of subsets of $\mathcal{U}$. The goal is to find a hitting set $S \subseteq \mathcal{U}$ for $\mathcal{F}$ such that $k \coloneqq \max_{B_i \in \mathcal{B}} |B_i \cap S|$ is minimized. Here, k is called the *sparseness* of the solution. Note that Sparse Hitting Set generalizes Fair Vertex Cover as defined above. Along a similar line, [5] considered *conflict-free* versions of various problems, including Hitting Set. In Conflict Free d-Hitting Set, we are given an instance $(\mathcal{U}, \mathcal{F}, k)$ of Hitting Set, and a conflict graph $H = (\mathcal{U}, E)$, and the goal is to find a hitting set $S \subseteq \mathcal{U}$ of size at most k, such that S induces an independent set in the conflict graph H.

Our Problem

Along the same line of work, we define a fair version of Hitting Set, which captures all of the aforementioned problems, and much more. Formally, the problem is defined as follows.

Fair Hitting Set
Input. An instance $\mathcal{I} = (\mathcal{U}, \mathcal{F}, \mathcal{B}, f : \mathcal{B} \to \mathbb{N}, k)$, where $\mathcal{U} = \{u_1, u_2, \ldots, u_n\}$ is the universe; $\mathcal{B}$ and $\mathcal{F}$ are two families of subsets of $\mathcal{U}$, where $\mathcal{F} = \{F_1, F_2, \ldots, F_m\}$, and $\mathcal{B} = \{B_1, B_2, \ldots, B_\ell\}$, and k is a positive integer.
Task. Determine whether there exists $S \subseteq \mathcal{U}$, with the following properties.
- $|S| \leq k$,
- S is a *hitting set* for $\mathcal{F}$, i.e., for every $F_i \in \mathcal{F}$, $S \cap F_i \neq \emptyset$, and
- For every $B_j \in \mathcal{B}$, $|S \cap B_j| \leq f(B_j)$.

We refer to a set $S \subseteq \mathcal{U}$ satisfying the above properties as a *fair hitting set* for $\mathcal{F}$, and use $|\mathcal{I}|$ to denote the size of the instance $\mathcal{I}$.

We note that Fair Hitting Set generalizes Sparse Hitting Set. Given an instance $(\mathcal{U}, \mathcal{F}, \mathcal{B})$ of Sparse Hitting Set, we iteratively solve instances $\mathcal{I}_i$ of Fair Hitting Set for $i = 1, 2, \ldots$. Here, an instance $\mathcal{I}_i$ of Fair Hitting Set is given by $(\mathcal{U}, \mathcal{F}, \mathcal{B}, f_i, |\mathcal{U}|)$, where $f_i(B_j) = i$ for all $B_j \in \mathcal{B}$. For the smallest i such that $\mathcal{I}_i$ is a yes-instance of Fair Hitting Set, we stop and conclude that i is the optimal sparseness of the given instance of Sparse Hitting Set. We note that Fair Hitting Set also generalizes the setting considered by [5].

1.1 Our Results, Techniques, and Relation to Hitting Set

First, we observe that Fair Hitting Set is a generalization of Hitting Set, by setting $\mathcal{B} = \emptyset$. Thus, Fair Hitting Set inherits all lower bound results from Hitting Set, namely, in general the problem is NP-hard as well as W[2]-hard parameterized by k, the solution size [2]. However, note that in the hard instances of Hitting Set, the sets in $\mathcal{F}$ can intersect arbitrarily. Indeed, consider an extreme case, when the sets in $\mathcal{F}$ are pairwise disjoint. In this setting Hitting Set is trivial to solve – an optimal solution must contain exactly one element from each set of $\mathcal{F}$. In contrast, we show that Fair Hitting Set remains NP-hard, as well as W[1]-hard w.r.t. k – and thus unlikely to be FPT – even in this simple setting. In particular, we show the following lower bound results, which are proved formally in the full version.

▶ **Theorem 1.** Fair Hitting Set *remains NP-hard when (1) the sets in $\mathcal{F}$ are pairwise disjoint, (2) each element appears in at most two distinct B_i's in $\mathcal{B}$, and (3) each $B_i \in \mathcal{B}$ has size exactly* 2. *Furthermore, assuming ETH, it is not possible to solve* Fair Hitting Set *in time $2^{o(t)}$, where $t = \max\{|\mathcal{U}|, |\mathcal{F}|, |\mathcal{B}|\}$..*

Fair Hitting Set *is W[1]-hard when parameterized by k, even when the sets in $\mathcal{F}$ are pairwise disjoint, and each $B_i \in \mathcal{B}$ has size exactly* 2.

The first result is obtained via a reduction from a problem of finding a "rainbow matching" on a path, and for the second result we give a parameter preserving reduction from k-Multicolored Independent Set. Given these lower bound results (Theorem 1), we study Fair Hitting Set under specific assumptions on the instance $\mathcal{I} = (\mathcal{U}, \mathcal{F}, \mathcal{B}, f, k)$. A natural question is: *under which assumptions?* To answer this we look at the known fixed-parameter tractability results for Hitting Set.

Hitting Set in Parameterized Complexity

Hitting Set is known to be W[2]-complete parameterized by the solution size in general. In other words, under widely believed complexity theoretic assumptions, it does not admit an FPT algorithm parameterized by the solution size. This motivates the study of Hitting Set in special cases. One particularly interesting case is Vertex Cover when the size of each set in $\mathcal{F}$ is exactly two. Vertex Cover is the most extensively studied problem in the parameterized complexity with a number of results in the FPT algorithms and kernelization in general graphs as well as special classes of graphs. Many of the techniques and results developed for Vertex Cover also extend d-Hitting Set, where each set in $\mathcal{F}$ has size at most d, for some constant d. More generally, Hitting Set is known to be FPT and admits a polynomial kernel in the case when the incidence graph $G_{\mathcal{U},\mathcal{F}}$, which is the bipartite graph on the vertex set $\mathcal{U} \uplus \mathcal{F}$ with edges denoting the set-containment, is $K_{i,j}$-free. That is, no i sets in $\mathcal{F}$ contain j elements in common, where i and j are assumed to be constants. This setting generalizes all the above settings as well as when the $G_{\mathcal{U},\mathcal{F}}$ is d-degenerate (since such graphs are $K_{d+1,d+1}$-free).

Our Algorithmic Results

Notably, we are able to extend almost all of the fixed-parameter tractability results for Hitting Set mentioned in the previous paragraph, under suitable assumptions on the set system $(\mathcal{U},\mathcal{B})$. We give a summary of our results in Figure 1.

More specifically, we obtain our results in the following steps. Consider a special case Fair Hitting Set, when the sets in $\mathcal{F}$ are pairwise disjoint, and each element appears in at most q sets in $\mathcal{B}$. Note that the first part of Theorem 1 implies that the problem is NP-hard even when $q = 2$. On the other hand, when $q = 1$, i.e., when both $\mathcal{F}$ and $\mathcal{B}$ are families of pairwise disjoint sets, then we observe Fair Hitting Set can be solved in polynomial time. Thus, $q = 1$ to 2 is a sharp transition between the tractability of the problem. Although the problem is NP-hard even for constant values of q, the following results are interesting in this setting. In particular, we show that the problem is FPT, and admits a polynomial kernel parameterized by k, if q is a constant.

