Parameterized Complexity of Conflict-Free Set Cover

Ashwin Jacob[1] · Diptapriyo Majumdar[2] · Venkatesh Raman[1]

Abstract
SET COVER is one of the well-known classical NP-hard problems. We study the *conflict-free* version of the SET COVER problem. Here we have a universe $\mathcal{U}$, a family $\mathcal{F}$ of subsets of $\mathcal{U}$ and a graph $G_{\mathcal{F}}$ on the vertex set $\mathcal{F}$ and we look for a subfamily $\mathcal{F}' \subseteq \mathcal{F}$ of minimum size that covers $\mathcal{U}$ and also forms an independent set in $G_{\mathcal{F}}$. We study conflict-free SET COVER in parameterized complexity by restricting the focus to the variants where SET COVER is fixed parameter tractable (FPT). We give upper bounds and lower bounds for the running time of conflict-free version of SET COVER with and without duplicate sets along with restrictions to the graph classes of $G_{\mathcal{F}}$. For example, when pairs of sets in $\mathcal{F}$ intersect in at most one element, for a solution of size k, we give

- an $f(k)|\mathcal{F}|^{o(k)}$ lower bound for any computable function f assuming ETH even if $G_{\mathcal{F}}$ is bipartite, but
- an $\mathcal{O}^*(3^{k^2})$ FPT algorithm ($\mathcal{O}^*$ notation ignores polynomial factors of input) when $G_{\mathcal{F}}$ is chordal.

A preliminary version appeared in the proceedings of CSR 2019 [18].

✉ Ashwin Jacob
ajacob@imsc.res.in

Diptapriyo Majumdar
diptapriyo.majumdar@rhul.ac.uk

Venkatesh Raman
vraman@imsc.res.in

[1] The Institute of Mathematical Sciences, HBNI, Chennai, India

[2] Royal Holloway, University of London, Egham, UK

1 Introduction and Previous Work

Covering problems are problems in combinatorics that ask whether a certain structure "covers" another. Covering problems are well-studied in theoretical computer science. Examples include VERTEX COVER, FEEDBACK VERTEX SET, CLUSTER VERTEX DELETION among others.

Several of these covering problems can be encapsulated by a problem called SET COVER which is one of the well-studied classical NP-hard problems. In the SET COVER problem, we have a universe $\mathcal{U}$, a family $\mathcal{F}$ of subsets of $\mathcal{U}$ and an integer k and the goal is to find a subfamily $\mathcal{F}'$ of size at most k such that $\bigcup_{S \in \mathcal{F}'} S = \mathcal{U}$.

SET COVER is very well-studied in a variety of algorithmic settings, especially in the realm of approximation algorithms and parameterized complexity. Unfortunately, SET COVER when parameterized by solution size k is W[2]-hard [8] and hence is unlikely to be fixed parameter-tractable (FPT).

It has been seen in computational problems [2–6, 9, 19, 21, 31, 32] where additional constraints are enforced on the solution we seek. One category of such problems are *choice* problems which can be described as follows. The set V from which we seek a solution subset is partitioned into groups and the solution requires to pick exactly one representative element from each of the groups. For example, consider the problem MULTICOLORED CLIQUE where the vertex set is partitioned into groups and the solution we seek is a clique such that exactly one vertex of the clique is present in each group. Another example in the geometric setting is by Arkin and Hassin [4] where they look at the following problem. Given a collection of points partitioned into groups and a matrix describing distance between pairs of points, find a set of points such that exactly one point is in each of the groups and the set has minimum diameter.

We can generalize these choice problems to a setting where we say that some pairs of elements in the problem are in conflict with each other and hence cannot go in the solution together. This can be modelled by defining a graph on the elements and an edge (u, v) is added if elements u and v do not go into the solution together or in other words form a conflict. Hence a solution without conflicts will form an independent set in this graph. Looking back at the example of MULTICOLORED CLIQUE, we have a conflict-graph where each group forms a clique and there are no edges across any pair of groups.

Conflict-free versions of classical problems in P like MAXIMUM FLOW [31], MAXIMUM MATCHING [9], SHORTEST PATH [21], KNAPSACK [32], BIN PACKING [11] and SCHEDULING [12] have been studied. A study of some geometric problems in the conflict-free setting was initiated recently [2, 3, 5, 6] motivated by various applications. Conflict-free version of graph problems like VERTEX COVER [19], FEEDBACK VERTEX SET [1], SPLIT VERTEX DELETION [19] have been studied from the parameterized point of view very recently. Some of the problems above are covering problems. Since SET COVER is a very general covering problem, studying the conflict-free version of SET COVER contributes to advancing this framework.

We look at the conflict-free version of SET COVER defined as follows:

CONFLICT-FREE SET COVER
Input: An universe $\mathcal{U}$, a family $\mathcal{F}$ of subsets of $\mathcal{U}$, a graph $G_{\mathcal{F}}$ with vertex set $\mathcal{F}$ and an integer k.
Goal: Is there a subfamily $\mathcal{F}' \subseteq \mathcal{F}$ of size at most k such that $\cup_{F \in \mathcal{F}'} F = \mathcal{U}$ and $\mathcal{F}'$ forms an independent set in $G_{\mathcal{F}}$?

Note that if $G_{\mathcal{F}}$ is edgeless, the problem is equivalent to SET COVER as every subset of vertices of $G_{\mathcal{F}}$ forms an independent set. Hence if $G_{\mathcal{F}}$ is from a graph class that contains edgeless graphs, when SET COVER is W[2]-hard with respect to some parameter, then CONFLICT-FREE SET COVER is also W[2]-hard with respect to the same parameter. Therefore, the only interesting cases of CONFLICT-FREE SET COVER are those special instances or parameterizations where SET COVER is FPT or when $G_{\mathcal{F}}$ is from a graph class that does not contain edgeless graphs.

One such example is SET COVER when parameterized by the size of the universe $\mathcal{U}$. There is a $2^{|\mathcal{U}|}(|\mathcal{U}| + |\mathcal{F}|)^{O(1)}$ time algorithm for this problem using dynamic programming over subsets of $\mathcal{U}$ [14]. Another example is a restricted version of SET COVER where every pair of sets in $\mathcal{F}$ intersect in at most c elements for a constant c. This version of SET COVER which we call c-INTERSECTION SET COVER is known to be FPT parameterized by k and has a kernel of universe size ck^2 and $\binom{ck^2}{c+1}$ sets [33]. Problems like COVERING POINTS BY LINES can be seen as special cases of c-INTERSECTION SET COVER [23].

We note that like the SET COVER problem, CONFLICT-FREE SET COVER is trivially FPT parameterized by $|\mathcal{F}|$ due to the simple brute-force algorithm of choosing at most k sets from $\mathcal{F}$.

Unlike the SET COVER problem, in CONFLICT-FREE SET COVER duplicate sets do play an important role. This is because two identical sets in the family $\mathcal{F}$ can have different neighborhood relations in the graph $G_{\mathcal{F}}$ which matters in the independence requirement of the solution. We study CONFLICT-FREE SET COVER both in the presence and absence of duplicate sets in $\mathcal{F}$. Note that if there are no duplicate sets in the family $\mathcal{F}$, $|\mathcal{F}| \leq 2^{|\mathcal{U}|}$.

Banik et al. [6] studied CONFLICT-FREE SET COVER in the context of some geometric covering problems having FPT algorithms. They showed that one of their geometric covering problems in the conflict-free setting is W[1]-hard parameterized by solution size k when $G_{\mathcal{F}}$ is from those classes of graphs where INDEPENDENT SET is W[1]-hard. They also showed that CONFLICT-FREE SET COVER is FPT parameterized by k whenever SET COVER is FPT parameterized by k when $G_{\mathcal{F}}$ has bounded arboricity.

Our results We focus on general CONFLICT-FREE SET COVER as well as the restricted version c-INTERSECTION SET COVER. Let us refer to the conflict-free version of c-INTERSECTION SET COVER as c-INTERSECTION CONFLICT-FREE SET COVER.

We refer to Tables 1 and 2 listing results for CONFLICT-FREE SET COVER and c-INTERSECTION CONFLICT-FREE SET COVER respectively.

- Our first result is an $f(k)|\mathcal{F}|^{o(k)}$ time lower bound for 1-INTERSECTION CONFLICT-FREE SET COVER assuming the Exponential Time Hypothesis (ETH). The lower bound holds even when $G_{\mathcal{F}}$ is restricted to bipartite graphs where INDEPENDENT SET is polynomial-time solvable. In contrast to this result, Banik et al. [6] showed hardness for their conflict-free geometric cover problem when $G_{\mathcal{F}}$ is from those classes of graphs where INDEPENDENT SET is W[1]-hard.

- For 1-INTERSECTION CONFLICT-FREE SET COVER with duplicate sets we give an $f(|\mathcal{U}|)|\mathcal{F}|^{o(|\mathcal{U}|)}$ lower bound assuming the ETH even when $G_{\mathcal{F}}$ is restricted to bipartite graphs where the INDEPENDENT SET problem can be solved in polynomial time.

 If there are no duplicate sets, the number of sets $|\mathcal{F}| \leq 2^{|\mathcal{U}|}$. Hence CONFLICT-FREE SET COVER is FPT as the trivial brute-force algorithm of choosing at most k sets from $\mathcal{F}$ is of complexity bounded by $\binom{|\mathcal{F}|}{k} \leq \binom{|\mathcal{F}|}{|\mathcal{U}|} \leq \binom{2^{|\mathcal{U}|}}{|\mathcal{U}|} \leq 2^{|\mathcal{U}|^2}$.

- For the upper bound $\binom{|\mathcal{F}|}{|\mathcal{U}|}$, we give a matching lower bound of $2^{o(|\mathcal{U}| \log |F|)}$ for any value of $|\mathcal{F}|$ as well assuming the ETH.

 We note that the problem does not have a polynomial kernel as when $G_{\mathcal{F}}$ is an empty graph, the problem becomes SET COVER parameterized by universe size which does not have a polynomial kernel unless $\mathtt{NP} \subseteq \mathtt{coNP/poly}$ [7].

