Partially Polynomial Kernels for Set Cover and Test Cover*

Manu Basavaraju[1], Mathew C. Francis[2], M. S. Ramanujan[3], and Saket Saurabh[1,3]

1 **University of Bergen, Norway**
   `manu.basavaraju@ii.uib.no`
2 **Indian Statistical Institute (Chennai Centre), India**
   `mathew@isichennai.res.in`
3 **Institute of Mathematical Sciences, India**
   `{msramanujan|saket}@imsc.res.in`

———— **Abstract** ————

In a typical covering problem we are given a universe $\mathcal{U}$ of size n, a family $\mathcal{S}$ ($\mathcal{S}$ could be given implicitly) of size m and an integer k and the objective is to check whether there exists a subfamily $\mathcal{S}' \subseteq \mathcal{S}$ of size at most k satisfying some desired properties. If $\mathcal{S}'$ is required to contain all the elements of $\mathcal{U}$ then it corresponds to the classical Set Cover problem. On the other hand if we require $\mathcal{S}'$ to satisfy the property that for every pair of elements $x, y \in \mathcal{U}$ there exists a set $S \in \mathcal{S}'$ such that $|S \cap \{x, y\}| = 1$ then it corresponds to the Test Cover problem. In this paper we consider a natural parameterization of Set Cover and Test Cover. More precisely, we study the $(n-k)$-Set Cover and $(n-k)$-Test Cover problems, where the objective is to find a subfamily $\mathcal{S}'$ of size at most $n-k$ satisfying the respective properties, from the kernelization perspective. It is known in the literature that both $(n-k)$-Set Cover and $(n-k)$-Test Cover do not admit polynomial kernels (under some well known complexity theoretic assumptions). However, in this paper we show that they do admit "partially polynomial kernels". More precisely, we give polynomial time algorithms that take as input an instance $(\mathcal{U}, \mathcal{S}, k)$ of $(n-k)$-Set Cover ($(n-k)$-Test Cover) and return an equivalent instance $(\tilde{\mathcal{U}}, \tilde{\mathcal{S}}, \tilde{k})$ of $(n-k)$-Set Cover (respectively $(n-k)$-Test Cover) with $\tilde{k} \leq k$ and $|\tilde{\mathcal{U}}| = \mathcal{O}(k^2)$ ($|\tilde{\mathcal{U}}| = \mathcal{O}(k^7)$). These results allow us to generalize, improve and unify several results known in the literature. For example, these immediately imply traditional kernels when input instances satisfy certain "sparsity properties". Using a part of our kernelization algorithm for $(n-k)$-Set Cover, we also get an improved FPT algorithm for this problem which runs in time $\mathcal{O}(4^k k^{\mathcal{O}(1)}(m+n))$ improving over the previous best of $\mathcal{O}(8^{k+o(k)}(m+n)^{\mathcal{O}(1)})$. On the other hand the partially polynomial kernel for $(n-k)$-Test Cover implies the first single exponential FPT algorithm, an algorithm with running time $\mathcal{O}(2^{\mathcal{O}(k^2)}(m+n)^{\mathcal{O}(1)})$. We believe such an approach will also be useful for other covering problems as well.

1998 ACM Subject Classification F.2.2 Nonnumerical Algorithms and Problems, G.2.1 Combinatorics, G.2.2 Graph Theory

Keywords and phrases Set Cover, Test Cover, Kernelization, Parameterized Algorithms

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2013.67

33rd Int'l Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2013).
Editors: Anil Seth and Nisheeth K. Vishnoi; pp. 67–78
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1   Introduction

The input to a covering problem consists of a universe $\mathcal{U}$ of size n, a family $\mathcal{S}$ ($\mathcal{S}$ could be given implicitly) of size m and an integer k and the objective is to check whether there exists a subfamily $\mathcal{S}' \subseteq \mathcal{S}$ of size at most k satisfying some desired properties. If $\mathcal{S}'$ is required to contain all the elements of U, then it corresponds to the classical Set Cover problem. On the other hand if we require $\mathcal{S}'$ to satisfy the property that for every pair of elements $x, y \in U$ there exists a set $S \in \mathcal{S}'$ such that $|S \cap \{x, y\}| = 1$, then it corresponds to the Test Cover problem. We study both these problems from the point of view of kernelization – a subfield of parameterized complexity.