▶ **Theorem 2.** *Let $(\mathcal{U},\mathcal{F},\mathcal{B},f : \mathcal{B} \to \mathbb{N},k)$ be an instance of* Fair Hitting Set. *Then,* Fair Hitting Set *can be solved in time $2^{\mathcal{O}(qk)}n^{\mathcal{O}(1)}$ time, when every element in $\mathcal{U}$ appears in at most q sets in $\mathcal{B}$ and any pair of sets in $\mathcal{F}$ are pairwise disjoint. Further,* Fair Hitting Set *admits a kernel of size $\mathcal{O}(kq^2\binom{kq}{q}\log k)$.*

Next we generalize Theorem 2 to a scenario where every element in $\mathcal{U}$ appears in at most q sets in $\mathcal{B}$ and at most d sets in $\mathcal{F}$.

▶ **Theorem 3.** *Let $(\mathcal{U},\mathcal{F},\mathcal{B},f : \mathcal{B} \to \mathbb{N},k)$ be an instance of* Fair Hitting Set. *Then,* Fair Hitting Set *can be solved in time $k^{\mathcal{O}(dk)}2^{\mathcal{O}(qk)}n^{\mathcal{O}(1)}$ time, when every element in $\mathcal{U}$ appears in at most q sets in $\mathcal{B}$ and at most d sets in $\mathcal{F}$.*

These results, Theorems 2 and 3, are obtained by the key observation that the problem can be modeled as finding a hitting set for $\mathcal{F}$ that is also an independent set in a suitably defined partition matroid that encodes the constraints imposed by $(\mathcal{U},\mathcal{B},f)$. This enables us to use the representative sets toolkit developed for matroids. This result is discussed in Section 3.

Next we consider a generalization of the above setting, where (1) each element appears in at most q sets in $\mathcal{B}$, and (2) the $G_{\mathcal{U},\mathcal{F}}$ is $K_{d,d}$-free. In this case, we combine the techniques developed in Hitting Set literature in the $K_{d,d}$-free setting, as well as, the representative sets based techniques developed in Section 3, to obtain the following result.

▶ **Theorem 4.** *Given an instance $\mathcal{I} = (\mathcal{U}, \mathcal{F}, \mathcal{B}, f, k)$ of* Fair Hitting Set*, such that $G_{\mathcal{U},\mathcal{F}}$ is $K_{d,d}$-free, and the frequency of each element in $\mathcal{B}$ is bounded by q, one can find an equivalent instance $\mathcal{I}' = (\mathcal{U}', \mathcal{F}', \mathcal{B}', f', k')$ of* Fair Hitting Set *in polynomial time, such that $|\mathcal{U}'| = \mathcal{O}(k^{d^2+q} d^d q^q)$, $|\mathcal{F}'| \le dk^d$, and $|\mathcal{B}'| = \mathcal{O}(k^{d^2+q} \cdot d^d q^{q+1})$, where d and q are assumed to be constants.*

Finally, we reach our most general case, where suppose (1) the $(\mathcal{U}, \mathcal{B})$ incidence graph is "nowhere dense" (defined formally in the full version; this class includes planar, excluded minor, bounded degree, and bounded expansion graphs), and (2) the $(\mathcal{U}, \mathcal{F})$ incidence graph is $K_{d,d}$-free. In this case, we obtain an FPT algorithm, parameterized by k and d. This result is in two steps. First, we proceed as prior to the case when each element appears in $f(k, d)$ sets of $\mathcal{F}$ (cf. Theorem 4). Next, since $(\mathcal{U}, \mathcal{B})$ incidence graph is nowhere dense, we reduce the problem of finding a Fair Hitting Set to FO model checking procedure on nowhere dense graphs, which is known to be FPT in the size of the formula. In particular, we show that the problem can be encoded by a variant of Induced Subgraph Isomorphism on nowhere dense graphs, where the size of the host graph we are searching for can be bounded by a function of k, d and the graph class.

▶ **Theorem 5.** *Let $\mathcal{G}$ be a nowhere dense graph class. Let $\mathcal{I} = (\mathcal{U}, \mathcal{F}, \mathcal{B}, f, k)$ be an instance of* Fair Hitting Set *such that the incidence graph $G \coloneqq G_{\mathcal{U},\mathcal{B}} \in \mathcal{G}$, and $G_{\mathcal{U},\mathcal{F}}$ is $K_{d,d}$-free for some $d \ge 1$. Then, one can solve* Fair Hitting Set *on $\mathcal{I}$ in time $h(k, d) \cdot |\mathcal{I}|^{\mathcal{O}(1)}$, for some function $h(\cdot, \cdot)$.*

2 Preliminaries

For an integer $\ell \ge 1$, we use the notation $[\ell] \coloneqq \{1, 2, \dots, \ell\}$. Let $\mathcal{R} = (\mathcal{U}, \mathcal{S})$ be a *set system*, where $\mathcal{U}$ is a finite set of elements (also called the *ground set* or the *universe*), and $\mathcal{S}$ is a family of subsets of $\mathcal{U}$. For an element $u \in \mathcal{U}$, and any $\mathcal{S}' \subseteq \mathcal{S}$, we use the notation $\mathcal{S}'(u) \coloneqq \{S \in \mathcal{S}' : u \in S\}$, i.e., $\mathcal{S}'(u)$ is the sub-family of sets from $\mathcal{S}'$ that contain u. For a subset $R \subseteq \mathcal{U}$, we denote $\mathcal{S} - R \coloneqq \{S \setminus R : S \in \mathcal{S}\}$. We use $G_{\mathcal{U},\mathcal{S}}$ to denote the incidence graph corresponding to the set system $(\mathcal{U}, \mathcal{S})$, i.e., $G_{\mathcal{U},\mathcal{S}}$ is a bipartite graph with bipartition $\mathcal{U} \uplus \mathcal{S}$, such that there is an edge between an element $e \in \mathcal{U}$ and a set $S \in \mathcal{S}$ iff $e \in S$.

In this paper, we work with finite, simple, undirected graphs. We use the standard graph theoretic notation and terminology, as defined in [3].

3 FPT Algorithm and Kernel Based on Representative Sets

In this section we design an algorithm and a kernel for a special case of Fair Hitting Set, using methods based on representative sets [4, 10]. Let $(\mathcal{U}, \mathcal{F}, \mathcal{B}, f : \mathcal{B} \to \mathbb{N}, k)$ be an instance of Fair Hitting Set. The first special case we consider is the following: every element in $\mathcal{U}$ appears in at most q sets in $\mathcal{B}$ and any pair of sets in $\mathcal{F}$ are pairwise disjoint.

Before this, however, we consider the special case of $q = 1$, i.e., when any pair of sets in $\mathcal{B}$, as well as that in $\mathcal{F}$ are disjoint. In this case, we can solve the problem in polynomial time, by reducing it to the problem of finding maximum flow in an auxiliary directed graph, defined as follows. The vertices of the graph are $\mathcal{B} \uplus \mathcal{U} \uplus \mathcal{F} \uplus \{s, t\}$. First, we add arcs (i.e., directed edges) from source s to each $B_j \in \mathcal{B}$, with capacity $f(B_j)$. Next, for every $u \in B_j$, we add an arc (B_j, u) of capacity 1. Similarly, for each $u \in F_i$, we add an arc (u, F_i), of capacity 1. Finally, we add arcs (F_i, t) of capacity ∞. It is straightforward to show that there exists a flow of value k in the graph iff there exists a fair hitting set of size k. We omit the details.