- On the positive side, we provide meta-theorems giving FPT algorithms for CONFLICT-FREE SET COVER parameterized by k whenever SET COVER is

Table 1 Table of results: CONFLICT-FREE SET COVER

$G_{\mathcal{F}}$ restriction	Solution size k	Universe size $|\mathcal{U}|$
General graph	W[1]-hard [6]	$\binom{|\mathcal{F}|}{|\mathcal{U}|} \leq 2^{|\mathcal{U}|^2}$ algorithm without duplicates, $2^{|\mathcal{U}| \log |\mathcal{F}|}$ ℓb (Theorem 5), W[1]-hard with duplicates (Lemma 3)
Empty graph	FPT* †	FPT †, npk †
Bipartite	$f(k)|\mathcal{F}|^{o(k)}$ ℓb (Theorem 1)	W[1]-hard with duplicates (Lemma 3)
Chordal		FPT (Theorem 9), npk †
d-degenerate	FPT* [6]	FPT †, npk †
Nowhere Dense	FPT* (Theorem 7)	FPT (Theorem 7), npk †
Bounded #MIS	FPT* (Theorem 6)	FPT (Theorem 6), npk †

FPT* denotes that the problem is FPT whenever the SET COVER variant is FPT, #MIS abbreviates 'number of maximal independent sets', † denotes results from existing literature (other than those cited), ℓb abbreviates 'lower bound', npk abbreviates 'no polynomial kernel'

Table 2 Table of results: c-INTERSECTION CONFLICT-FREE SET COVER with duplicates

$G_\mathcal{F}$ restriction	Solution size k	Universe size $|\mathcal{U}|$
General graph	W[1]-hard even if $c = 1$ (Theorem 1)	W[1]-hard even if $c = 1$ (Lemma 3)
Empty graph	FPT, Polynomial Kernel [33]	FPT, Polynomial Kernel [33]
Bipartite	W[1]-hard even if $c = 1$ (Theorem 1)	W[1]-hard even if $c = 1$ (Lemma 3)
Chordal	FPT (Theorem 10)	FPT (Theorem 9)
Cluster	FPT (Corollary 2), Polynomial Kernel (Theorem 11)	FPT (Corollary 2) Polynomial Kernel (Theorem 11)
d-degenerate	FPT † [6]	FPT †
Nowhere Dense	FPT (Theorem 7)	FPT (Theorem 7)
Bounded #MIS	FPT (Theorem 6)	FPT (Theorem 6)

FPT* denotes that the problem is FPT whenever the SET COVER variant is FPT, #MIS denote number of maximal independent sets, † denotes results from existing literature (other than those cited)

FPT and $G_\mathcal{F}$ belongs to graph classes which are sparse; for example, graphs of bounded degeneracy or nowhere dense graphs. This is proved using the recently introduced independence covering family [26]. Furthermore, if $G_\mathcal{F}$ is a dense graph like split or co-chordal, we give FPT algorithm whenever SET COVER is FPT. This algorithm works for a large class of graphs where the number of maximal independent sets is polynomial in the number of vertices (that are sets in the family in our case).

- For c-INTERSECTION CONFLICT-FREE SET COVER, we give an FPT algorithm parameterized by k when we restrict $G_\mathcal{F}$ to chordal graphs. This contrasts the hardness result we have for c-INTERSECTION CONFLICT-FREE SET COVER in bipartite graphs.
- Furthermore, when we restrict $G_\mathcal{F}$ to a subclass of chordal graphs called cluster graphs, we obtain a polynomial kernel for c-INTERSECTION CONFLICT-FREE SET COVER parameterized by k.
- For CONFLICT-FREE SET COVER parameterized by $|\mathcal{U}|$, since solution size $k \leq |\mathcal{U}|$, the FPT results listed above for CONFLICT-FREE SET COVER parameterized by k also follow for $|\mathcal{U}|$. Furthermore, we give an FPT algorithm for CONFLICT-FREE SET COVER parameterized by $|\mathcal{U}|$ even in presence of duplicates when we restrict $G_\mathcal{F}$ to interval graphs via a dynamic programming algorithm using the ordering of the corresponding intervals. We extend this idea and give an FPT algorithm for chordal graphs which is a superclass of interval graphs via dynamic programming on the clique tree decomposition of the graph.
- We also study the CONFLICT-FREE SET COVER problem where there is an underlying (linearly representable) matroid on the family of subsets, and we want the solution to be an independent set in the matroid. Banik et al. [6] studied this version for a specialization of SET COVER where the sets are intervals on a real line.

We show that even the more general problem (where the sets in the family are arbitrary) is FPT when parameterized by the universe size, using the idea of dynamic programming over representative families [15].

We note that this result can be obtained as a corollary of a result by Bevern et al. [34] where they give algorithms for a generalization of our problem called uncapacitated facility location problem with multiple matroid constraints. But our algorithm is simpler and has a better running time when the corresponding matroid has huge rank.

Structure of the Paper In Section 2, we introduce notions used and give definitions related to graphs, matroids and basic notions of parameterized complexity. In Section 3, we give the hardness results of some of the variants of CONFLICT-FREE SET COVER . In Section 4, we give some FPT algorithms and kernels for some variants of CONFLICT-FREE SET COVER when the conflict-graph is restricted to various graph classes. Finally in Section 5, we give an FPT algorithm for MATROIDAL CONFLICT-FREE SET COVER parameterized by universe size.

2 Preliminaries

We use $[n]$ to denote the set $\{1, \ldots, n\}$. We use the standard terminologies of the graph theory book by Diestel [10]. For a graph $G = (V, E)$, we denote n as the number of vertices and m as the number of edges. For $S \subseteq V(G)$, we denote $G[S]$ to be the subgraph induced on S. A *complement* of graph G is a graph H on the same vertices such that two distinct vertices of H are adjacent if and only if they are not adjacent in G. A set $S \subseteq V(G)$ is called an *independent set* if for all $u, v \in S, (u, v) \notin E(G)$. Similarly for a subset $S \subseteq V(G)$, $G[S]$ is called a *clique* if for every $u, v \in S, (u, v) \in E(G)$. A neighborhood of a vertex v in a graph G is the set $N_G(v) = \{u \in V(G) : (u, v) \in E(G)\}$. Elements in $N_G(v)$ are called the neighbours of v. The neighborhood of v, denoted by $N_G[v]$ is $v \cup N_G(v)$.

An *interval* graph is a graph formed from a set of intervals on the real line, with a vertex for each interval and an edge between vertices whose intervals intersect. A *chord* in a cycle is an edge between two non-adjacent vertices of the cycle. A *chordal* graph is a graph in which any cycle of four or more vertices has a chord. A *cluster* graph is a graph where every connected component forms a clique. A graph G is said to be *d-degenerate* if every subgraph of G has a vertex of degree at most d. The *arboricity* of a graph is the minimum number of forests into which its edges can be partitioned.

A *k-subdivision* of a graph G is the graph created from G by subdividing every edge by exactly k vertices. A graph class is a *somewhere dense* graph class when there exists a threshold t such that every complete graph appears as a t-subdivision in a subgraph of a graph in the class. To the contrary, if such a threshold does not exist, the class is *nowhere dense*. We refer to [29] for more details on nowhere dense graphs.

Definition 1 (Tree decomposition) Given a graph $G = (V, E)$, a *tree decomposition* is a pair (X, T), where $X = \{X_1, \ldots, X_n\}$ is a family of subsets of V, and T is a tree whose nodes are the subsets X_i, satisfying the following properties:

- $\bigcup\limits_{i=1}^{n} X_i = V$.
- For all edges $(u, v) \in E$, there is a subset X_i that contains both u and v.
- If X_i and X_j both contain a vertex v, then all nodes X_k of the tree in the (unique) path between X_i and X_j contain v as well.

The sets X_i are called the bags corresponding to node i.

Definition 2 (Treewidth) The width of tree decomposition (X, T) equals $\max_{t \in V(T)} |X_t| - 1$. The *treewidth* of a graph G is the minimum possible width of a tree decomposition of G.

Definition 3 (Nice Tree decomposition) A *nice tree decomposition* is a tree decomposition T satisfying the following properties:

- Let us arbitrarily root the tree T. For the root of the tree r, $X_r = \emptyset$.
- $X_l = \emptyset$ for all the leaf nodes of the tree.
- Every other node of T are one of three types:
    - Introduce Node: A node i with exactly one child j such that $X_i = X_j \cup \{v\}$ for some vertex $v \notin X_j$.
    - Forget Node: A node i with exactly one child j such that $X_i = X_j \setminus \{v\}$ for some vertex $v \in X_j$.
    - Join Node: A node i with exactly two children j and j' such that $X_i = X_j = X_{j'}$.

Lemma 1 (Lemma 7.4 of [8], Lemma 13.1.3 of [22]) *Given a tree decomposition (X, T) of a graph G of width at most k, in polynomial time one compute a nice tree decomposition (X', T') of G of width at most k that has at most $\mathcal{O}(k|V(G)|)$ nodes. Moreover, for each $t' \in V(T')$, there is a $t \in V(T)$ such that $X_{t'} \subseteq X_t$.*

Definition 4 (Matroid) A *matroid* M is a pair $(E, \mathcal{I})$ where E is the ground set and $\mathcal{I}$ is the family of subsets of E (called the independent sets of M) satisfying the following properties:

- $\emptyset \in \mathcal{I}$.
- If $A' \subseteq A$ and $A \in \mathcal{I}$, then $A' \in \mathcal{I}$.
- If $A, B \in \mathcal{I}$ and $|A| < |B|$, then there exists an $e \in B \setminus A$ such that $A \cup \{e\} \in \mathcal{I}$.

Definition 5 (Rank of a matroid) For a matroid $M = (E, \mathcal{I})$, an inclusion wise maximal set of $\mathcal{I}$ is called a basis of the matroid. It can be shown that all the bases of a matroid have the same size. This size is called the rank of the matroid M.

Definition 6 (Linear Matroid) Let A be a matrix over a field $\mathbb{F}$ and let E be the set of columns of A. We define a matroid $M = (E, \mathcal{I})$ as follows: A subset $X \subseteq E$ is an independent set of M if and only if the corresponding columns of A are linearly independent over $\mathbb{F}$. The matroids that can be defined by such a construction over some field $\mathbb{F}$ are called *linear matroids*. The matrix A corresponding to the matroid M is called the *linear representation* of M.

For more details on matroids we refer to [30].

2.1 Parameterized Complexity Notions

Definition 7 (Fixed Parameter Tractability) A subset $L \subseteq \Sigma^* \times \mathbb{N}$ is a parameterized language. A parameterized language L is said to be *fixed parameter tractable* (or FPT) if there exists an algorithm $\mathcal{B}$, a constant c and a computable function f such that $\forall x, \forall k$, $\mathcal{B}$ on input (x, k) runs in at most $f(k) \cdot |x|^c$ time and outputs 1 if and only if $(x, k) \in L$. We call the algorithm $\mathcal{B}$ a fixed parameter algorithm (or FPT algorithm).

Definition 8 (Parameterized Reduction) Let $P_1, P_2 \in \Sigma^* \times \mathbb{N}$ be two parameterized languages. Suppose there exists an algorithm $\mathcal{B}$ that takes input (I, k) (an instance of P_1) and constructs an instance (I', k') of P_2 such that the following conditions are satisfied.

- (I, k) is YES-INSTANCE if and only if (I', k') is YES-INSTANCE.
- $k' \in f(k)$ for some function depending only on k.
- Algorithm $\mathcal{B}$ runs in $g(k)|I|^{\mathcal{O}(1)}$ time.

Then we say that there exists a *parameterized reduction* from P_1 to P_2.

A closely related notion to fixed parameter tractability is the notion of Kernelization defined below.

Definition 9 (Kernelization) Let $L \subseteq \sum^* \times \mathbb{N}$ be a parameterized language. *Kernelization* is a procedure that replaces the input instance (I, k) by a reduced instance (I', k') such that

- $k' \leq f(k)$, $|I'| \leq g(k)$ for some function f, g depending only on k.
- $(I, k) \in L$ if and only if $(I', k') \in L$.
- The reduction from (I, k) to (I', k') must be computable in $poly(|I| + k)$ time.