The goal of parameterized complexity is to find ways of solving NP-hard problems more efficiently than brute force: here the aim is to restrict the combinatorial explosion to a parameter that is hopefully much smaller than the input size. Formally, a *parameterization* of a problem is assigning an integer k to each input instance and we say that a parameterized problem is *fixed-parameter tractable (FPT)* if there is an algorithm that solves the problem in time $f(k) \cdot |I|^{\mathcal{O}(1)}$, where $|I|$ is the size of the input and f is an arbitrary computable function depending on the parameter k only. For more background, the reader is referred to the following monograph [10].

In this paper, we consider a natural parameterization of Set Cover and Test Cover. More precisely, we study the $(n-k)$-Set Cover and the $(n-k)$-Test Cover problems, where the objective is to find a subfamily $\mathcal{S}'$ of size at most $n - k$ with k as the parameter. Both these questions are motivated by the fact that there is a trivial solution of size n and the fact that Set Cover is W[2]-hard when parameterized by the solution size [10]. Both these parameterizations have been studied before in the literature. Gutin et al. [8] showed that the $(n-k)$-Set Cover (or equivalently, the $(m-k)$-Hitting Set) problem has an algorithm with running time $2^{\mathcal{O}(k^2)}(m+n)^{\mathcal{O}(1)}$. This was further improved by Crowston et al. [4], who gave a deterministic algorithm for this problem running in time $(2e)^{2k+\mathcal{O}(\log k)}(m+n)^{\mathcal{O}(1)}$ and a randomized algorithm running in expected time $8^{k+\mathcal{O}(\sqrt{k})}(m+n)^{\mathcal{O}(1)}$. Crowston et al. [5] studied, among other parameterizations, the $(n-k)$-Test Cover problem with the parameter k and gave an algorithm with running time $f(k) \cdot (m+n)^{\mathcal{O}(1)}$ for this problem. The function $f(k)$ has a tower of 2's in its exponent. In another paper, Crowston et al. [3] studied this problem on set families where the size of each set is bounded by some constant r. However, in this paper our focus would be on obtaining *partially polynomial kernels* for the $(n-k)$-Set Cover and $(n-k)$-Test Cover problems.

Preprocessing (data reduction or kernelization) as a strategy of coping with hard problems is universally used in almost every implementation. The history of preprocessing, such as applying reduction rules to simplify truth functions, can be traced back to the 1950's. Today, combining tools from parameterized complexity and classical complexity it has become possible to derive upper and lower bounds on the sizes of reduced instances, or so called *kernels*. A parameterized problem is said to admit a *kernel* if there is a polynomial time algorithm (the degree of polynomial is independent of k), called a *kernelization* algorithm, that reduces the input instance to an instance with size bounded by some function $h(k)$ of k only, while preserving the answer. If the function $h(k)$ is polynomial in k, then we say that the problem admits a polynomial kernel. While positive kernelization results have appeared regularly over the last two decades, the first results establishing infeasibility of polynomial kernels for specific problems have appeared only recently. In particular, Bodlaender et al. [2] and Fortnow and Santhanam [7] have developed a framework for showing that a problem does not admit a polynomial kernel unless $\mathsf{coNP} \subseteq \mathsf{NP/Poly}$, which is deemed unlikely. In

this paper, the parameterizations we consider for SET COVER and TEST COVER are known to exclude polynomial kernels unless an unlikely collapse happens in complexity theory. In particular, Gutin et al. [8] showed that the $(n − k)$-SET COVER problem does not have a polynomial kernel unless coNP $\subseteq$ NP/Poly while Gutin et al. [9] showed that the $(n − k)$-TEST COVER problem does not have a polynomial kernel unless coNP $\subseteq$ NP/Poly.