No.	$\mathcal{G}_{\mathcal{U},\mathcal{B}}$	$\mathcal{G}_{\mathcal{U},\mathcal{F}}$	Results
1	$q = 1$ ($B_i^{\prime}s$ are disjoint)	$d = 1$ ($F_i^{\prime}s$ are disjoint)	Polynomial time
2	$q = 2$	$d = 1$	NP-Hard No sub-exp algo(full version)
3	q	$d = 1$	$2^{\mathcal{O}(qk)} \cdot \lvert\mathcal{I}\rvert^{\mathcal{O}(1)}$(Theorem 2) $\mathcal{O}(kq^2\binom{kq}{q}\log k)$(kernel) (Theorem 12)
4	q	d	$k^{\mathcal{O}(dk)}2^{qk}\lvert\mathcal{I}\rvert^{\mathcal{O}(1)}$ (Theorem 3)
5	q	$K_{d,d}$-free	$k^{\mathcal{O}(d^2+q)}d^d q^{q+1}$(kernel) (Theorem 4)
6	apex-minor free	$K_{d,d}$-free	FPT/$(k+d)$ (full version)
7	nowhere dense	$K_{d,d}$-free	FPT/$(k+d)$ (Theorem 5)
8	$K_{2,2}$-free	$d = 1$	$W[1]$-Hard/k (full version)

Figure 1 An overview of different results obtained in this paper. In the second (resp. third) column, we state the assumption on the set system $(\mathcal{U}, \mathcal{B})$ (resp. $(\mathcal{U}, \mathcal{F})$). In rows 1-5 (resp. rows 1-4) q (resp. d) denotes the maximum frequency of an element in $\mathcal{B}$ (in $\mathcal{F}$). In the last column, we mention our results in the respective settings, and give corresponding references. Note that some of the references can be found in the full version.

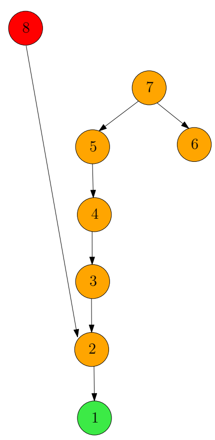

Figure 2 A Hasse diagram of the settings considered in Figure 1, where the number in each node corresponds to the row in the table. An arrow from node i to node j indicates that the setting in row i generalizes the setting in row j. Nodes colored in green, orange (resp. red) color indicate that the setting is solvable in polynomial, FPT time (resp. is $W[1]$-hard).

Note that $q \geq 2$, but the sets in $\mathcal{F}$ are pairwise disjoint, the problem is NP-hard. In this case, To design both our algorithm and the kernel we first embed the fairness constraints imposed by $\mathcal{B}$ in a combinatorial object called a *partition matroid*. A partition matroid is a set system $\mathcal{M} = (E, \mathcal{I})$, defined as follows. The ground set E is partitioned into ℓ subsets $E_1 \uplus E_2 \uplus \ldots \uplus E_\ell$, such that a set $S \subseteq E$ belongs to the family $\mathcal{I}$ iff for each $1 \leq j \leq \ell$, it holds that $|E_j \cap S| \leq k_j$, where $k_1, k_2, \ldots, k_\ell$ are non-negative integers.

It might be observed that the definition of a partition matroid closely resembles the fairness constraints, i.e., for each B_j, the hitting set H must satisfy $|H \cap B_i| \leq f(B_i)$. However, this idea does not quite work, since the sets $B_j \in \mathcal{B}$ are not disjoint – indeed, otherwise we could solve the problem in polynomial time, as discussed earlier. Nevertheless, we can salvage the situation by making q distinct copies of every element $u \in \mathcal{U}$, and replacing each of the occurrences of u in q distinct B_j's with a unique copy. The resulting set system is a partition matroid that exactly captures the fairness constraints. Correspondingly, in each set of $\mathcal{F}$, we replace an original element with all of its q copies. Recall that we want to find a hitting set for $\mathcal{F}$; however, in the new formulation, we must now ensure that if we pick *at least* one copy of element in the solution, we pick *all* of its copies in the solution.

Thus, our solution is an independent set of $\mathcal{M}$ that (1) is a hitting set for $\mathcal{F}$, and (2) picks either 0 or q copies of every element. To find such a solution in time FPT in k and q (resp. to reduce the size of the instance), we use a sophisticated tool developed in parameterized complexity, called *representative sets*. Later, we generalize this idea to the case where very element in $\mathcal{U}$ appears in at most q sets in $\mathcal{B}$ *and* at most d sets in $\mathcal{F}$. In the next section, we formally define the partition matroid, and in the subsequent sections, we apply the toolkit of representative sets to design our FPT algorithm and the kernel.

3.1 Partition Matroid and Our Solution

In a partition matroid we have a universe $\widetilde{U}$, partitioned into $\widetilde{U}_1, \cdots, \widetilde{U}_\ell$, together with positive integers $k_1, \cdots, k_\ell$, and a family of independent sets $\mathcal{I}$, such that $X \subseteq \widetilde{U}$ is in $\mathcal{I}$ if and only if $|X \cap \widetilde{U}_i| \leq k_i$, $i \in [\ell]$.

Let $(\mathcal{U}, \mathcal{B})$ be the given set system such that each element $u \in \mathcal{U}$ appears in *at most* q sets of $\mathcal{B}$. For an element $u \in \mathcal{U}$, let $q(u) \leq q$ denote the number of sets in $\mathcal{B}$, that u appears in. Further, for an element $u \in \mathcal{U}$, let $\mathsf{copies}(u) = \{u^1, u^2, \ldots, u^{q(u)}\}$. We define

$$\widetilde{U} = \bigcup_{u \in \mathcal{U}} \mathsf{copies}(u).$$

Next, we need to define a partition of $\widetilde{U}$. Towards this, we use the information about the sets in $\mathcal{B}$. We know that each element $u \in B$ appears in $q(u)$ sets and we have made $q(u)$ copies of u, thus we use *distinct* and *unique* copy of u in each sets in $\mathcal{B}$ in which u appears. This results in $\widetilde{\mathcal{B}} = \{\widetilde{B}_i : B_i \in \mathcal{B}\}$, where $\widetilde{B}_i$ is the set corresponding to an original set B_i, after replacing elements with their copies. Observe that for every pair of indices $i \neq j$ we have that $\widetilde{B}_i \cap \widetilde{B}_j = \emptyset$ and $\cup_i \widetilde{B}_i = \widetilde{U}$. This immediately gives a partition of $\widetilde{U}$. Finally, we define $k_i = f(B_i)$. This completes the description of the partition matroid we will be using. We will call this matroid as $\mathcal{M} = (\widetilde{U}, \mathcal{I})$

Given a subset $X \subseteq \widetilde{U}$, we define a set associated with X, called $\mathsf{projection}(X)$ as follows. The set $\mathsf{projection}(X) \subseteq U$, contains an element $u \in U$ if and only if $\mathsf{copies}(u) \cap X \neq \emptyset$. Similarly, we define a notion of embedding. For a set $A \subseteq \mathcal{U}$, let $\mathsf{embed}(A) = \cup_{u \in A} \mathsf{copies}(u)$. This brings us to the following lemma which relates our problem and finding an independent set in the matroid.