If $g(k) = k^{\mathcal{O}(1)}$ then we say that L admits a *polynomial kernel*.

We now define a notion of reduction rule that is useful in designing kernels.

Definition 10 (Reduction Rule) Let $L \subseteq \sum^* \times \mathbb{N}$ be a parameterized language. A *reduction rule* is a procedure computable in $poly(|I| + k)$ time that replaces the

input instance (I, k) by a reduced instance (I', k') such that $(I, k) \in L$ if and only if $(I', k') \in L$.

The property of the reduction rule that it translates an instance to an equivalent one, i.e $(I, k) \in L$ if and only if $(I', k') \in L$ is called the *safeness* of the reduction rule.

W-hierarchy In order to capture parameterized languages being FPT or not, the W-hierarchy is defined as FPT $\subseteq$ W[1] $\subseteq \ldots \subseteq$ XP. It is believed that this subset relation is strict. Hence a parameterized language that is hard for some complexity class above FPT is unlikely to be FPT. If a parameterized language $L \subseteq \Sigma^* \times \mathbb{N}$ can be solved by an algorithm running in $\mathcal{O}(n^{f(k)})$ time, then we say $L \in$ XP. In such situation we also say that L admits an XP algorithm.

We use the following conjecture to prove lower bounds.

Conjecture 1 (Exponential Time Hypothesis (ETH)([17])) 3-CNF-SAT *cannot be solved in $2^{o(n)}$ time where the input formula has n variables and $m = \mathcal{O}(n)$ clauses.*

For more details of parameterized complexity, we refer to [8].

3 Hardness Results for Conflict-Free Set Cover

3.1 1-Intersection Conflict-Free Set Cover Parameterized by Solution Size k

The problem c-INTERSECTION SET COVER is known to be in FPT [33]. On the contrary, for the conflict-free version we show the following.

Theorem 1 1-INTERSECTION CONFLICT-FREE SET COVER *cannot be solved in time $f(k)|\mathcal{F}|^{o(k)}$ for solution size k in bipartite graphs for any computable function f assuming the ETH.*

Proof We give a reduction from the problem MULTICOLORED BICLIQUE [8] defined as follows:

MULTICOLORED BICLIQUE   **Parameter:** k
Input: A bipartite graph $G = (A \cup B, E)$, an integer k, a partition of A into k sets $A_1, A_2, \ldots, A_k$ and a partition of B into k sets $B_1, B_2, \ldots, B_k$.
Question: Does there exist a subgraph of G isomorphic to the biclique $K_{k,k}$ with one vertex from each of the sets A_i and B_i?

Given an instance of $(G, k, A_1, \ldots, A_k, B_1, \ldots, B_k)$ of MULTICOLORED BICLIQUE with $V(G) = \{v_1, v_2, \ldots, v_n\}$, we construct an instance of CONFLICT-FREE SET COVER $(\mathcal{U}, \mathcal{F}, G_{\mathcal{F}}, 2k + 1)$ without duplicates as follows:

We define the universe $\mathcal{U} = [2k] \cup V(G) \cup \{x\}$. Now, we associate a set corresponding to each vertex of the graph G. For a vertex v_j of $V(G)$, let S_{v_j} denote the

corresponding set. For $i \in [k]$, if $v_j \in A_i$, define $S_{v_j} = \{v_j, i\}$. For $i \in [2k] \setminus [k]$, if $v_j \in B_{i-k}$, define $S_{v_j} = \{v_j, i\}$. Define a set $D = V(G) \cup \{x\}$. We have $\mathcal{F} = \bigcup_{v \in V(G)} S_v \cup \{D\}$. The graph $G_\mathcal{F}$ is obtained by taking the complement of the graph G, removing all the edges in $G[A]$ and $G[B]$ and adding an isolated vertex corresponding to the set D. Note that the graph $G_\mathcal{F}$ remains bipartite.

Note that $\mathcal{F}$ is defined in such a way that all pairs of sets intersect in at most one element. Also there are no duplicate sets in this instance as only the set S_{v_j} other than D contains the element v_j and only D contains the element x.

We claim that $(G, k, A_1, \ldots, A_k, B_1, \ldots, B_k)$ is a YES-INSTANCE of MULTI-COLORED BICLIQUE if and only if $(\mathcal{U}, \mathcal{F}, G_\mathcal{F}, 2k + 1)$ is a YES-INSTANCE of 1-INTERSECTION CONFLICT-FREE SET COVER.

Let $S = \{a_1, \ldots, a_k, b_1, \ldots, b_k\}$ be the vertices in G that form a multicolored biclique. Then $\mathcal{F}' = \{D, S_{a_1}, \ldots, S_{a_k}, S_{b_1}, \ldots, S_{b_k}\}$ covers $\mathcal{U}$ as D covers $V(G) \cup \{x\}$ and $i \in S_{a_i}$ for $i \in [k]$ and $i \in S_{b_{i-k}}$ for $i \in [2k] \setminus [k]$. Since the edges across A and B in G are non-edges in $G_\mathcal{F}$ and D is an isolated vertex, $\mathcal{F}'$ forms an independent set in $G_\mathcal{F}$. In the reverse direction, let $\mathcal{F}' = \{S_1, \ldots, S_{2k+1}\}$ be a solution of size $2k + 1$ covering $\mathcal{U}$. The set D has to be part of the solution $\mathcal{F}'$ as only the set D contains the element x. Now note that an element $i \in [k]$ can be covered only by sets S_v where $v \in A_i$. Similarly an element $i \in [2k] \setminus [k]$ can be covered only by sets S_v where $v \in B_{i-k}$. Hence the vertices of the sets in $\mathcal{F}'$ are such that there is at least one vertex from each of the sets A_i and B_i. Since the budget is limited to $2k$ after picking D, exactly one vertex from each of the sets A_i and B_i is contained in $\mathcal{F}'$. Since the vertices $\mathcal{F}' \setminus \{D\}$ form an independent set in the bipartite graph $G_\mathcal{F}$, the corresponding vertices form a biclique in G.

Since MULTICOLORED BICLIQUE cannot be solved in time $f(k)|V(G)|^{o(k)}$ for solution size k assuming ETH [28], the theorem follows. $\qquad \square$

3.2 Conflict-Free Set Cover Parameterized by $|\mathcal{U}|$

In the section, we give lower bound results for CONFLICT-FREE SET COVER when parameterized by the universe size $|\mathcal{U}|$. We study the problem in both the cases when the family $\mathcal{F}$ has duplicate sets and when it does not.

3.2.1 The Family $\mathcal{F}$ has Duplicates

In this section, we study the case when duplicate sets are allowed in the family $\mathcal{F}$. Banik et al. [6] has the following hardness result for this case when the graph $G_\mathcal{F}$ is restricted to a class where finding independent set of size k is W[1]-hard.

Theorem 2 (Banik et al. [6]) *If for a subclass of graphs $\mathcal{G}$, finding an independent set of size k is W[1]-hard parameterized by k, then* CONFLICT-FREE SET COVER *parameterized by $|\mathcal{U}|$ is W[1]-hard when $G_\mathcal{F}$ is restricted to the class $\mathcal{G}$.*

Bipartite graphs is one class of graphs where the INDEPENDENT SET problem can be solved in polynomial time. In contrast to Theorem 2, we show that

1-INTERSECTION CONFLICT-FREE SET COVER on bipartite graphs is W[1]-hard.
Note that in Theorem 1 proven previously, the size of the universe can be much larger
than the solution size k and hence the hardness result does not follow from it.

Theorem 3 1-INTERSECTION CONFLICT-FREE SET COVER *parameterized by* $|\mathcal{U}|$
is W[1]-*hard on bipartite graphs.*

Proof We again give a reduction from the W[1]-hard problem MULTICOLORED
BICLIQUE.

Given an instance of MULTICOLORED BICLIQUE, we construct an instance of 1-
INTERSECTION CONFLICT-FREE SET COVER as follows: $\mathcal{U} = [2k]$. Let S_v denote
the set corresponding to vertex v we add to $\mathcal{F}$. For $i \in [k]$, if $v \in A_i$, define $S_v = \{i\}$.
For $i \in [2k] \setminus [k]$, if $v \in B_{i-k}$, define $S_v = \{i\}$. The graph G' is obtained by
complementing the graph G and removing edges in the graphs $G[A]$ and $G[B]$. The
graph G' remains bipartite. Since every set in $\mathcal{F}$ is of size one, sets can pairwise
intersect in at most one elements. Hence we can conclude that the instance we have
constructed is a valid 1-INTERSECTION CONFLICT-FREE SET COVER instance.

Note that the construction is very similar to that in Theorem 1, the difference being
the vertex v is not added to sets S_v.

The correctness proof follows similar to Theorem 1. □

3.2.2 The Family $\mathcal{F}$ has No Duplicates

If there are no duplicate sets, the number of sets $|\mathcal{F}| \leq 2^{|\mathcal{U}|}$. Hence CONFLICT-FREE
SET COVER is FPT as the trivial brute-force algorithm of choosing at most k sets
from $\mathcal{F}$ is of complexity bounded by $\binom{|\mathcal{F}|}{k} \leq \binom{|\mathcal{F}|}{|\mathcal{U}|} \leq \binom{2^{|\mathcal{U}|}}{|\mathcal{U}|} \leq 2^{|\mathcal{U}|^2}$. In this section,
we give a lower bound of $2^{o(|\mathcal{U}| \log |\mathcal{F}|)}$ under ETH for CONFLICT-FREE SET COVER
without duplicates when $G_{\mathcal{F}}$ is bipartite. We do so by giving an appropriate reduction
from the following variant of MULTICOLORED BICLIQUE defined below.

SMALL MULTICOLORED BICLIQUE **Parameter:** k
Input: A bipartite graph $G = (A \cup B, E)$, an integer k, a partition of A into
k sets $A_1, A_2, \ldots, A_k$ and a partition of B into k sets $B_1, B_2, \ldots, B_k$ such that
$|A_i| = |B_i| = s$ where $k \leq s \leq 2^k/2k$.
Question: Does there exist a subgraph of G isomorphic to the biclique $K_{k,k}$ with
one vertex from each of the sets A_i and B_i?

We first note that the reduction from 3-COLORING used in [24] can be modified
so that we get the following lower bound for SMALL MULTICOLORED BICLIQUE.

Theorem 4 SMALL MULTICOLORED BICLIQUE *cannot be solved in time* $2^{o(k \log s)}$
under the ETH.

Proof We give a reduction from 3-COLORING problem. Let G, a graph with N vertices be the instance of 3-Coloring problem.

Let $k = \frac{N \cdot \log 3}{\log s}$.

Divide vertices of G into k groups $V_1, V_2, \dots V_k$ of equal size, each size being $\frac{\log s}{\log 3}$.

For each set V_i, list out all the possible valid 3-colorings. There would be at most $3^{|V_i|} \le 3^{\log s/\log 3} = 2^{\log 3 \cdot \log s/\log 3} = 2^{\log s} = s$ colorings. If there is no valid coloring for some V_i, we can conclude that we have a NO-INSTANCE of 3-COLORING. Duplicate some valid colorings so that the number of colorings is exactly s. Let us call list of colorings of V_i as P_i. Let $P = \cup_i P_i$.