However, recently, a weaker notion of kernelization was introduced by Betzler et al. [1] who called it "partial" kernelization, where, instead of reducing the entire instance to one bounded by the size of the parameter as is the case with traditional kernelization algorithms, we try to bound "part" of the instance. For example, in the case of covering problems, instead of bounding the entire instance, the objective could be to give a bound on the either the universe or the family of sets as a polynomial function of k. For most covering problems, a bound on either the universe or the family of sets as a function of k immediately implies an FPT algorithm and in that respect, partially polynomial kernels are just as useful as traditional kernels.

Our contribution. In this paper, we show the existence of partially polynomial kernels for both the $(n − k)$-SET COVER and $(n − k)$-TEST COVER problems by giving polynomial time algorithms which produce an equivalent instance where the size of the universe is bounded by a polynomial in the parameter. In particular, we give polynomial time algorithms that, given instances of $(n − k)$-SET COVER and $(n − k)$-TEST COVER, return equivalent instances where the size of the universe is $\mathcal{O}(k^2)$ and $\mathcal{O}(k^7)$ respectively. Both our algorithms are based around computing a greedy mini-set (test) cover and using that as the starting point, we design "marking" rules with which we identify irrelevant elements and sets. The marking rules and proofs for $(n − k)$-SET COVER are relatively simpler when compared to the more complex marking rules and more involved proofs required for $(n − k)$-TEST COVER. These results allow us to generalize, improve and unify several results known in the literature [8, 4, 5, 3].

Our results imply traditional kernels for "sparse instances" of these problems, that is, instances where each element appears in a constant number of sets or the bipartite element-sets incidence graph excludes some fixed graphs as an induced subgraph or every set has bounded size. We also show how we can use a part of our kernelization algorithm for $(n − k)$-SET COVER to obtain an algorithm with running time $\mathcal{O}(4^k k^{\mathcal{O}(1)}(m + n))$ which gives a significant improvement over the previously known best deterministic algorithm running in time $(2e)^{2k+\mathcal{O}(\log k)}(m + n)^{\mathcal{O}(1)}$ and the best randomized algorithm running in expected time $8^{k+\mathcal{O}(\sqrt{k})}(m+n)^{\mathcal{O}(1)}$. Finally, the partially polynomial kernel for $(n−k)$-TEST COVER implies the first single exponential FPT algorithm, an algorithm with running time $\mathcal{O}(2^{\mathcal{O}(k^2)}(m + n)^{\mathcal{O}(1)})$, for the same. This significantly improves over the running time of the earlier algorithm for the problem [5]. We believe such an approach will be useful for other covering problems as well.

2 Bounded Universe kernel for $(n − k)$-SET COVER

In this section we give results for the $(n − k)$-SET COVER problem. We denote an instance of the problem by $(\mathcal{U}, \mathcal{S}, k)$ where $\mathcal{U}$ is the universe to be covered, $\mathcal{S}$ is the family of sets available and k is the parameter. Let $\mathcal{Q}(\mathcal{T})$ be the set of elements covered by the subcollection $\mathcal{T} \subseteq \mathcal{S}$, that is $\mathcal{Q}(\mathcal{T}) = \bigcup_{T \in \mathcal{T}} T$. A collection of sets $\mathcal{T}$ is said to be a *k-mini set cover* if $|\mathcal{Q}(\mathcal{T})| \geq |\mathcal{T}| + k$ and $|\mathcal{T}| \leq k$. If a sub collection $\mathcal{T} \subseteq \mathcal{S}$ is a k-mini set cover, then we say that $\mathcal{T}$ is a k-mini set cover *of* $\mathcal{S}$. The definition of mini set covers is motivated by Lemma 2 below. Note that the notion of a k-mini set cover is actually equivalent to the k-mini-hitting

set defined in [8] and a lemma equivalent to Lemma 2 is already proved in that paper. But for the sake of completeness, we restate the definition and proofs related to mini-hitting sets using the set cover terminology so that they appear natural in the context of mini-set covers. We first describe the **Greedy-mini-set($\mathcal{S}, k$)** algorithm which constructs a collection $\mathcal{T} \subseteq \mathcal{S}$ with certain properties, which we show imply the presence of a k-mini set cover.