▶ **Lemma 6.** *An input $(\mathcal{U}, \mathcal{F}, \mathcal{B}, f : \mathcal{B} \to \mathbb{N}, k)$ is a yes-instance if and only if there exists an independent set $X \in \mathcal{I}$ of the matroid $\mathcal{M} = (\widetilde{U}, \mathcal{I})$ such that (1) $|\mathsf{projection}(X)| \leq k$, (2) $X = \mathsf{embed}(\mathsf{projection}(X))$, and (3) $\mathsf{projection}(X)$ is a hitting set for $\mathcal{F}$.*

Proof. In the forward direction, let $S \subseteq \mathcal{U}$ be a solution to the original problem. Suppose $S = \{s_1, s_2, \ldots, s_k\}$. Consider $X = \mathsf{embed}(S)$. Observe that by definition, $S = \mathsf{projection}(X)$. So $|\mathsf{projection}(X)| = k$ and $\mathsf{projection}(X)$ is a hitting set for $\mathcal{F}$. We claim that X is an independent set because if not, there exists a part $\widetilde{B}_i$ which satisfies that $|X \cap \widetilde{B}_i| > k_i = f(B_i)$. As $S = \mathsf{projection}(X)$ and for every $u \in \widetilde{B}_i$, B_i contains a v such that $u \in \mathsf{copies}(v)$, this results in $|S \cap B_i| > k_i$ implying $|S \cap B_i| > f(B_i)$ which contradicts that S is a valid solution to the original problem.

In the reverse direction, let $X \in \mathcal{I}$ be an independent set satisfying both the conditions. Let $S = \mathsf{projection}(X)$. We claim that, S is a solution for the original instance because if not, there exist a part B_i which satisfies that $|S \cap B_i| > f(B_i) = k_i$. As $X = \mathsf{embed}(S)$ and for every $u \in B_i$, an unique copy from $\mathsf{copies}(u)$ is contained in $\widetilde{B}_i$, which results in $|X \cap \widetilde{B}_i| > k_i$ which contradicts that X is an independent set. ◀

3.2 Computation of the Desired Independent Set

In this section we give an algorithm to compute an independent set $X \in \mathcal{I}$ of the matroid $\mathcal{M} = (\widetilde{U}, \mathcal{I})$ such that $|\mathsf{projection}(X)| \leq k$ and $\mathsf{projection}(X)$ is a hitting set for $\mathcal{F}$ (as given by Lemma 6). We will design a dynamic programming algorithm based on representative families to compute the desired independent set. Towards this we first give the required definitions. We start with the definition of an *ℓ-representative family.*

▶ **Definition 7** (ℓ-Representative Family)**.** *Given a matroid $M = (E, \mathcal{I})$ and a family $\mathcal{S}$ of subsets of E, we say that a subfamily $\widehat{\mathcal{S}} \subseteq \mathcal{S}$ is* ℓ-representative *for $\mathcal{S}$ if the following holds: for every set $Y \subseteq E$ of size at most ℓ, if there is a set $X \in \mathcal{S}$ disjoint from Y with $X \cup Y \in \mathcal{I}$, then there is a set $\widehat{X} \in \widehat{\mathcal{S}}$ disjoint from Y with $\widehat{X} \cup Y \in \mathcal{I}$. If $\hat{\mathcal{S}} \subseteq \mathcal{S}$ is ℓ-representative for $\mathcal{S}$ we write $\widehat{\mathcal{S}} \subseteq^{\ell}_{rep} \mathcal{S}$.*

In other words if some independent set in $\mathcal{S}$ can be extended to a larger independent set by adding ℓ new elements, then there is a set in $\widehat{\mathcal{S}}$ that can be extended by the same ℓ elements. We say that a family $\mathcal{S} = \{S_1, \ldots, S_t\}$ of sets is a *p-family* if each set in $\mathcal{S}$ is of size p.

▶ **Proposition 8** ([4, 9, Theorem 3.8, Theorem 1.3])**.** *Let $M = (E, \mathcal{I})$ be a partition matroid, $\mathcal{S} = \{S_1, \ldots, S_t\}$ be a p-family of independent sets. Then there exists $\widehat{\mathcal{S}} \subseteq^{\ell}_{rep} \mathcal{S}$ of size $\binom{p+\ell}{p}$. Furthermore, given a representation A_M of M over a field $\mathbb{F}$, there is a deterministic algorithm computing $\widehat{\mathcal{S}} \subseteq^{\ell}_{rep} \mathcal{S}$ of size at most $\binom{p+\ell}{p}$ in $\mathcal{O}\left(\binom{p+\ell}{p} tp^{\omega} + t\binom{p+\ell}{p}^{\omega-1} + ||A_M||^{\mathcal{O}(1)}\right)$ operations over $\mathbb{F}$, where $||A_M||$ denotes the length of A_M in the input.*

For the purpose of this article, it is enough to know that partition matroids are "representable" [10, Proposition 3.5] and a "truncation" of partition matroids are computable in deterministic polynomial time [9, Theorem 1.3]. This results in Proposition 8, which we will use for our algorithm without giving further definitions of representation and truncation [4, 9, 10].

Let $\mathcal{F} = \{F_1, F_2, \ldots, F_m\}$ be the subsets of $\mathcal{U}$, k be a positive integer. Since, the sets in $\mathcal{F}$ are pairwise disjoint, the number of sets in $\mathcal{F}$ is upper bounded by k. We call a set $S \subseteq U$, a *potential solution*, if for all $j \in [\ell]$, $|S \cap B_j| \leq f(B_j)$. Let

$$\mathcal{S}_i := \{S : S \text{ is a potential solution }, |S| = i \text{ and for all } j \in [i] |S \cap F_j| = 1\}.$$

Given $\mathcal{S}_i$, we define $\mathcal{S}_i^{\mathsf{emb}}$ as $\{\mathsf{embed}(S) \mid S \in \mathcal{S}_i\}$. Observe that $\mathcal{S}_i^{\mathsf{emb}} \subseteq \mathcal{I}$ and each set has size at most qi. Notice that, each set in $\mathcal{S}$ has size *exactly* i, but the same can not be said about the sets in $\mathcal{S}_i^{\mathsf{emb}}$. However, since each element occurs in at most q sets of $\mathcal{B}$, we have that each set in $\mathcal{S}_i^{\mathsf{emb}}$ has size at most qi.

Our algorithm checks whether $\mathcal{S}_k$ is non-empty or not. Towards that first observe that $\mathcal{S}_k$ is non-empty if and only if $\mathcal{S}_k^{\mathsf{emb}}$ is non-empty. So the testing of non-emptiness of $\mathcal{S}_k$ boils down to checking whether $\mathcal{S}_k^{\mathsf{emb}}$ is non-empty or not. We test whether $\mathcal{S}_k^{\mathsf{emb}}$ is non-empty by computing $\widehat{\mathcal{S}}_k^{\mathsf{emb}} \subseteq_{rep}^{0} \mathcal{S}_k^{\mathsf{emb}}$ and checking whether $\widehat{\mathcal{S}}_k^{\mathsf{emb}}$ is non-empty. To argue the correctness of the algorithm, first we have the following observation.

▶ **Observation 9.** *$\mathcal{S}_k^{\mathsf{emb}} \neq \emptyset$ iff $\widehat{\mathcal{S}}_k^{\mathsf{emb}} \neq \emptyset$.*

Proof. Since $\widehat{\mathcal{S}}_k^{\mathsf{emb}} \subseteq \mathcal{S}_k^{\mathsf{emb}}$, the reverse direction is immediate. Now we argue the forward direction. Suppose $\mathcal{S}_k^{\mathsf{emb}}$, then it contains some set A. Note that A trivially satisfies $A \cap \emptyset = \emptyset$. Therefore, since $\widehat{\mathcal{S}}_k^{\mathsf{emb}} \subseteq_{rep}^{0} \mathcal{S}_k^{\mathsf{emb}}$, there must exist a set $\hat{A} \in \widehat{\mathcal{S}}_k^{\mathsf{emb}}$ such that $\hat{A} \cap \emptyset = \emptyset$, i.e., $\widehat{\mathcal{S}}_k^{\mathsf{emb}} \neq \emptyset$. ◀

Thus, having computed the representative family $\widehat{\mathcal{S}}_k^{\mathsf{emb}}$ all we need to do is to check whether it is non-empty. All that remains is an algorithm that computes the representative family $\widehat{\mathcal{S}}_k^{\mathsf{emb}}$.