Create a graph H with two copies of P, A and B as its vertex set with the corresponding partitions $P_1, \dots, P_k$ being $A_1, \dots, A_k$ and $B_1, \dots, B_k$. Let (A_i, c) and (B_i, c) denote the vertex corresponding to coloring c in sets A_i and B_i respectively. We add edges as follows:

Look at colorings $c_1 \in P_i$ and $c_2 \in P_j$. If $i \ne j$ and the colorings c_1 of $G[V_i]$ and c_2 of $G[V_j]$ together forms a valid coloring in the graph $G[V_i \cup V_j]$, add edges from vertex (A_i, c_1) to (B_j, c_2) and from (A_j, c_2) to (B_i, c_1).

Now we claim that $(H, A_1, \dots, A_k, B_1, \dots, B_k, k)$ is a YES-INSTANCE of SMALL MULTICOLORED BICLIQUE if and only if G has a 3-coloring. For the reverse direction, let C be a valid 3-coloring of G. Let $C|_{V_i}$ denote the coloring C restricted to V_i. We claim the vertices $(A_i, C|_{V_i})$ and $(B_i, C|_{V_i})$ forms a biclique. Suppose not. Then there is an absence of edge between two vertices $(A_{i_1}, C|_{V_{i_1}})$ and $(B_{i_2}, C|_{V_{i_2}})$. But then this means that $C|_{V_{i_1}} \cup C|_{V_{i_2}} = C|_{V_{i_1} \cup V_{i_2}}$ is not a valid coloring of $G[V_{i_1} \cup V_{i_2}]$ giving a contradiction.

For the forward direction, let the vertices $(A_1, c_1), \dots, (A_k, c_k), (B_1, c_1), \dots, (B_k, c_k)$ form a biclique. We say that $\cup_i c_i$ is a valid coloring of the graph G. Suppose not. Then there is a monochromatic edge (u, v) in G. Both u and v cannot belong to a group V_i as corresponding 3-coloring c_i is a valid 3-coloring of $G[V_i]$. So u and v belong to different groups i_1 and i_2. But then there will not be an edge between vertices (A_{i_1}, c_{i_1}) and (B_{i_2}, c_{i_2}) as c_{i_1} and c_{i_2} together does not form a valid 3-coloring of $G[V_{i_1} \cup V_{i_2}]$. contradicting that the vertices $(A_1, c_1), \dots, (A_k, c_k), (B_1, c_1), \dots, (B_k, c_k)$ form a biclique.

Now suppose there is $2^{o(k \log s)}$ running time algorithm for SMALL MULTICOLORED BICLIQUE. Then there is a $2^{o(N \cdot \log 3)} = 2^{o(N)}$ time algorithm for 3-coloring violating the ETH. □

Theorem 5 CONFLICT-FREE SET COVER *without duplicates when $G_{\mathcal{F}}$ is bipartite cannot be solved in time $2^{o(|\mathcal{U}| \log |\mathcal{F}|)}$ under ETH.*

Proof Given an instance of $(G, A_1, \dots, A_k, B_1, \dots, B_k)$ of SMALL MULTICOLORED BICLIQUE with $V(G) = \{v_1, v_2, \dots, v_n\}$, we construct an instance of CONFLICT-FREE SET COVER $(\mathcal{U}, \mathcal{F}, G_{\mathcal{F}}, 2k + 1)$ without duplicates as follows:

Let us define sets $Z = \{z_1, z_2, \dots, z_{\lceil \log n \rceil}\}$ and $O = \{o_1, o_2, \dots, o_{\lceil \log n \rceil}\}$.

We define the universe $\mathcal{U} = [2k] \cup Z \cup O \cup \{x\}$.

Let us look at vertex $v_j \in V$ and construct sets $S_{v_j} \in \mathcal{F}$. Let us map j to its binary representation $b_1, b_2, \ldots, b_{\lceil \log n \rceil}$ where b_i denotes the ith bit of the number j. We create a set T_j as follows: for all $i \in [\lceil \log n \rceil]$, when $b_i = 0$, add z_i to T_j, else add o_i to T_j. For $i \in [k]$, if $v_j \in A_i$, define $S_{v_j} = \{i\} \cup T_j$. For $i \in [2k] \setminus [k]$, if $v_j \in B_{i-k}$, define $S_{v_j} = \{i\} \cup T_j$. Define another set $D = Z \cup O \cup \{x\}$. We have $\mathcal{F} = \bigcup_{v \in V(G)} S_v \cup \{D\}$. The graph $G_{\mathcal{F}}$ is obtained by taking the complement of the graph G, removing the edges in the graphs $G[A]$ and $G[B]$ independent and adding an isolated vertex corresponding to the set D. Note that the graph $G_{\mathcal{F}}$ remains bipartite.

Note that the construction is almost exactly the same as in Theorem 1 but the vertices are encoded in binary form.

We now claim that $(G, k, A_1, \ldots, A_k, B_1, \ldots, B_k)$ is a YES-INSTANCE of SMALL MULTICOLORED BICLIQUE if and only if $(\mathcal{U}, \mathcal{F}, G_{\mathcal{F}}, 2k + 1)$ is a YES-INSTANCE of CONFLICT-FREE SET COVER .

Let $S = \{a_1, \ldots, a_k, b_1, \ldots, b_k\}$ be the vertices in G that form a multicolored biclique. Then $\mathcal{F}' = \{D, S_{a_1}, \ldots, S_{a_k}, S_{b_1}, \ldots, S_{b_k}\}$ covers $\mathcal{U}$ as D covers $Z \cup O \cup \{x\}$ and $i \in S_{a_i}$ for $i \in [k]$ and $i \in S_{b_{i-k}}$ for $i \in [2k] \setminus [k]$. Since the edges across A and B in G are non-edges in $G_{\mathcal{F}}$ and D is an isolated vertex, $\mathcal{F}'$ forms an independent set in $G_{\mathcal{F}}$. In the reverse direction, let $\mathcal{F}' = \{S_1, \ldots, S_{2k+1}\}$ be a solution of size $2k + 1$ covering $\mathcal{U}$. The set D has to be part of the solution $\mathcal{F}'$ as only the set D contains the element x. Now note that an element $i \in [k]$ can be covered only by sets S_v where $v \in A_i$. Similarly an element $i \in [2k] \setminus [k]$ can be covered only by sets S_v where $v \in B_{i-k}$. Hence the vertices of the sets in $\mathcal{F}'$ are such that there is at least one vertex from each of the sets A_i and B_i. Since the budget is limited to $2k$ after picking D, exactly one vertex from each of the sets A_i and B_i is contained in $\mathcal{F}'$. Since the vertices $\mathcal{F}' \setminus \{D\}$ form an independent set in the bipartite graph $G_{\mathcal{F}}$, the corresponding vertices form a biclique in G.

Note that in the SMALL MULTICOLORED BICLIQUE instance, $n = 2k \cdot s \leq 2^k$. Since $\log n \leq k$, $|U| \leq 4k + 1$.

Now suppose CONFLICT-FREE SET COVER has an algorithm with running time $2^{o(|\mathcal{U}| \log |\mathcal{F}|)}$. Since $s = \frac{|\mathcal{F}|}{2k}$ and $|\mathcal{U}| \leq 4k + 1$, we have a running time of $2^{o(4k \log(2k \cdot s))} = 2^{o(k(\log s + \log k))} = 2^{o(k \log s)}$ for SMALL MULTICOLORED BICLIQUE violating the ETH.   □

4 Algorithms

In this section, we give algorithms for variants of CONFLICT-FREE SET COVER when the graph $G_{\mathcal{F}}$ is restricted to different graph classes.

4.1 Conflict-Free Set Cover Parameterized by Solution Size k

In the following results, we restrict the graph $G_{\mathcal{F}}$.

4.1.1 Graphs With Bounded Number of Maximal Independent Sets

Theorem 6 *When $G_\mathcal{F}$ is restricted to a graph where the number of maximal independent sets is polynomial in $|\mathcal{F}|$, if the restricted variant of* SET COVER *can be solved in $\mathcal{O}^*(f(k))$ time, then the corresponding* CONFLICT-FREE SET COVER *variant can be solved in $\mathcal{O}^*(f(k))$ time.*

Proof We first note that since the maximal independent sets of a graph can be enumerated with polynomial delay(the maximum time taken between outputting two consecutive solutions) [20], they can be enumerated in time polynomial in $|\mathcal{F}|$ for the given graph $G_\mathcal{F}$.

For each maximal independent set I of $G_\mathcal{F}$, we run the $\mathcal{O}^*(f(k))$ algorithm for SET COVER with the family $\mathcal{F}$ containing sets corresponding to the vertices in I. Since the solution X of CONFLICT-FREE SET COVER is an independent set, $X \subseteq I'$ for some maximal independent set I'. So if the SET COVER algorithm returns YES for any I, return YES, else return NO.   □

As the number of maximal independent sets in split graphs (since at most one vertex of the clique can be in the independent set), co-chordal graphs [16] and $2K_2$-free graphs [13] are polynomial in the number of vertices and can be enumerated in polynomial time, we have the following corollary.

Corollary 1 *If* SET COVER *can be solved in $\mathcal{O}^*(f(k))$ time, then* CONFLICT-FREE SET COVER *can be solved in $\mathcal{O}^*(f(k))$ time when $G_\mathcal{F}$ is restricted to split graphs, co-chordal graphs or $2K_2$-free graphs.*

4.1.2 Nowhere Dense Graphs

Nowhere dense graph class contains a number of graph classes such as graphs with bounded degree, graphs with bounded local treewidth, graphs with bounded expansion and graphs that locally exclude a fixed minor.

We define the notion of *k-Independence Covering Family* introduced by [26].

Definition 11 (*k-Independence Covering Family*) For a graph G and integer k, a family of independent sets of G is called an independence covering family for (G, k), denoted by $\mathscr{F}(G, k)$, if for any independent set X in G of size at most k, there exists an independent set $Y \in \mathscr{F}(G, k)$ such that $X \subseteq Y$.

In [26], the authors construct a k-independence covering family for nowhere dense graphs.

Lemma 2 (Lokshtanov et al. [26]) *Let G be a nowhere dense graph and k be an integer. There is a deterministic algorithm that runs in time*

$$\mathcal{O}\left(f\left(k, \frac{1}{k}\right) \cdot n^{1+o(1)} + g(k) \cdot \binom{k^2}{k} \cdot 2^{o(k^2)} \cdot n(n+m)\log n \right)$$

and outputs a k-independence covering family for (G, k) of size $\mathcal{O}(g(k)\binom{k^2}{k} \cdot 2^{o(k^2)} \cdot n \log n)$ where f is a computable function and $g(k) = (f(k, \frac{1}{k}))^k$.

We get the following theorem.