Greedy-mini-set($\mathcal{S}, k$)
Start with $\mathcal{T} = \emptyset$.
Repeat until there are no more sets in $\mathcal{S} \setminus \mathcal{T}$ that can be added to $\mathcal{T}$ or until $|\mathcal{T}| \geq k$:
      If there exists $T \in \mathcal{S} \setminus \mathcal{T}$ such that $|\mathcal{Q}(\mathcal{T} \cup \{T\})| \geq |\mathcal{Q}(\mathcal{T})| + 2$, then $\mathcal{T} = \mathcal{T} \cup \{T\}$.

▶ **Lemma 1.** *If a collection of sets $\mathcal{S}$ is such that $|\mathcal{Q}(\mathcal{S})| \geq |\mathcal{S}| + k$, then $\mathcal{S}$ contains a k-mini set cover.*

Proof. Let $\mathcal{T}$ be the set returned by the algorithm **Greedy-mini-set($\mathcal{S}, k$)**. Clearly, $|\mathcal{T}| \leq k$. We claim that $\mathcal{T}$ is a k-mini set cover. Suppose that it is not. Then, it must be the case that $|\mathcal{Q}(\mathcal{T})| - |\mathcal{T}| < k$. Then, for every set $S \in \mathcal{S} \setminus \mathcal{T}$, $|\mathcal{Q}(\{S\}) \setminus \mathcal{Q}(\mathcal{T})| \leq 1$. This implies that $|\mathcal{Q}(\mathcal{S} \setminus \mathcal{T}) \setminus \mathcal{Q}(\mathcal{T})| \leq |\mathcal{S} \setminus \mathcal{T}|$. Therefore, we have that $|\mathcal{T}| + |\mathcal{S} \setminus \mathcal{T}| > |\mathcal{Q}(\mathcal{T})| - k + |\mathcal{Q}(\mathcal{S} \setminus \mathcal{T}) \setminus \mathcal{Q}(\mathcal{T})|$, which implies that $|\mathcal{S}| > |\mathcal{Q}(\mathcal{S})| - k$, which is a contradiction. This completes the proof of the lemma. ◀

▶ **Lemma 2.** *An instance $(\mathcal{U}, \mathcal{S}, k)$ of $(n-k)$-SET COVER is a YES instance if and only if there is a k-mini set cover $\mathcal{T} \subseteq \mathcal{S}$.*

Proof. We can assume that $\mathcal{Q}(\mathcal{S}) = \mathcal{U}$ as otherwise, there is no solution. Suppose that $\mathcal{S}$ contains a k-mini set cover $\mathcal{T}$. By definition, $|\mathcal{Q}(\mathcal{T})| \geq |\mathcal{T}| + k$. For every element $u \in \mathcal{U} \setminus \mathcal{Q}(\mathcal{T})$, add a set from $\mathcal{S}$ containing u to $\mathcal{T}$. Let $\mathcal{T}'$ denote the collection of sets so obtained. Clearly, $\mathcal{T}'$ is a set cover. Note that we add $n - |\mathcal{Q}(\mathcal{T})|$ sets to $\mathcal{T}$ to obtain $\mathcal{T}'$. Therefore, $|\mathcal{T}'| = n - |\mathcal{Q}(\mathcal{T})| + |\mathcal{T}| \leq n - k$ and hence $\mathcal{T}'$ is a set cover of size at most $n - k$ for $\mathcal{U}$.

If $\mathcal{S}$ contains a set cover $\mathcal{T}$ of size at most $n - k$, then we have $|\mathcal{T}| \leq n - k$ and $\mathcal{Q}(\mathcal{T}) = n$, and therefore by Lemma 1, $\mathcal{T}$ contains a k-mini set cover. This completes the proof of the lemma. ◀

Now we are ready to prove the two main results of this section.