Let $\mathcal{Z}$ be a family of sets and ℓ be an integer, then $\mathcal{Z}[\ell]$ is a subset of $\mathcal{Z}$ that contains *all the sets of* $\mathcal{Z}$ of size *exactly* ℓ. We describe a dynamic programming based algorithm. Let $\mathcal{D}$ be an array indexed from integers in $\{0, 1, \ldots, k\}$. The entry $\mathcal{D}[i]$ stores the following for all $j \in \{i, \ldots, qi\}$, $\widehat{\mathcal{S}}_i^{\mathsf{emb}}[j] \subseteq_{rep}^{qk-j} \mathcal{S}_i^{\mathsf{emb}}[j]$.

We fill the entries in the matrix $\mathcal{D}$ in the increasing order of indexes. For $i = 0$, $\mathcal{D}[0] = \emptyset$. Suppose, we have filled all the entries until the index i. Then consider the set

$$\mathcal{N}^{i+1} = \{X' = X \cup \mathsf{embed}(\{u\}) : X \in \mathcal{D}[i],\ u \in F_{i+1}, \mathsf{projection}(X') \text{ is a potential solution}\}$$

We partition sets in $\mathcal{N}^{i+1}$ based on sizes. Let $\mathcal{N}^{i+1}[j]$ denote all the sets in $\mathcal{N}^{i+1}$ of size j.

▷ Claim 10. For all $j \in \{i+1, \ldots, q(i+1)\}$, $\mathcal{N}^{i+1}[j] \subseteq_{rep}^{qk-j} \mathcal{S}_{i+1}^{\mathsf{emb}}[j]$.

Proof. Let $S \in \mathcal{S}_{i+1}^{\mathsf{emb}}[j]$ and Y be a set of size at most $qk - j$ (which is essentially an independent set of the matroid $\mathcal{M} = (\widetilde{U}, \mathcal{I})$) such that $S \cap Y = \emptyset$ and $S \cup Y \in \mathcal{I}$. We will show that there exists a set $S' \in \mathcal{N}^{i+1}[j]$ such that $S' \cap Y = \emptyset$ and $S \cup Y \in \mathcal{I}$. This will imply the desired result. Since $S \in \mathcal{S}_{i+1}^{\mathsf{emb}}[j]$ there exists an element $u \in F_{i+1}$ such that

$$S = (S \setminus \mathsf{embed}(\{u\})) \cup \mathsf{embed}(\{u\}).$$

Let $S_i = (S \setminus \mathsf{embed}(\{u\}))$. Since, S is an independent set of the matroid $\mathcal{M} = (\widetilde{U}, \mathcal{I})$, we have that S_i is an independent set of the matroid $\mathcal{M} = (\widetilde{U}, \mathcal{I})$ (hereditary property). Further, $|\mathsf{projection}(S_i)| = i$ and $\mathsf{projection}(S_i)$ is a hitting set for $F_1, \ldots, F_i$. This implies that $S_i \in \mathcal{S}_i^{\mathsf{emb}}$. Let $Y_i = Y \cup \mathsf{embed}(\{u\})$. Notice that since $S \cap Y = \emptyset$ and $S \cup Y \in \mathcal{I}$, we have that Y_i is an independent set and $S_i \cup Y_i = S \cup Y \in \mathcal{I}$. Let $|S_i| = j'$. Then, we know that $\mathcal{D}[i]$ contains $\widehat{\mathcal{S}}_i^{\mathsf{emb}}[j'] \subseteq_{rep}^{qk-j'} \mathcal{S}_i^{\mathsf{emb}}[j']$. This implies that there exists a set $S_i' \in \widehat{\mathcal{S}}_i^{\mathsf{emb}}[j']$ such that $S_i' \cup Y_i \in \mathcal{I}$. This implies that $S_i' \cup \mathsf{embed}(\{u\})$ is in $\mathcal{N}^{i+1}$. Further, since $|S| = \sum_{x \in \mathsf{projection}(S)} |\mathsf{embed}(\{x\})|$, we have that $|S_i' \cup \mathsf{embed}(\{u\})| = |S_i| = j$. This implies that $S_i' \cup \mathsf{embed}(\{u\})$ is in $\mathcal{N}^{i+1}[j]$. Thus, we can take $S' = S_i' \cup \mathsf{embed}(\{u\})$. This completes the proof. ◁

We fill the entry for $\mathcal{D}[i+1]$ as follows. We first compute $\mathcal{N}^{i+1}$. Observe that the sets in $\mathcal{N}^{i+1}$ have sizes ranging from $i+1$ to $q(i+1)$. Now we apply Proposition 8 on each of $\mathcal{N}^{i+1}[j]$, $j \in \{i+1, \ldots, q(i+1)\}$, and compute $qk-j$ representative. That is, we compute $\widehat{\mathcal{N}}^{i+1}[j] \subseteq^{qk-j}_{rep} \mathcal{N}^{i+1}[j]$. We set

$$\mathcal{D}[i+1] = \bigcup_{j=i+1}^{q(i+1)} \widehat{\mathcal{N}}^{i+1}[j].$$

Observe that the number of sets in $\mathcal{D}[i]$ of size j is upper bounded by $\binom{q(k-i)+j}{j} \leq \binom{qk}{di} \leq 2^{\mathcal{O}(qk)}$. Hence, the time taken to compute $\mathcal{D}[i]$ is upper bounded by $2^{\mathcal{O}(qk)}n^{\mathcal{O}(1)}$. Thus, the time taken to compute $\mathcal{D}[i+1]$ requires at most qk invocations of Proposition 8. This itself takes $2^{\mathcal{O}(qk)}n^{\mathcal{O}(1)}$ time. This completes the proof, resulting in the following result.

▶ **Theorem 2.** *Let $(\mathcal{U}, \mathcal{F}, \mathcal{B}, f : \mathcal{B} \to \mathbb{N}, k)$ be an instance of* FAIR HITTING SET*. Then,* FAIR HITTING SET *can be solved in time $2^{\mathcal{O}(qk)}n^{\mathcal{O}(1)}$ time, when every element in $\mathcal{U}$ appears in at most q sets in $\mathcal{B}$ and any pair of sets in $\mathcal{F}$ are pairwise disjoint. Further,* FAIR HITTING SET *admits a kernel of size $\mathcal{O}(kq^2\binom{kq}{q}\log k)$.*

Theorem 2 can be generalized to the scenario where every element in $\mathcal{U}$ appears in at most q sets in $\mathcal{B}$ and at most d sets in $\mathcal{F}$. Observe that if each element appear in at most d sets of $\mathcal{F}$, then the total number of sets that a subset of size k of $\mathcal{U}$ can hit is upper bounded by dk, else we immediately return that given instance is a NO-instance. Let $S = \{u_1, \ldots, u_k\}$ be a hypothetical solution to our problem. Now, with the help of S, we partition $\mathcal{F}$ as follows. Let $\mathcal{F}_i$ denote all sets in $\mathcal{F}$ that contain u_i and none of $\{u_1, \ldots, u_{i-1}\}$. Clearly, $\mathcal{F}_i$, $i \in [k]$, partitions $\mathcal{F}$. Now we can design a dynamic programming algorithm similar to the one employed in Theorem 2, where in each iteration we grow our representative family by elements that only hit sets in $\mathcal{F}_i$ and not in $\mathcal{F}_j$, $j > i$. This will result in the following theorem, whose proof can be found in the full version.