Theorem 7 *If the restricted variant of* SET COVER *can be solved in $\mathcal{O}^*(h(k))$ time with solution size k and a computable function h, then the corresponding* CONFLICT-FREE SET COVER *variant has an algorithm with running time $\mathcal{O}^*(h(k)g(k)\binom{k^2}{k} \cdot 2^{o(k^2)})$ for nowhere dense graphs for a computable function g.*

Proof We use Lemma 2 on $G_\mathcal{F}$ to get a k-independence covering family $\mathscr{F}(G_\mathcal{F}, k)$. For each independent set $Y \in \mathscr{F}(G_\mathcal{F}, k)$, we run the algorithm for SET COVER for the instance $(\mathcal{U}, Y, k)$ in $\mathcal{O}^*(h(k))$ time. If for any of the sets Y, $(\mathcal{U}, Y, k)$ is a YES-INSTANCE, we return YES. Otherwise we return NO.

Let X be the solution of size k. There is a set Y in $\mathscr{F}(G_\mathcal{F}, k)$ such that $X \subseteq Y$. Hence when we run the algorithm for SET COVER in instance $(\mathcal{U}, Y, k)$, since $G[Y]$ is an independent set, the algorithm will return X. □

We note that Banik et al. [6] has proven that CONFLICT-FREE SET COVER is FPT parameterized by k if the SET COVER variant is FPT parameterized by k when $G_\mathcal{F}$ is a graph of bounded arboricity. The result also holds for graphs with bounded degeneracy as the degeneracy of a graph is also bounded when the arboricity is bounded. A k-Independence Covering Family can also be constructed for graphs with bounded degeneracy [26]. We note that an alternate algorithm for CONFLICT-FREE SET COVER parameterized by k when $G_\mathcal{F}$ has bounded degeneracy can be obtained using the ideas used for nowhere dense graphs earlier. Note that graphs with bounded degeneracy contain many other graph classes such as planar graphs and graphs with bounded treewidth.

4.2 Conflict-Free Set Cover Parameterized by $|\mathcal{U}|$ when $\mathcal{F}$ has Duplicates

We remind that when $\mathcal{F}$ has no duplicates, CONFLICT-FREE SET COVER parameterized by $|\mathcal{U}|$ is trivially FPT as $|\mathcal{F}| \leq 2^{|\mathcal{U}|}$. Hence we focus the case when there are duplicate sets in $\mathcal{F}$. Again we restict the graph $G_\mathcal{F}$.

4.2.1 Interval Graphs

Before we state our result, let us focus on some properties of interval graphs. Let us order the vertices of a given interval graph G as $v_1, \ldots, v_n$ based on the increasing value of their left endpoints. Let the indices $1, \ldots, n$ denote the intervals. Let $l(i)$ and $r(i)$ denote the left and right endpoints of interval i respectively.

Look at a vertex v_i and its neighborhood $N(v_i)$ in the set $\{v_{i+1}, \ldots, v_n\}$. Let $v_j, v_k \in \{v_{i+1}, \ldots, v_n\}$ such that $(v_i, v_j) \in E(G)$ and $(v_i, v_k) \notin E(G)$. By definition, v_i and v_j has an edge if intervals i and j intersect. Hence $l(j) \leq r(i)$. Also since intervals i and k do not intersect, $r(i) \leq l(k)$. Hence we have $l(j) \leq l(k)$. Since

this is true for any non-neighbor of v_i in $\{v_{i+1}, \ldots, v_n\}$, we have shown that all the non-neighbors of v_i to its right comes after the last neighbor of v_i to its right. We make use of this ordering to give a dynamic programming algorithm for CONFLICT-FREE SET COVER with duplicates on interval graphs. Note that the ordering can be obtained in time linear in $|V(G)|$ by arranging them according to their leftmost endpoints.

Theorem 8 CONFLICT-FREE SET COVER *with duplicate sets when $G_{\mathcal{F}}$ is restricted to interval graphs can be solved in $\mathcal{O}^*(2^{|\mathcal{U}|})$ time.*

Proof Let the sets of $\mathcal{F} = \{S_1, \ldots S_m\}$ be ordered in the reverse order of the ordering described above. For each subset $W \subseteq \mathcal{U}$, and $i \in [m]$, define $DP[W, i]$ as the size of the minimum set $X \subseteq \{S_1, \ldots S_i\}$ such that X covers W and vertices of X are independent in $G_{\mathcal{F}}$. Initially, set $DP[\emptyset, 0] = 0$ and $DP[X, 0] = \infty$ when $X \neq \emptyset$. We have the following recursive formula for $DP[W, i]$.

$$DP[W, i] = \min \left\{ 1 + DP[W \setminus S_i, \ell], DP[W, i - 1] \right\} \qquad (1)$$

where ℓ is the index of the rightmost non-neighbor of S_i in $G_{\mathcal{F}}[\{S_1, \ldots, S_{i-1}\}]$.

The correctness proof of the above equation is as follows.

Let X be the optimal solution for $DP[W, i]$. The subfamily X either contains the set S_i or it does not. When X does not contain S_i, then it is a valid candidate for $DP[W, i - 1]$ and hence $|X| \geq DP[W, i - 1]$. When it contains S_i, $X \setminus \{S_i\}$ is a valid candidate for $DP[W \setminus S_i, \ell]$ and hence $|X| - 1 \geq DP[W \setminus S_i, \ell]$. Hence $DP[W, i] \geq \min \left\{ 1 + DP[W \setminus S_i, \ell], DP[W, i - 1] \right\}$.

Let Y be the optimal solution for $DP[W \setminus S_i, \ell]$. Then $Y \cup S_i$ is a valid candidate for $DP[W, i]$ since $\{S_1, \ldots S_\ell\}$ contains only non-neighbors of S_i as all the neighbors of S_i follows after the rightmost non-neighbor of S_i which is S_ℓ. Hence $DP[W, i] \leq 1 + DP[W \setminus S_i, l]$. Let Z be the optimal solution for $DP[W, i - 1]$. Then Z is also a valid candidate for $DP[W, i]$. Hence $DP[W, i] \leq DP[W, i - 1]$.

The entry $DP[W, m]$ contains the size of the minimum-sized solution of CONFLICT-FREE SET COVER . The number of subproblems is $\sum_{j \in [|\mathcal{U}|]} \binom{|\mathcal{U}|}{j} \cdot m$ and at each subproblem $\mathcal{O}(m)$ time is spent to find ℓ. Hence the running time is $\sum_{j \in [|\mathcal{U}|]} \binom{|\mathcal{U}|}{j} \cdot \mathcal{O}(m) = \mathcal{O}^*(2^{|\mathcal{U}|})$.   □

Now we give a $\mathcal{O}^*(3^{|\mathcal{U}|})$-time dynamic programming algorithm for chordal graphs which is a superclass of interval graphs.

4.2.2 Chordal Graphs

A *clique tree decomposition* is a tree decomposition T where for all nodes $i \in V(T)$, the vertices of in the bag X_i are such that $G[X_i]$ forms a clique. All chordal graphs have clique tree decompositions that can be found in polynomial time [16]. Given a clique tree decomposition, it can be converted to a nice clique tree decomposition in polynomial time using Lemma 1. Note from Lemma 1 that every bag of the new nice tree decomposition is a subset of some bag of the original tree decomposition. Hence every bags in the nice tree decompositions are also cliques.

In the theorem below, we give an algorithm for CONFLICT-FREE SET COVER with duplicates on chordal graphs using dynamic programming on the nice clique tree decomposition of the graph.

Theorem 9 CONFLICT-FREE SET COVER *with duplicates on chordal graphs can be solved in* $O^*(3^{|\mathcal{U}|})$ *running time.*

Proof For the instance $(\mathcal{U}, \mathcal{F}, G_{\mathcal{F}}, k)$ of CONFLICT-FREE SET COVER , let T be the tree of the nice clique tree decomposition of the chordal graph $G_{\mathcal{F}}$. For a node $i \in V(T)$, let T_i denote the subtree rooted at node i, V_i denote the vertices of G in the bags of nodes of T_i and X_i denote the vertices in the bag of node i. Note that since we are looking for a solution that is also independent set in the chordal graph, from each bag no more than one vertex can be in the solution as $G[X_i]$ forms a clique.

For each subset $W \subseteq \mathcal{U}$, node $i \in V(T)$ and $x \in X_i$, let $DP[W, i, x]$ denote the size of the minimum-sized independent set Y of the graph $G[V_i]$ covering W such that $x \in Y$. Node x can take empty value $\emptyset$ as well to denote no vertex is picked from the bag X_i. Initially, set all entries to ∞ denoting that no such solution exists. We have the following recurrence relations for each type of node in T to compute $DP[W, i, x]$:

- Leaf Node:
$$DP[W, i, \emptyset] = \begin{cases} 0 \text{ if } W = \emptyset, \\ \infty \text{ otherwise} \end{cases}$$

- Introduce Node: Let i the the parent of node j and vertex v is introduced in X_i.
$$DP[W, i, x] = \begin{cases} DP[W, j, x] \text{ if } x \neq v \\ 1 + DP[W \setminus S_v, j, \emptyset] \text{ when } x = v \end{cases}$$

- Forget Node: Let i the the parent of node j and vertex v is forgotten in X_i.
$$DP[W, i, x] = \begin{cases} DP[W, j, x] \text{ if } x \neq \emptyset \\ \min\{DP[W, j, \emptyset], DP[W, j, v]\} \text{ when } x = \emptyset \end{cases}$$

- Join Node: Let i be the parent of two nodes j and j' and $X_i = X_j = X'_j$.

$$DP[W, i, x] = \begin{cases} \min_{W_1 \subseteq W} \{DP[W_1, j, x] + DP[W \setminus W_1, j', x] - 1\} \text{ if } x \neq \emptyset \\ \min_{W_1 \subseteq W} \{DP[W_1, j, \emptyset] + DP[W \setminus W_1, j', \emptyset]\} \text{ when } x = \emptyset \end{cases}$$

The entry $DP[\mathcal{U}, r, \emptyset]$ contains the size of the minimum-sized solution of CONFLICT-FREE SET COVER where r is the root of the tree. The number of subproblems is $\mathcal{O}(\sum_{j=1}^{|\mathcal{U}|} \binom{|\mathcal{U}|}{j} \cdot |T| \cdot |\mathcal{F}|)$. The maximum time spent on computing $DP[W, i, x]$ where $|W| = j$ is $\mathcal{O}(2^j)$ for going over all subsets W_1 at the join node. Hence the overall running time is $\mathcal{O}^*(\sum_{j=1}^{|\mathcal{U}|} \binom{|\mathcal{U}|}{j} \cdot 2^j) = \mathcal{O}^*(3^{|\mathcal{U}|})$.

Correctness of Recurrence Relations For ease of writing, let us denote the terms present in the left hand side of the equation as LHS and in the right hand side of the equation as RHS. For each recurrence relations defined above, we prove its correctness by showing inequality in both sides. We use the term *optimal* solution for a DP entry to denote the minimum-sized conflict-free set cover corresponding to the entry and *candidate* solution for a DP entry to denote any conflict-free set cover corresponding to the entry (need not be of minimum size).

- Introduce Node:

    Let X be the optimal solution for the entry $DP[W, i, x]$. By definition $x \in X$. If $x \neq v$, X is also a candidate solution for $DP[W, j, x]$ as $V_i \setminus \{v\} = V_j$. If $x = v$, $X \setminus \{v\}$ is a candidate solution for $DP[W, j, \emptyset]$ as no $y \in X_i$, $y \neq v$ can be in $X \setminus \{v\}$ since $G[X_i]$ is a clique. In either case, the value at RHS can be either the size of LHS or even lower. Hence, $LHS \geq RHS$.