▶ **Theorem 3.** *There is an algorithm that, given an instance $(\mathcal{U}, \mathcal{S}, k)$ of $(n-k)$-SET COVER, runs in polynomial time and returns an equivalent instance $(\mathcal{U}', \mathcal{S}', k)$ such that $|\mathcal{U}'| \leq 2k^2 - 2$.*

Proof. We first run the algorithm **Greedy-mini-set($\mathcal{S}, k$)** to construct a collection of sets $\mathcal{T}$. If $\mathcal{T}$ is a k-mini set cover, then we are done. Suppose that $\mathcal{T}$ is not a k-mini set cover. Note that if $|\mathcal{T}| \geq k$, then $\mathcal{T}$ is a k-mini set cover. So we can assume that $|\mathcal{T}| < k$. Since $\mathcal{T}$ is not a k-mini set cover, this implies that $|\mathcal{Q}(\mathcal{T})| \leq 2k - 2$.

Now any set in $\mathcal{S} \setminus \mathcal{T}$ covers at most one element from $\mathcal{U} \setminus \mathcal{Q}(\mathcal{T})$. Given $u \in \mathcal{Q}(\mathcal{T})$ and $v \in \mathcal{U} \setminus \mathcal{Q}(\mathcal{T})$, we shall denote by $\mathcal{S}_{u,v}$ all the sets in $\mathcal{S}$ that contain both u and v. We now carry out a marking procedure that marks at most $k(2k-2)$ elements from $\mathcal{U} \setminus \mathcal{Q}(\mathcal{T})$. For each element $u \in \mathcal{Q}(\mathcal{T})$, we mark k as yet unmarked elements from $\mathcal{U} \setminus \mathcal{Q}(\mathcal{T})$ such that $\mathcal{S}_{u,v} \neq \emptyset$. If there are no such k elements, we mark as many as we can. Note that during the marking procedure, no element is ever marked more than once. For an element $u \in \mathcal{Q}(\mathcal{T})$, let M_u denote the elements that are marked for u. Let $\mathcal{U}' = \mathcal{Q}(\mathcal{T}) \cup \bigcup_{u \in \mathcal{Q}(\mathcal{T})} M_u$. We shall define $\mathcal{S}' \subseteq \mathcal{S}$ to be all those sets which contain only elements from $\mathcal{U}'$ and we return the instance $(\mathcal{U}', \mathcal{S}', k)$. Note that $|\mathcal{U}'| \leq 2k^2 - 2$ and that the marking procedure can be carried

out in polynomial time. Therefore, it only remains for us to show that the instance $(\mathcal{U}', \mathcal{S}', k)$ of $(n - k)$-Set Cover is equivalent to the instance $(\mathcal{U}, \mathcal{S}, k)$ of the same problem.

Any k-mini set cover for $(\mathcal{U}', \mathcal{S}', k)$, is also clearly a k-mini set cover for $(\mathcal{U}, \mathcal{S}, k)$. To prove the other direction, we show that if there exists a k-mini set cover for $(\mathcal{U}, \mathcal{S}, k)$, then there exists a k-mini set cover for $(\mathcal{U}', \mathcal{S}', k)$.

Let $\mathcal{T}'$ be a k-mini set cover for $(\mathcal{U}, \mathcal{S}, k)$ which has the least number of sets from $\mathcal{S} \setminus \mathcal{S}'$. We shall show that $\mathcal{T}' \subseteq \mathcal{S}'$ and therefore it is also a k-mini set cover for $(\mathcal{U}', \mathcal{S}', k)$. Suppose that $\mathcal{T}' \not\subseteq \mathcal{S}'$. Pick a set $T \in \mathcal{T}'$ such that $T \notin \mathcal{S}'$. Let $W = T \cap \mathcal{Q}(\mathcal{T})$ and let z be the only element in $T \setminus \mathcal{Q}(\mathcal{T})$. Note that z exists, as otherwise $T \subseteq \mathcal{Q}(\mathcal{T}) \subseteq \mathcal{U}'$ in which case T would have been chosen in $\mathcal{S}'$. Also, $W \neq \emptyset$ because $|T| \geq 2$ (if $|T| \leq 1$, then $\mathcal{T}' \setminus \{T\}$ is a k-mini set cover with smaller number of sets from $\mathcal{S} \setminus \mathcal{S}'$).