▶ **Theorem 3.** *Let $(\mathcal{U}, \mathcal{F}, \mathcal{B}, f : \mathcal{B} \to \mathbb{N}, k)$ be an instance of* FAIR HITTING SET*. Then,* FAIR HITTING SET *can be solved in time $k^{\mathcal{O}(dk)}2^{\mathcal{O}(qk)}n^{\mathcal{O}(1)}$ time, when every element in $\mathcal{U}$ appears in at most q sets in $\mathcal{B}$ and at most d sets in $\mathcal{F}$.*

3.3 A Kernel for a Special Case of Fair Hitting Set using Matroids

In this section we design a polynomial kernel for the same special case of FAIR HITTING SET, that we considered in the last section. Let $(\mathcal{U}, \mathcal{F}, \mathcal{B}, f : \mathcal{B} \to \mathbb{N}, k)$ be an instance of FAIR HITTING SET and assume that every element in $\mathcal{U}$ appears in at most q sets in $\mathcal{B}$ and any pair of sets in $\mathcal{F}$ are pairwise disjoint. To design our kernel we will again use Lemma 6 that says that an input $(\mathcal{U}, \mathcal{F}, \mathcal{B}, f : \mathcal{B} \to \mathbb{N}, k)$ is a yes-instance if and only if there exists an independent set $X \in \mathcal{I}$ of the matroid $\mathcal{M} = (\widetilde{U}, \mathcal{I})$ such that $|\mathsf{projection}(X)| \leq k$ and $\mathsf{projection}(X)$ is a hitting set for $\mathcal{F}$.

Let $\mathcal{F} = \{F_1, F_2, \ldots, F_m\}$ be the subsets of $\mathcal{U}$, and k be a positive integer. Since, the sets in $\mathcal{F}$ are pairwise disjoint, the number of sets in $\mathcal{F}$ is upper bounded by k. In particular, we assume that $m = k$. We define $\mathcal{F}_i^{\mathsf{emb}} = \{\mathsf{embed}(\{u\}) \mid u \in F_i\}$. For our kernel we apply the following reduction rules. We start with some simple reduction rules.

▶ **Reduction Rule 1.** Let $(\mathcal{U}, \mathcal{F}, \mathcal{B}, f : \mathcal{B} \to \mathbb{N}, k)$ be an instance of FAIR HITTING SET.

- If there exists an element $u \in \mathcal{U}$, such that u does not appear in any sets in $\mathcal{F}$ then delete it from $\mathcal{U}$ and all the sets in $\mathcal{B}$ that it appears in.
- If there exists a set $B \in \mathcal{B}$ such that $B = \emptyset$, then delete B, and take f as the restriction of f on $\mathcal{B} \setminus \{B\}$.

- If there exists a set $B \in \mathcal{B}$ such that $f(B) = 0$, then we do as follows: $\mathcal{U} := \mathcal{U} \setminus \{B\}$; delete all the elements of B from all the sets in $\mathcal{B}$ and $\mathcal{F}$ that it appears in. If some set in $\mathcal{F}$ becomes empty then return a trivial No-instance. Else, take f as the restriction of f on $\mathcal{B} \setminus \{B\}$ and keep the integer k unchanged.

Soundness of Reduction Rule 1 is obvious and hence omitted. The next reduction rule is the main engine of our kernel.

▶ **Reduction Rule 2.** Let $(\mathcal{U}, \mathcal{F}, \mathcal{B}, f : \mathcal{B} \to \mathbb{N}, k)$ be an instance of Fair Hitting Set. If there exists a pair of integers $i \in [k]$ and $j \in [q]$ such that $|\mathcal{F}_i^{\mathsf{emb}}[j]| > \binom{kq}{j}$, then do as follows. Compute $\widehat{\mathcal{F}}_i^{\mathsf{emb}}[j] \subseteq_{rep}^{qk-j} \mathcal{F}_i^{\mathsf{emb}}[j]$. Let $F \in \mathcal{F}_i^{\mathsf{emb}}[j]$ that do not appear in $\widehat{\mathcal{F}}_i^{\mathsf{emb}}[j]$. Then obtain a reduced instance as follows.
- $\mathcal{U} := \mathcal{U} \setminus \mathsf{projection}(F)$
- Delete $\mathsf{projection}(F)$ from all the sets in $\mathcal{B}$ and $\mathcal{F}$ that it appears in.
- The function f and k remains the same.

▶ **Lemma 11.** *Reduction Rule 2 is sound.*

Proof. Let $(\mathcal{U}, \mathcal{F}, \mathcal{B}, f : \mathcal{B} \to \mathbb{N}, k)$ be an instance of Fair Hitting Set, and let $(\mathcal{U}', \mathcal{F}', \mathcal{B}', f : \mathcal{B} \to \mathbb{N}, k)$ be the reduced instance, after an application of Reduction Rule 2. It is easy to see that a solution to the reduced instance can directly be lifted to the input instance. Thus, we focus on forward direction.

In the forward direction, let S be a solution to $(\mathcal{U}, \mathcal{F}, \mathcal{B}, f : \mathcal{B} \to \mathbb{N}, k)$. Then, by Lemma 6, it implies that $\mathsf{embed}(S) \in \mathcal{I}$ (of the matroid $\mathcal{M} = (\widetilde{U}, \mathcal{I})$) and $|S| \leq k$ and $S = \mathsf{projection}(\mathsf{embed}(S))$ is a hitting set for $\mathcal{F}$. Let $u = \mathsf{projection}(F)$. Then, we have that $\mathcal{U} := \mathcal{U} \setminus \{u\}$. If $u \notin S$, then S is also the solution to $(\mathcal{U}', \mathcal{F}', \mathcal{B}', f : \mathcal{B}' \to \mathbb{N}, k)$. So we assume that $u \in S$.

Observe that $F = \mathsf{embed}(\{u\})$, $|F| = j$, and u belongs to F_i. Further, since every set in $\mathcal{F}$ are pairwise disjoint we have that the only job of u is to hit the set F_i. Consider, $Y = \mathsf{embed}(S) \setminus \mathsf{embed}(\{u\})$. Since, $\mathsf{embed}(S) \in \mathcal{I}$, we have that $Y \in \mathcal{I}$ (hereditary property of the matroid), and the size of Y is upper bounded by $qk - j$. The last assertion follows from the fact that for any element $v \in \mathcal{U}$, the size of $\mathsf{embed}(\{v\})$ is upper bounded by q and $|\mathsf{embed}(S)| = \sum_{x \in S} |\mathsf{embed}(\{x\})| \leq qk$. This implies that there exists $F' \in \widehat{\mathcal{F}}_i^{\mathsf{emb}}[j] \subseteq_{rep}^{qk-j} \mathcal{F}_i^{\mathsf{emb}}[j]$ such that $Y \cup \{F'\} \in \mathcal{I}$. Since, $|\mathsf{projection}(\mathsf{embed}(S))| \leq k$, we have that $|\mathsf{projection}(Y)| \leq k - 1$. Thus, $|\mathsf{projection}(Y \cup \{F'\})| \leq k$. Now we need to show that $\mathsf{projection}(Y \cup \{F'\})$ is a hitting set for $\mathcal{F}$. This follows from the fact that $u' = \mathsf{projection}(F') \in F_i$. In other words, we have shown that $S' = S \setminus \{u\} \cup \{u'\}$ is a desired hitting set for $\mathcal{F}$. This concludes the proof. ◀

Finally, we get the following kernel.