    Let Y be the optimal solution for $DP[W, j, x]$. If $x \neq \emptyset$, the set Y is also a candidate solution for $DP[W, i, x]$. Hence $LHS \leq RHS$. If $x = \emptyset$, look at Z, the solution for $DP[W \setminus S_v, j, x]$. Since all the edges of v in the graph $G[V_i]$ is in bag X_i, $Z \cup \{v\}$ is also an independent set and it covers W. Hence both Y and Z are candidate solutions for $DP[W, i, x]$ when $x = \emptyset$. Hence $LHS \leq RHS$.

- Forget Node:

    Let X be the optimal solution for $DP[W, i, x]$ such that $X \cap X_i = \{x\}$ with $x \neq \emptyset$. Since $V_j = V_i$, X is also a solution for $G[V_j]$ such that $X \cap X_i = \{x\}$ and hence a candidate solution for $DP[W, j, x]$. Hence $LHS \geq RHS$. Similarly we can prove the inequality in the other direction.

    When $x = \emptyset$, let X be the optimal solution for $DP[W, i, x]$ such that $X \cap X_i = \emptyset$. Since $V_j = V_i$, X is also a solution for $G[V_j]$ such that $X \cap X_i = \emptyset$ and hence a candidate solution for $DP[W, j, x]$. Also if $v \in X$, X is a candidate solution for $DP[W, j, v]$ as well. If $v \in X$, $DP[W, j, \emptyset] \geq DP[W, j, v]$. Hence $LHS \geq RHS$.

    Let Y be the optimal solution for the minimum of two entries $DP[W, j, \emptyset]$ and $DP[W, j, v]$. If the minimum is $DP[W, j, v]$, then $Y \setminus \{v\}$ is a candidate solution of $DP[W, i, \emptyset]$. Else Y is also a candidate solution of $DP[W, i, \emptyset]$. Hence $LHS \leq RHS$.

- Join Node:

    Let X be the optimal solution for the entry $DP[W, i, x]$. When $x \neq \emptyset$ and $G[X_i]$ forms a clique, $X \cap X_i = \{x\}$. Let W_1 and W_2 be the subset of elements covered by $Y_j = X \cap V_j$ and $Y_{j'} = X \cap V_{j'}$ respectively. Note that since X covers W, $W_1 \cup W_2 = W$. Since X is an independent set, Y_j and $Y_{j'}$ are independent sets as well as they both are subsets of X. Note that $Y_j \cap Y_{j'} = \{x\}$. Hence Y_j and $Y_{j'}$ respectively are candidate solutions to entries $DP[W_1, j, x]$ and $DP[W \setminus W_1, j', x]$ as $W \setminus W_1 \subseteq W_2$. Since x is the only entry common to both of them, we have $LHS \geq RHS$.

    Let Z_j and $Z_{j'}$ be the optimal solutions for the entries $DP[W_1, j, x]$ and $DP[W_2, j', x]$ where $W_2 = W \setminus W_1$. Since $X_i = X_j = X'_j$ and $G[X_i]$ forms a clique, $Z_j \cap X_j = Z_{j'} \cap X_j = \{x\}$. Look at the set $Z = Z_j \cup Z_{j'}$. The set Z is an independent set since Z_j and $Z_{j'}$ are independent sets and since there are no

edges across $G[V_j \setminus X_i]$ and $G[V_{j'} \setminus X_i]$ by the definition of tree decomposition. Hence Z is a candidate solution for the entry $DP[W, i, x]$ of size $|Z_j| + |Z_{j'}| - 1$. Therefore $LHS \leq RHS$.

When $x = \emptyset$, using similar arguments we can prove that $LHS = RHS$. □

4.3 c-Intersection Conflict-Free Set Cover Parameterized by k

When $G_{\mathcal{F}}$ is a chordal graph, we could not come up with a meta-theorem like we had earlier in Section 4.1 for split graphs, nowhere dense graphs etc which gave an FPT algorithm for CONFLICT-FREE SET COVER given that the restricted version of SET COVER has an FPT algorithm. Hence we focus on a particular restriction of SET COVER known to be FPT which is c-INTERSECTION SET COVER and give an FPT algorithm for the conflict-free version. Note that on the contrary, Theorem 1 shows that the problem is W[1]-hard when $G_{\mathcal{F}}$ is bipartite even when $c = 1$.

Given the instance $(\mathcal{U}, \mathcal{F}, G_{\mathcal{F}}, k)$. We start the algorithm with the following reduction rule.

Reduction Rule 1 *If there is a set $S \in \mathcal{F}$ such that $|S| > ck$, then put S in the solution and drop k by 1. The new instance is $(\mathcal{U}', \mathcal{F}', G'_{\mathcal{F}}, k - 1)$ where $\mathcal{U}' = \mathcal{U} \setminus S$, $\mathcal{F}' = \mathcal{F} \setminus N[S]$ and $G'_{\mathcal{F}} = G_{\mathcal{F}}[\mathcal{F}']$.*

Claim 1 *Reduction Rule 1 is safe.*

Proof Let $I' = (\mathcal{U}', \mathcal{F}', G'_{\mathcal{F}}, k')$ be the instance of c-INTERSECTION CONFLICT-FREE SET COVER after applying Reduction Rule 1 to instance $I = (\mathcal{U}, \mathcal{F}, G_{\mathcal{F}}, k)$ for a set $S \in \mathcal{F}$. We show that I is a YES-INSTANCE if and only if I' is a YES-INSTANCE.

Let $\mathcal{X} \subseteq \mathcal{F}$ be a solution of size at most k. We claim that $S \in \mathcal{X}$. Suppose not. The elements of S has to be covered by the other sets in $\mathcal{F}$. We know that for any set $S' \in \mathcal{F}$, $|S' \cap S| \leq c$. Since $|\mathcal{X}| \leq k$, $\mathcal{X}$ can cover only at most ck elements of S. Since $|S| > ck$, $\mathcal{X}$ do cover the set S giving a contradiction.

We claim that the set $\mathcal{X}' = \mathcal{X} \setminus S$ is a solution of size at most $k' \leq k - 1$ to the instance I'. Suppose not. Note that all the sets in $\mathcal{X}'$ are present in $\mathcal{F}' = \mathcal{F} \setminus N[S]$ as they cannot be present in $N[S]$ which would contradict the fact that $\mathcal{X}$ is an independent set in $G_{\mathcal{F}}$. Since $\mathcal{X}$ covers $\mathcal{U}$, $\mathcal{X}'$ covers $\mathcal{U}' = \mathcal{U} \setminus S$. Also since $\mathcal{X}$ is an independent set in $G_{\mathcal{F}}$, $\mathcal{X}'$ is an independent set in $G'_{\mathcal{F}}$. Hence I' is a YES-INSTANCE.

Conversely, let Y' be a solution of size k' to the instance I'. We claim that $Y = Y' \cup S$ is a solution of size at most k to the instance I. Since Y' is an independent set in $G'_{\mathcal{F}}$ and $\mathcal{F}' = \mathcal{F} \setminus N[S]$, Y is an independent set in $G_{\mathcal{F}}$. Also since $\mathcal{U}' = \mathcal{U} \setminus S$, Y covers $\mathcal{U}$. □

We apply Reduction Rule 1 exhaustively. Note that by applying Reduction Rule 1, we introduce duplicate sets. Afterwards, we can assume that the size of every set in $\mathcal{F}$ has size at most ck. We now apply the following reduction rule.

Reduction Rule 2 *If $|\mathcal{U}| > ck^2$, return NO.*

Since every set in $\mathcal{F}$ has size at most ck, a solution of size at most k can cover at most ck^2 elements. Hence if $|\mathcal{U}| > ck^2$, there is no solution of size k and hence we return NO.

After applying Reduction Rules 1 and 2 exhaustively in order, we get an instance where the universe size $|\mathcal{U}| \leq ck^2$, a function of k. Hence the problem can now be treated as an instance of CONFLICT-FREE SET COVER parameterized by $|\mathcal{U}|$. We use Theorem 9 to get an FPT algorithm with running time $\mathcal{O}^*(3^{|\mathcal{U}|}) = \mathcal{O}^*(3^{ck^2})$. Hence we have the following theorem.

Theorem 10 c-INTERSECTION CONFLICT-FREE SET COVER *when $G_{\mathcal{F}}$ is a chordal graph has an algorithm with a running time of $\mathcal{O}^*(3^{ck^2})$.*

4.3.1 Polynomial Kernel in Cluster Graphs

In this section, we show a polynomial kernel for cluster graphs which is a subclass of chordal graphs.

We initially apply Reduction Rules 1 and 2 exhaustively in order. Hence we can assume that the universe size is at most ck^2.

We first claim that there are only $\binom{ck^2}{c+1}$ distinct sets present in $\mathcal{F}$. Let us look at an arbitrary subset A of $c + 1$ elements from $\mathcal{U}$. There is only one set $S \in \mathcal{F}$ such that $A \subseteq S$. Suppose there also exist $S' \in \mathcal{F}, S' \neq S$ such that $A \subseteq S'$. Then we have $A \subseteq S \cap S'$. Hence $|S' \cap S| > c$ giving a contradiction.

Hence we can create an injective map from each distinct set in $\mathcal{F}$ of size at least $c + 1$ to a subset of $c + 1$ elements of $\mathcal{U}$. Since there are at most $\binom{|\mathcal{U}|}{c+1}$ such subsets, there are at most $\binom{ck^2}{c+1}$ distinct elements in $\mathcal{F}$ of size at least $c + 1$. The number of distinct sets in $\mathcal{F}$ of size at most c is also bounded by $\binom{|\mathcal{U}|}{c+1} \leq \binom{ck^2}{c+1}$. Hence we can conclude that the total number of distinct elements in $\mathcal{F}$ is at most $\binom{ck^2}{c+1}$.

Hence to bound the size of $\mathcal{F}$, we only need to bound the number of duplicates in $\mathcal{F}$.

Let $C_1, C_2, \ldots, C_p$ the components in the cluster graph $G_{\mathcal{F}}$, each component being a clique. We have the following reduction rule.

Reduction Rule 3 *If a component C_i where $i \in [p]$ has two vertices v and v' where the set corresponding to both vertices is the same set S, delete v' from $\mathcal{F}$.*

Both the vertices v and v' cover the same set S and have the same closed neighborhood set which is the the entire clique. Since a solution will contain only at most one vertex from C_i as it is a clique, the Reduction Rule 3 is safe.

We apply Reduction Rule 3 to all components C_i for $i \in p$. Since all the sets in C_i are distinct afterwards, we have $|C_i| \leq \binom{ck^2}{c+1}$.

We have the following reduction rule to take care of duplicate sets among different components C_i.

Reduction Rule 4 *For each distinct set $S \in \mathcal{F}$, if there are more than k vertices whose corresponding set is S, keep arbitrarily selected $k + 1$ vertices whose set is S and delete the rest of the vertices.*

After applying this rule, we can conclude that every set in $\mathcal{F}$ has at most k duplicates.