Consider an element $u \in W$. The fact that $T \notin \mathcal{S}'$ means that $z \notin M_u$. This implies that the marking procedure marked k elements from $\mathcal{U} \setminus \mathcal{Q}(\mathcal{T})$ other than z for u. Therefore, $|M_u| = k$. Since $T \setminus \mathcal{Q}(\mathcal{T}) = \{z\}$ and $M_u \subset \mathcal{U} \setminus \mathcal{Q}(\mathcal{T})$, we have $T \cap M_u = \emptyset$. Recall that every set in $\mathcal{S}$ contains at most one element from $\mathcal{U} \setminus \mathcal{Q}(\mathcal{T})$. This means that every set in $\mathcal{S}$ contains at most one element from M_u. As $|\mathcal{T}' \setminus \{T\}| \leq k - 1$, it must be the case that at least one of the elements in M_u is not covered by $\mathcal{T}'$. For every element $u \in W$, we define $r(u)$ to be an element from $M_u \setminus \mathcal{Q}(\mathcal{T}')$. Recall that for distinct $u, v \in W$, the sets M_u and M_v are disjoint by definition. This means that for distinct $u, v \in W$, $r(u) \neq r(v)$. We denote by S_u any set from the collection $\mathcal{S}_{u, r(u)}$ (recall that $\mathcal{S}_{u, r(u)}$ is the collection of sets in $\mathcal{S}$ which contain both u and $r(u)$). Note that $\mathcal{S}_{u, r(u)} \neq \emptyset$ since $r(u) \in M_u$. We now claim that the collection $\mathcal{T}'' = (\mathcal{T}' \setminus \{T\}) \cup \bigcup_{u \in W} S_u$ is also a k-mini set cover.

Note that the collection $\mathcal{T}''$ covers every element covered by the collection $\mathcal{T}'$ except z since $W \subseteq \bigcup_{u \in W} S_u \subseteq \mathcal{Q}(\mathcal{T}'')$. However, for each $u \in W$, $\mathcal{T}''$ also covers at least one element that was not covered by $\mathcal{T}'$ (recall that the element $r(u)$ is not covered by $\mathcal{T}'$). Also, since for distinct $u, v \in W$, $r(u) \neq r(v)$, we have $\left| \left(\bigcup_{u \in W} S_u \right) \setminus \mathcal{Q}(\mathcal{T}') \right| \geq |W|$. Thus, $|\mathcal{Q}(\mathcal{T}'')| \geq |\mathcal{Q}(\mathcal{T}')| - 1 + |W|$. Clearly, we have $|\mathcal{T}''| = |\mathcal{T}'| + |W| - 1$. Therefore, since $\mathcal{T}'$ was a k-mini set cover, $|\mathcal{Q}(\mathcal{T}'')| \geq |\mathcal{T}'| + k + |W| - 1 = |\mathcal{T}''| + k$. Now, by Lemma 1, $\mathcal{T}''$ contains a k-mini set cover. Since for every $u \in W$, $S_u \in \mathcal{S}'$, this k-mini set cover uses at least one less set from $\mathcal{S} \setminus \mathcal{S}'$ than $\mathcal{T}'$, which contradicts our choice of $\mathcal{T}'$. Therefore, we conclude that $\mathcal{T}'$ is a k-mini set cover for $(\mathcal{U}', \mathcal{S}', k)$. This completes the proof of the theorem. ◀

For our next result, we need the notion of *path decomposition*. Let $G = (V, E)$ be a graph. A *path decomposition* of G is a pair $(T, \mathcal{X} = \{X_t\}_{t \in V(T)})$ where T is a path and $\mathcal{X}$ is a collection of subsets of V such that:

- $\forall e = \{u, v\} \in E, \ \exists t \in V(T) : \{u, v\} \subseteq X_t$ and
- $\forall v \in V, \ T[\{t \mid v \in X_t\}]$ is non-empty and connected.

The *width* of $(T, \mathcal{X})$ is equal to $\max\{|X_t| - 1 \mid t \in V(T)\}$ and the *pathwidth* of $G = (V, E)$, denoted by $\mathbf{pw}(G)$, is the minimum width over all path decompositions of G.