▶ **Theorem 12.** *Let $(\mathcal{U}, \mathcal{F}, \mathcal{B}, f : \mathcal{B} \to \mathbb{N}, k)$ be an instance of* Fair Hitting Set *such that every element in $\mathcal{U}$ appears in at most q sets in $\mathcal{B}$ and any pair of sets in $\mathcal{F}$ are pairwise disjoint. Then,* Fair Hitting Set *admits a kernel of size $\mathcal{O}(kq^2\binom{kq}{q}\log k)$.*

Proof. For our algorithm we apply Reduction Rules 1 and 2 exhaustively. If any application of these rules return that the input is a No-instance, we return the same. The correctness of the algorithm follows from the correctness of Reduction Rules 1 and 2. Further it is clear that the algorithm runs in polynomial time. What remains to show is that the reduced instance is upper bounded by the claimed function.

For convenience we assume that the reduced instance is also denoted by $(\mathcal{U}, \mathcal{F}, \mathcal{B}, f : \mathcal{B} \to \mathbb{N}, k)$. Since, Reduction Rule 2 is not applicable we have that each set F_i, $i \in [k]$, is upper bounded by $\sum_{j=1}^{q} \binom{kq}{j} \leq q\binom{kq}{q}$. This implies that $|\mathcal{U}| \leq kq\binom{kq}{q}$. Further, since every element in $\mathcal{U}$ appears in at most q sets in $\mathcal{B}$, we have that the number of non-empty sets in $\mathcal{B}$ is upper bounded by $q|\mathcal{U}| \leq kq^2\binom{kq}{q}$. Since, Reduction Rule 1 is not applicable we have that there are no empty-sets and hence $|\mathcal{B}| \leq kq^2\binom{kq}{q}$. Further, to represent the function f we need at most $\mathcal{O}(|\mathcal{B}| \log k) \leq \mathcal{O}(kq^2\binom{kq}{q} \log k)$ bits. This completes the proof. ◀

4 Reduction from $K_{d,d}$-free $G_{\mathcal{U},\mathcal{F}}$ to Bounded Frequency in $\mathcal{F}$

In Section 4.1, we consider a special case of the above setting: (1) $G_{\mathcal{U},\mathcal{F}}$ is $K_{d,d}$-free, and (2) each element in $\mathcal{U}$ has frequency at most q in $\mathcal{B}$. For this case, we design a polynomial kernel. For the sake of brevity, we give a detailed, yet informal, overview of the kernelization, and defer the formal details to the full version.

A part of the kernelization procedure can also be used to bound the frequency of an element in $\mathcal{F}$ by a function of k and d. We give an alternate, self-contained proof of this theorem in the full version. This reduction is used as the first step in some of our results, such as Section 5.

4.1 Polynomial Kernel for $K_{d,d}$-free $G_{\mathcal{U},\mathcal{F}}$ and Bounded Frequency in $\mathcal{B}$

Consider an input $(\mathcal{U}, \mathcal{F}, \mathcal{B}, f : \mathcal{B} \to \mathbb{N}, k)$ of Fair Hitting Set. In this section we design a polynomial kernel for Fair Hitting Set problem when $G_{\mathcal{U},\mathcal{F}}$ is $K_{d,d}$-free and frequency of each element in $\mathcal{B}$ is at most q. We fix d and q for the rest of the section. Without loss of generality we assume that $d \geq 2$, $k \geq 2$. We also assume that we do not have multisets in $\mathcal{F}$ and $\mathcal{B}$.

Under these assumptions, the kernelization algorithm consists of two phases. In the first phase we apply some reduction rules to bound the size of $|\mathcal{F}|$. In the second phase, we use the partition matroid $\mathcal{M} = (\widetilde{U}, \mathcal{I})$, as defined in Section 3.1 using $\mathcal{B}$ to design a reduction rule to bound the number of elements. Now we discuss each of these phases in more detail.

Phase 1

We first define the following easy reduction rules that handle some of the easy cases

- We can delete an empty set from $\mathcal{B}$ without affecting the instance.
- We can delete an element $u \in \mathcal{U}$ that is not contained in any set in $\mathcal{F}$.
- If there exists a set $B \in \mathcal{B}$ with $f(B) = 0$, then we consider two cases.
  1. If there exists some $F \in \mathcal{F}$ such that $F \subseteq B$, then we have a no-instance.
  2. Otherwise, we can delete B from $\mathcal{B}$, and delete all elements of B from the universe $\mathcal{U}$ as well as the corresponding sets in $\mathcal{F}$.

Each such rule can be implemented in polynomial time. We emphasize that these reduction rules are repeatedly applied in this order, after each application of subsequent rules.

Next, we consider the following case. If there exists an element $u \in \mathcal{U}$ contained in at least dk^{d-1} sets of $\mathcal{F}$, then in *polynomial time* we can find a non-empty set $X \subseteq \mathcal{U}$ of size at most $d-1$ with the following properties: (1) X intersects with *every* fair hitting set of size at most k, and (2) the number of sets in $\mathcal{F}$ that contain X as a subset is *large*, i.e., at least dk^{d-1}. This is where we crucially use the fact that $G_{\mathcal{U},\mathcal{F}}$ is $K_{d,d}$-free. Now, we can use such a set X to reduce the size of instance, by either finding some $u \in \mathcal{U}$, or some $F_i \in \mathcal{F}$ that can be deleted without affecting the instance.

Note that since we reduce the size of instance in each application of the rule, this rule is applicable only polynomially many times. Furthermore, when the rule is not applicable, it follows that every element of $\mathcal{U}$ is contained in at most dk^{d-1} sets of $\mathcal{F}$. Here, we observe that we did not make any specific assumptions about the hypergraph $(\mathcal{U}, \mathcal{B})$, except that we may delete an element of $\mathcal{U}$ or a set from $\mathcal{B}$. Thus, the resulting hypergraph is a sub-hypergraph of the original hypergraph $(\mathcal{U}, \mathcal{B})$. Thus, if $G_{\mathcal{U},\mathcal{B}}$ satisfies some hereditary property Π, then the resulting incidence graph continues to satisfy Π. Thus, phase 1 constitutes a proof of .

At this step, if the number of sets in $\mathcal{F}$ is larger than dk^d, no subset of size k can hit all sets in $\mathcal{F}$. Thus, we simply conclude that such an instance is a no-instance of the problem. This concludes Phase 1.

Suppose in the original instance $G_{\mathcal{U},\mathcal{B}}$ satisfied that the frequency of every element in $\mathcal{B}$ is at most q. Then, at this step, we can use an approach similar to Theorem 3 to design an FPT algorithm that runs in time $d^k k^{kd} \cdot 2^{\mathcal{O}(kq)} \cdot |\mathcal{I}|^{\mathcal{O}(1)}$. Now, we proceed to Phase 2, where we further reduce the size of an instance to design a kernel under the assumption when the frequency in $\mathcal{B}$ is bounded by q.