Claim 2 *Reduction Rule 4 is safe on instances of c-INTERSECTION CONFLICT-FREE SET COVER where $G_{\mathcal{F}}$ is a cluster graph.*

Proof Let $I' = (\mathcal{U}', \mathcal{F}', G'_{\mathcal{F}}, k')$ be the instance of c-INTERSECTION CONFLICT-FREE SET COVER after applying Reduction Rule 4 to instance $I = (\mathcal{U}, \mathcal{F}, G_{\mathcal{F}}, k)$ for a set $S \in \mathcal{F}$. We show that I is a YES-INSTANCE if and only if I' is a YES-INSTANCE.

Let X be a solution of size at most k for I. We construct a subset of vertices X' in the instance I' as follows. The set X' is initially empty. In phase 1, for each vertex $v \in X$, if the number of duplicates of the corresponding set S_v is not more than k in I, then v is also present in I'. Add v to X'. Mark v and the corresponding component containing v. If the number of duplicates of S_v is more than k in I, we do nothing.

After we do this for every vertex in X, phase 2 begins. Every unmarked vertex $v \in X$ has more than k duplicates in I. For each such vertex v we add a vertex w to X' from an unmarked component whose corresponding set is S_v. Mark the corresponding component containing w.

Note that the procedure to construct X' terminates without fail. This is because there is an unmarked component containing vertex S_v at every step where a vertex is added since we keep $k + 1$ duplicates for v which is present in different components of $G_{\mathcal{F}}$.

We claim that X' is a solution for the instance I'. Clearly X' covers $\mathcal{U}$ as the sets corresponding to each vertices in X' remains the same as X. Also since at each time, a vertex in X' is added from an unmarked cluster, X' also forms an independent set.

Conversely a solution Y for I' is also a solution for I as all the vertices of Y are also present in I. □

After applying reduction rules 1 to 4 exhaustively in order, it is easy to see that we get a kernel for c-INTERSECTION CONFLICT-FREE SET COVER with universe size ck^2 and family size $(k + 1) \cdot \binom{ck^2}{c+1}$. We have the following theorem.

Theorem 11 *c-INTERSECTION CONFLICT-FREE SET COVER parameterized by k when $G_{\mathcal{F}}$ is a cluster graph has a kernel with universe size ck^2 and family size $(k + 1) \cdot \binom{ck^2}{c+1}$.*

Using the kernel, we get a better FPT algorithm when $G_{\mathcal{F}}$ is a cluster graph by going over all the k-sized subsets of $\mathcal{F}$.

Corollary 2 *c*-INTERSECTION CONFLICT-FREE SET COVER *parameterized by k when $G_{\mathcal{F}}$ is a cluster graph has an FPT algorithm with running time* $\mathcal{O}^*\left(\left((k+1) \cdot \binom{ck^2}{c+1}\right)^k\right)) = \mathcal{O}^*(k^{\mathcal{O}(ck)}).$

5 Matroidal Conflict-Free Set Cover

In this section, we study the MATROIDAL CONFLICT-FREE SET COVER problem where the conflicting condition is being an independent set in a (representable) matroid.

Let $\mathbb{F}_{p^\ell}$ denote a finite field of order p^ℓ where p is a prime and ℓ is a positive integer. Also we denote by $\mathbb{Q}$ the field of rationals. Let us first define the MATROIDAL CONFLICT-FREE SET COVER problem as follows.

MATROIDAL CONFLICT-FREE SET COVER
Input: A universe $\mathcal{U}$, a family $\mathcal{F}$ of subsets of $\mathcal{U}$, a linear representation of a matroid $M = (\mathcal{F}, \mathcal{I})$ over a field $\mathbb{F}$ where $\mathbb{F} = \mathbb{F}_{p^\ell}$ or $\mathbb{F}$ is $\mathbb{Q}$ and an integer k.
Goal: Is there a subfamily $\mathcal{F}' \subseteq \mathcal{F}$ of size at most k such that $\bigcup_{F \in \mathcal{F}'} F = \mathcal{U}$ and $\mathcal{F}'$ forms an independent set in M?

Note that we need the linear representation of the matroid M over a field $\mathbb{F}$ where $\mathbb{F} = \mathbb{F}_{p^\ell}$ or $\mathbb{F}$ is $\mathbb{Q}$. This is due to technical reasons which will be revealed later.

We give a dynamic programming algorithm for MATROIDAL CONFLICT-FREE SET COVER containing duplicate sets using computation of representative sets noting that the similar ideas used in [6] for INTERVAL COVERING can be extended to MATROIDAL CONFLICT-FREE SET COVER .

For $W \subseteq \mathcal{U}$, let $\mathcal{B}^W$ denote the collection of subfamilies X of $\mathcal{F}$ of size at most k such that X covers W and forms an independent set in the matroid M.

$$\mathcal{B}^W = \{X \subseteq \mathcal{F} \;\Big|\; |X| \le k, W \subseteq \bigcup_{S \in X} S \text{ and } X \in \mathcal{I}\}$$

Note that $\mathcal{B}^{\mathcal{U}}$ contains all the solutions of size at most k of MATROIDAL CONFLICT-FREE SET COVER . Hence we solve the MATROIDAL CONFLICT-FREE SET COVER problem by checking whether $\mathcal{B}^{\mathcal{U}}$ is empty or not.

Definition 12 (q-representative family [27]) Let $M = (E, \mathcal{I})$ be a matroid and $\mathcal{A}$ be a family of sets of size p in M. For sets $A, B \subseteq E$, we say that A *fits* B if $A \cap B = \emptyset$ and $A \cup B \in \mathcal{I}$. A subfamily $\hat{\mathcal{A}} \subseteq \mathcal{A}$ is said to q-represent $\mathcal{A}$ if for every set B of size q such that there is an $A \in \mathcal{A}$ that fits B, there is an $\hat{A} \in \hat{\mathcal{A}}$ that also fits B. We use $\hat{\mathcal{A}} \subseteq_{rep}^q \mathcal{A}$ to denote that $\hat{\mathcal{A}}$ q-represents $\mathcal{A}$.

Lemma 3 (Fomin et al. [15]) *For a matroid $M = (E, \mathcal{I})$ and $\mathcal{S} \subseteq E$, if $\mathcal{S}_1 \subseteq_{rep}^q \mathcal{S}$ and $\mathcal{S}_2 \subseteq_{rep}^q \mathcal{S}_1$, then $\mathcal{S}_2 \subseteq_{rep}^q \mathcal{S}$.*

Note that $\mathcal{B}^{\mathcal{U}}$ is nonempty if and only if $\hat{\mathcal{B}}^{\mathcal{U}} \subseteq_{rep}^0 \mathcal{B}^{\mathcal{U}}$ is nonempty. Let us define $\mathcal{B}^{Wj}$ as the subset of $\mathcal{B}^W$ containing sets of size exactly j. We use $\hat{\mathcal{B}}^W \subseteq_{rep}^{1,...,k} \mathcal{B}^W$ to denote that $\hat{\mathcal{B}}^W$ contains the union of all the i-representative families of $\mathcal{B}^W$ where $1 \le i \le k$. In other words,

$$\hat{\mathcal{B}}^W = \bigcup_{j=1}^{k} \left(\hat{\mathcal{B}}^{Wj} \subseteq_{rep}^{k-j} \mathcal{B}^{Wj} \right)$$

Lemma 4 (Lokshtanov et al. [25]) *Let $M = (E, \mathcal{I})$ be a linear matroid of rank n and S be a family of t independent sets of size p. Let A be a $n \times |E|$ matrix representation of M over a field $\mathbb{F}$ where $\mathbb{F} = \mathbb{F}_{p^\ell}$ or $\mathbb{F}$ is $\mathbb{Q}$. Then there is a deterministic algorithm to compute $\hat{S} \subseteq_{rep}^q S$ of size $np\binom{p+q}{p}$ in $\mathcal{O}\left(\binom{p+q}{p} tp^3 n^2 + t\binom{p+q}{p}^{\omega-1}(pn)^{\omega-1} \right) + (n + |E|)^{O(1)})$ operations over $\mathbb{F}$ where ω is the matrix multiplication exponent.*

Note that Lemma 4 is applicable only when the matroid is represented over a field $\mathbb{F}$ where $\mathbb{F} = \mathbb{F}_{p^\ell}$ or $\mathbb{F}$ is $\mathbb{Q}$. This is why we imposed a similar restriction for the matroid representation in the definition of MATROIDAL CONFLICT-FREE SET COVER .

Theorem 12 MATROIDAL CONFLICT-FREE SET COVER *can be solved in $\mathcal{O}^*(2^{(\omega+1)\cdot|\mathcal{U}|})$ time where ω is the matrix multiplication exponent.*

Proof Let $\mathcal{D}$ be an array of size $2^{|\mathcal{U}|}$ with $\mathcal{D}[W]$ storing the family $\hat{\mathcal{B}}^W \subseteq_{rep}^{1,...,k} \mathcal{B}^W$. We compute the entries of $\mathcal{D}$ in the increasing order of subsets of $\mathcal{U}$. To do so we compute the following:

$$\mathcal{N}^W = \bigcup_{S_i \in \mathcal{F}} (\mathcal{D}[W \setminus S_i] \bullet S_i) \cap \mathcal{I} \tag{2}$$

where $\mathcal{A} \bullet \mathcal{B} = \{A \cup B \mid A \in \mathcal{A} \text{ and } B \in \mathcal{B} \text{ and } A \cap B = \emptyset\}$.

We show that $\mathcal{N}^W \subseteq_{rep}^{1,...,k} \mathcal{B}^W$. Let $S \in \mathcal{B}^{Wj}$ and Y be a set of size $k - j$ such that $S \cap Y = \emptyset$ and $S \cup Y \in \mathcal{I}$. We give a set $\hat{S} \in \mathcal{N}^{Wj}$ such that $\hat{S} \cap Y = \emptyset$ and $\hat{S} \cup Y \in \mathcal{I}$.

Let $S = \{S_1, S_2, \ldots, S_j\}$. Let $S' = S \setminus \{S_j\}$. Let $Y' = Y \cup \{S_j\}$. Then, $|S'| = j - 1$ and $|Y'| = k - j + 1$. Since S' covers $W \setminus S_j$, $S' \in \mathcal{B}^{(W \setminus S_j)(j-1)}$. By definition, $D[W \setminus S_j]$ contains $\hat{\mathcal{B}}^{(W \setminus S_j)(j-1)} \subseteq_{rep}^{k-j+1} \mathcal{B}^{(W \setminus S_j)(j-1)}$ and hence a set $S^* \in D[W \setminus S_j]$ such that $S^* \cap Y' = \emptyset$ and $S^* \cup Y' \in \mathcal{I}$. From (2), $S^* \cup \{S_j\} \in \mathcal{N}^W$. The set $\hat{S} = S^* \cup \{S_j\}$ is such that $\hat{S} \cap Y = \emptyset$ and $\hat{S} \cup Y \in \mathcal{I}$. Hence $\mathcal{N}^W \subseteq_{rep}^{1,...,k} \mathcal{B}^W$.

We store $\hat{\mathcal{N}}^W \subseteq_{rep}^{1,...,k} \mathcal{N}^W$ in $\mathcal{D}[W]$. The sets $\hat{\mathcal{N}}^{Wj}$ are computed using Lemma 4. We have $\hat{\mathcal{N}}^{Wj} \subseteq_{rep}^{k-j} \mathcal{N}^{Wj} \subseteq_{rep}^{k-j} \mathcal{B}^{Wj}$ for all $1 \le j \le k$. Hence from Lemma 3, we have $\mathcal{D}[W] = \hat{\mathcal{N}}^W \subseteq_{rep}^{1,...,k} \mathcal{B}^W$.

We now focus on the running time to compute $\mathcal{D}[W]$ and the size of $\mathcal{D}[W]$. Assume that $\mathcal{D}[Y]$ is precomputed for all subsets $Y \subseteq W$. We have $|\mathcal{D}[Y]| = |\hat{\mathcal{N}}^Y| = \sum_{j=1}^{k} |\hat{\mathcal{N}}^{Yj}|$. From Lemma 4, $|\hat{\mathcal{N}}^{Yj}| \le |\mathcal{F}| \cdot k \cdot \binom{k}{j}$. Hence from (2), putting

$Y = W \setminus S_i$, we have $|\mathcal{N}^{Wj}| \leq |\mathcal{F}|^2 \cdot k \cdot \binom{k}{j}$. Using Lemma 4, the time to compute $\hat{\mathcal{N}}^{Wj} \subseteq_{rep}^{k-j} \mathcal{N}^{Wj}$ is $\mathcal{O}^*\left(\binom{k}{j}^2 + \binom{k}{j}^\omega\right)$ where ω is the exponent for matrix multiplication. Hence the total time to compute $\mathcal{D}[W]$ is $\sum_{j=1}^{k} \mathcal{O}^*(\binom{k}{j}^\omega) = \mathcal{O}^*(2^{\omega k})$. The size of $\mathcal{D}[W]$ is $\mathcal{O}(|\mathcal{F}| \cdot k \cdot \sum_{j=1}^{k} \binom{k}{j}) = \mathcal{O}(2^k \cdot k \cdot |\mathcal{F}|)$.

The overall running time to check if $\mathcal{D}[U]$ is empty or not is bounded by $\mathcal{O}^*(2^{|\mathcal{U}|} \cdot 2^{\omega k}) = \mathcal{O}^*(2^{\omega|\mathcal{U}|+|\mathcal{U}|}) = \mathcal{O}^*(10.361^{|\mathcal{U}|})$.   □

We note that an FPT algorithm for MATROIDAL CONFLICT-FREE SET COVER parameterized by k can be obtained as a corollary of a result by Bevern et al. [34]. The authors give an algorithm for a generalization of SET COVER called uncapacitated facility location problem with multiple matroid constraints. This algorithm also uses the idea of representative families that we use. But the algorithm involves further sophistications as they work on a general problem. The running time for MATROIDAL CONFLICT-FREE SET COVER from Bevern et al is $2^{O(r \log r)} n^2$ where r is the rank of the matroid. The rank r is bounded by the universe size $|\mathcal{U}|$. The running time is $2^{O(|\mathcal{U}| \log |\mathcal{U}|)} n^2$ when $r = \mathcal{O}(|\mathcal{U}|)$ in which case our algorithm from Theorem 12 with running time $\mathcal{O}^*(2^{(\omega+1) \cdot |\mathcal{U}|})$ is better. But when the rank is smaller, the algorithm by Bevern et al. is better.

6 Conclusion

We have initiated a systematic study of CONFLICT-FREE SET COVER with various parameterizations and restrictions to $G_\mathcal{F}$. When parameterized by the solution size k and when the restricted SET COVER variant is FPT parameterized by k, we have shown W[1]-hardness for the corresponding CONFLICT-FREE SET COVER variant when the conflict graph $G_\mathcal{F}$ is bipartite and gave FPT algorithms when $G_\mathcal{F}$ is nowhere dense or has bounded number of independent sets. When parameterized by the universe size (hence SET COVER variant is FPT), we have shown W[1]-hardness when $G_\mathcal{F}$ is bipartite and gave FPT algorithms when $G_\mathcal{F}$ is chordal, nowhere dense or has bounded number of independent sets. One open question is to identify a general characterization for the graph classes of $G_\mathcal{F}$ when CONFLICT-FREE SET COVER becomes FPT for the above two cases.

We gave an FPT algorithm for c-INTERSECTION CONFLICT-FREE SET COVER when $G_\mathcal{F}$ is a chordal graph but only managed to find a polynomial kernel for cluster graphs, a subgraph of chordal graphs. Finding a polynomial kernel for c-INTERSECTION CONFLICT-FREE SET COVER when $G_\mathcal{F}$ is a chordal graph remains open.

References

1. Agrawal, A., Jain, P., Kanesh, L., Lokshtanov, D., Saurabh, S.: Conflict free feedback vertex set: a parameterized dichotomy. In: 43rd International Symposium on Mathematical Foundations of Computer Science (MFCS 2018). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2018)

2. Arkin, E.M., Banik, A., Carmi, P., Citovsky, G., Katz, M.J., Mitchell, J.S., Simakov, M.: Choice is hard. In: International Symposium on Algorithms and Computation, pp. 318–328. Springer (2015)

3. Arkin, E.M., Banik, A., Carmi, P., Citovsky, G., Katz, M.J., Mitchell, J.S., Simakov, M.: Conflict-free covering. In: Conference on Computational Geometry, p. 17 (2015)

4. Arkin, E.M., Hassin, R.: Minimum-diameter covering problems. Networks: An International Journal **36**(3), 147–155 (2000)

5. Banik, A., Panolan, F., Raman, V., Sahlot, V.: Fréchet distance between a line and avatar point set. Algorithmica **80**(9), 2616–2636 (2018)

6. Banik, A., Panolan, F., Raman, V., Sahlot, V., Saurabh, S.: Parameterized complexity of geometric covering problems having conflicts. Algorithmica **82**(1), 1–19 (2020)

7. Cygan, M., Dell, H., Lokshtanov, D., Marx, D., Nederlof, J., Okamoto, Y., Paturi, R., Saurabh, S., Wahlström, M.: On problems as hard as CNF-SAT. ACM Transactions on Algorithms (TALG) **12**(3), 41 (2016)

8. Cygan, M., Fomin, F.V., Kowalik, Ł., Lokshtanov, D.., Marx, D.., Pilipczuk, M.., Pilipczuk, M.., Saurabh, S..: Parameterized Algorithms. Springer, Berlin (2015)

9. Darmann, A., Pferschy, U., Schauer, J., Woeginger, G.J.: Paths, trees and matchings under disjunctive constraints. Discret. Appl. Math. **159**(16), 1726–1735 (2011)

10. Diestel, R.: Graph Theory. Springer, Berlin (2006)

11. Epstein, L., Favrholdt, L.M., Levin, A.: Online variable-sized bin packing with conflicts. Discret. Optim. **8**(2), 333–343 (2011)

12. Even, G., Halldórsson, M.M., Kaplan, L., Ron, D.: Scheduling with conflicts: online and offline algorithms. Journal of Scheduling **12**(2), 199–224 (2009)

13. Farber, M.: On diameters and radii of bridged graphs. Discret. Math. **73**(3), 249–260 (1989)

14. Fomin, F.V., Kratsch, D., Woeginger, G.J.: Exact (exponential) algorithms for the dominating set problem. In: International Workshop on Graph-Theoretic Concepts in Computer Science, pp. 245–256. Springer (2004)

15. Fomin, F.V., Lokshtanov, D., Panolan, F., Saurabh, S.: Efficient computation of representative families with applications in parameterized and exact algorithms. Journal of the ACM (JACM) **63**(4), 29 (2016)

16. Golumbic, M.C.: Algorithmic graph theory and perfect graphs. Annals of Discrete Mathematics **57** (2004)

17. Impagliazzo, R., Paturi, R., Zane, F.: Which problems have strongly exponential complexity? Journal of Computer and System Sciences (JCSS) **63**(4), 512–530 (2001)

18. Jacob, A., Majumdar, D., Raman, V.: Parameterized complexity of conflict-free set cover. In: International Computer Science Symposium in Russia, pp. 191–202. Springer (2019)

19. Jain, P., Kanesh, L., Misra, P.: Conflict Free Version of Covering Problems on Graphs: Classical and Parameterized. Theory Comput. Syst. **64**(6), 1067–1093 (2020). https://doi.org/10.1007/s00224-019-09964-6

20. Johnson, D.S., Yannakakis, M., Papadimitriou, C.H.: On generating all maximal independent sets. Inf. Process. Lett. **27**(3), 119–123 (1988)

21. Kann, V.: Polynomially bounded minimization problems which are hard to approximate. In: International Colloquium on Automata, Languages, and Programming, pp. 52–63. Springer (1993)

22. Kloks, T.: Treewidth: Computations and Approximations, vol. 842. Springer Science & Business Media, Berlin (1994)

23. Kumar, V.S.A., Arya, S., Ramesh, H.: Hardness of set cover with intersection 1. In: Rolim, J.D.P., Welzl, E. (eds.) Automata, Languages and Programming, pp. 624–635. Springer, Berlin (2000)

24. Lokshtanov, D., Marx, D., Saurabh, S.: Slightly superexponential parameterized problems. SIAM J. Comput. **47**(3), 675–702 (2018)

25. Lokshtanov, D., Misra, P., Panolan, F., Saurabh, S.: Deterministic truncation of linear matroids. ACM Transactions on Algorithms (TALG) **14**(2), 14 (2018)

26. Lokshtanov, D., Panolan, F., Saurabh, S., Sharma, R., Zehavi, M.: Covering small independent sets and separators with applications to parameterized algorithms. In: Proceedings of the Twenty-Ninth

Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 2785–2800. Society for Industrial and Applied Mathematics (2018)

27. Marx, D.: A parameterized view on matroid optimization problems. Theor. Comput. Sci. **410**(44), 4471–4479 (2009)

28. Marx, D., Salmasi, A., Sidiropoulos, A.: Constant-factor approximations for asymmetric TSP on nearly-embeddable graphs. In: Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2016, vol. 60 of LIPIcs, pp. 16:1–16:54. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2016)

29. Nešetřil, J., De Mendez, P.O.: On nowhere dense graphs. Eur. J. Comb. **32**(4), 600–617 (2011)

30. Oxley, J.G.: Matroid Theory, vol. 3. Oxford University Press, USA (2006)

31. Pferschy, U., Schauer, J.: The maximum flow problem with conflict and forcing conditions. In: Network Optimization, pp. 289–294. Springer (2011)

32. Pferschy, U., Schauer, J.: Approximation of knapsack problems with conflict and forcing graphs. J. Comb. Optim. **33**(4), 1300–1323 (2017)

33. Raman, V., Saurabh, S.: Short cycles make W-hard problems hard: FPT algorithms for W-hard problems in graphs with no short cycles. Algorithmica **52**(2), 203–225 (2008)

34. van Bevern, R., Tsidulko, O.Y., Zschoche, P.: Fixed-parameter algorithms for maximum-profit facility location under matroid constraints. In: International Conference on Algorithms and Complexity, pp. 62–74. Springer (2019)