Phase 2

We observe that the number of elements in $\mathcal{U}$ with frequency at least d in $\mathcal{F}$ is bounded by $\binom{|\mathcal{F}|}{d} \cdot d$. Let $\mathcal{U}'$ denote the elements that are contained in at most $d-1$ sets of $\mathcal{F}$. We define equivalence classes of $\mathcal{U}'$, such that all elements in the same class belong to all the sets of $\mathcal{Y}$, where $\mathcal{Y} \subseteq \mathcal{F}$ is a sub-family of size at most $d-1$. Let us denote such a subset by $\mathsf{ExactNbr}(\mathcal{Y})$.

We use the matroid-based techniques developed in the previous section, in order to reduce the number of distinct elements in $\mathsf{ExactNbr}(\mathcal{Y})$ that we need to remember. In particular, we show that for every $\mathcal{Y} \subseteq \mathcal{F}$, we only need to remember at most $\binom{kq}{q}$ distinct elements, and we may delete the rest in a careful manner.

Thus, at the end, we have the following. The number of elements with degree (i.e., frequency) at most $d-1$ in $G_{\mathcal{U},\mathcal{F}}$ is bounded by $d \cdot \binom{|\mathcal{F}|}{d} \cdot \binom{kq}{q}$. Accounting for the elements with large degree, the total number of elements in $\mathcal{U}$ is bounded by $\mathcal{O}(k^{\mathcal{O}(d^2+q)} d^d q^q)$. Since each element has degree at most q in $G_{\mathcal{U},\mathcal{B}}$, we can also bound the number of sets in $\mathcal{B}$. Observe that each of our reduction rules can be applied in polynomial time and only polynomially many times. Thus, we prove the following theorem.

▶ **Theorem 4.** *Given an instance $\mathcal{I} = (\mathcal{U}, \mathcal{F}, \mathcal{B}, f, k)$ of* Fair Hitting Set*, such that $G_{\mathcal{U},\mathcal{F}}$ is $K_{d,d}$-free, and the frequency of each element in $\mathcal{B}$ is bounded by q, one can find an equivalent instance $\mathcal{I}' = (\mathcal{U}', \mathcal{F}', \mathcal{B}', f', k')$ of* Fair Hitting Set *in polynomial time, such that $|\mathcal{U}'| = \mathcal{O}(k^{d^2+q} d^d q^q)$, $|\mathcal{F}'| \le dk^d$, and $|\mathcal{B}'| = \mathcal{O}(k^{d^2+q} \cdot d^d q^{q+1})$, where d and q are assumed to be constants.*

5 Parameterization by $k + d$ when $G_{\mathcal{U},\mathcal{B}}$ is nowhere dense and $G_{\mathcal{U},\mathcal{F}}$ is $K_{d,d}$-free

We give a brief overview of the following theorem, a formal proof can be found in the full version.

▶ **Theorem 5.** *Let $\mathcal{G}$ be a nowhere dense graph class. Let $\mathcal{I} = (\mathcal{U}, \mathcal{F}, \mathcal{B}, f, k)$ be an instance of* Fair Hitting Set *such that the incidence graph $G := G_{\mathcal{U},\mathcal{B}} \in \mathcal{G}$, and $G_{\mathcal{U},\mathcal{F}}$ is $K_{d,d}$-free for some $d \ge 1$. Then, one can solve* Fair Hitting Set *on $\mathcal{I}$ in time $h(k, d) \cdot |\mathcal{I}|^{\mathcal{O}(1)}$, for some function $h(\cdot, \cdot)$.*

First, we reduce the given instance, where $G_{\mathcal{U},\mathcal{F}}$ is $K_{d,d}$-free, to an instance where the size of $|\mathcal{F}|$ is bounded by $d \cdot k^d$. Thus, it suffices to design an FPT algorithm parameterized by k, d and $m := |\mathcal{F}|$. The rest of the section focuses on designing such an algorithm. We prove Theorem 5 by reducing the problem to FO model checking on G. [1] Due to lack of space, the necessary definitions and background pertaining to nowhere dense graph classes and first-order logic is given in the full version.

We reduce the problem of deciding whether $\mathcal{I}$ is a yes-instance of FAIR HITTING SET to the problem of First-Order (FO) model checking. We first "guess" the structure of a hypothetical solution (i.e., $S \subseteq \mathcal{U}$ such that S is a fair hitting set for $\mathcal{F}$), if any. More specifically, we guess the exact size k' of a hypothetical solution, and the exact subset of $\mathcal{F}$ that is hit by each element of the solution. Note that there are at most $2^{\mathcal{O}(km)}$ possible guesses. For each such guess, we create a first-order logic formula that is true if and only if such a solution is a fair hitting set, i.e., it hits all the sets in F, and for each $B_j \in \mathcal{B}$, the size of the intersection of B_j with the solution is at most $f(B_j)$. We show that the size of the formula is upper bounded by a function of k and $|\mathcal{F}|$. Finally, since model checking of first-order logic formulas can be decided in polynomial time on nowhere dense graphs, the theorem follows.

References

1. Johannes Blum, Yann Disser, Andreas Emil Feldmann, Siddharth Gupta, and Anna Zych-Pawlewicz. On sparse hitting sets: from fair vertex cover to highway dimension. *CoRR*, abs/2208.14132, 2022. doi:10.48550/arXiv.2208.14132.
2. Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. doi:10.1007/978-3-319-21275-3.
3. Reinhard Diestel. *Graph Theory, 4th Edition*, volume 173 of *Graduate texts in mathematics*. Springer, 2012.
4. Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, and Saket Saurabh. Efficient computation of representative families with applications in parameterized and exact algorithms. *J. ACM*, 63(4):29:1–29:60, 2016. doi:10.1145/2886094.
5. Pallavi Jain, Lawqueen Kanesh, and Pranabendu Misra. Conflict free version of covering problems on graphs: Classical and parameterized. *Theory Comput. Syst.*, 64(6):1067–1093, 2020. doi:10.1007/s00224-019-09964-6.
6. Richard M Karp. Reducibility among combinatorial problems. In *Complexity of computer computations*, pages 85–103. Springer, 1972.
7. Dusan Knop, Tomás Masarík, and Tomás Toufar. Parameterized complexity of fair vertex evaluation problems. In Peter Rossmanith, Pinar Heggernes, and Joost-Pieter Katoen, editors, *44th International Symposium on Mathematical Foundations of Computer Science, MFCS 2019, August 26-30, 2019, Aachen, Germany*, volume 138 of *LIPIcs*, pages 33:1–33:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.MFCS.2019.33.
8. Li-Shin Lin and Sartaj Sahni. Fair edge deletion problems. *IEEE Trans. Computers*, 38(5):756–761, 1989. doi:10.1109/12.24280.
9. Daniel Lokshtanov, Pranabendu Misra, Fahad Panolan, and Saket Saurabh. Deterministic truncation of linear matroids. *ACM Trans. Algorithms*, 14(2):14:1–14:20, 2018. doi:10.1145/3170444.
10. Dániel Marx. A parameterized view on matroid optimization problems. *Theor. Comput. Sci.*, 410(44):4471–4479, 2009. doi:10.1016/j.tcs.2009.07.027.
11. Tomás Masarík and Tomás Toufar. Parameterized complexity of fair deletion problems. *Discret. Appl. Math.*, 278:51–61, 2020. doi:10.1016/j.dam.2019.06.001.

[1] More accurately, we reduce to FO model checking on a *colored* version of G, where the vertices of G are partitioned into a number of *color classes*, and in the FO formula, we may require that a vertex belongs to a specific color class. Formal definitions can be found in the full version.