▶ **Theorem 4.** *There is an algorithm that, given an instance $(\mathcal{U}, \mathcal{S}, k)$ of $(n - k)$-Set Cover, runs in time $\mathcal{O}(4^k k^{\mathcal{O}(1)}(m + n))$ and either returns a set cover of size at most $n - k$ or correctly concludes that no such set cover exists.*

Proof. Let $\mathcal{T}$ be the set returned by the algorithm **Greedy-mini-set$(\mathcal{S}, k)$**. If $\mathcal{T}$ is a k-mini set cover, then using Lemma 2, we can construct in polynomial time and return an $(n - k)$-set cover. So let us assume that $\mathcal{T}$ is not a k-mini set cover. Note that if $|\mathcal{T}| \geq k$, then $\mathcal{T}$ is a k-mini set cover. So we can assume that $|\mathcal{T}| < k$. Since $\mathcal{T}$ is not a k-mini set cover, this implies that $|\mathcal{Q}(\mathcal{T})| \leq 2k - 2$. Furthermore, we have that every set in $\mathcal{S} \setminus \mathcal{T}$ covers at most 1 element from $\mathcal{U} \setminus \mathcal{Q}(\mathcal{T})$.

We now consider the incidence bipartite graph $B = (X, Y)$ of the family $(\mathcal{U}, \mathcal{S})$ where the vertices in X correspond to $\mathcal{S}$, those in Y correspond to $\mathcal{U}$. Let Y' be those vertices in Y which correspond to $\mathcal{U} \setminus \mathcal{Q}(\mathcal{T})$. Since every set in $\mathcal{S} \setminus \mathcal{T}$ covers at most 1 element from $\mathcal{U} \setminus \mathcal{Q}(\mathcal{T})$, it is clear that in B, a vertex in X has at most one neighbour in Y'. Therefore, the induced subgraph $B' = B[X \cup Y']$ is a disjoint union of isolated vertices and stars. This implies that $\mathbf{pw}(B') \leq 1$. Therefore, we have that $\mathbf{pw}(B) \leq 2k$ since $|Y \setminus Y'| < 2k$.

Now, it only remains for us to solve the following problem. Given a set family $(\mathcal{U}, \mathcal{S})$ and an integer k such that the corresponding incidence bipartite graph B has pathwidth at most $2k$, the objective is to find the minimum set cover of this family. Observe that since X and Y are the bipartition's of B which correspond to $\mathcal{U}$ and $\mathcal{S}$ respectively, the objective is to find the smallest subset of Y which *dominates* the set X. Therefore, this is an instance of the RED-BLUE DOMINATING SET problem on graphs of bounded pathwidth, which is known to have a $2^p(n + m)^{\mathcal{O}(1)}$ time algorithm due to [6, Lemma 6], where p is the pathwidth of the graph. In fact, the algorithm mentioned in [6, Lemma 6] can be made to work in time $\mathcal{O}(2^p p^{\mathcal{O}(1)}(m + n))$. Using the fact that $\mathcal{Q}(\mathcal{T}) < 2k$ and by making sure that we do only one pass through $\mathcal{S}$, we can obtain $\mathcal{T}$ and $\mathcal{Q}(\mathcal{T})$ in time $\mathcal{O}(k(m + n))$ and thus the path decomposition of B of width at most $2k$ can be obtained in time $\mathcal{O}(k(m + n))$. Therefore, using the algorithm mentioned in [6], we get an algorithm for $(n - k)$-SET COVER running in time $\mathcal{O}(4^k k^{\mathcal{O}(1)}(m + n))$. This completes the proof of the theorem. ◀

Since $(n - k)$-SET COVER is parameterized equivalent to $(m - k)$-HITTING SET we get the following result as a corollary to Theorem 4.

▶ **Corollary 5.** $(m - k)$-HITTING SET *can be solved in time* $\mathcal{O}(4^k k^{\mathcal{O}(1)}(m + n))$.

The algorithm mentioned in Corollary 5 improves the known (deterministic as well as randomized) algorithms for $(m - k)$-HITTING SET obtained in [4]. This algorithm was used as a subroutine in [4] to obtain a FPT algorithm for $(\nu(F) + k)$-SAT. In $(\nu(F) + k)$-SAT we are a given a CNF formula Φ and a positive integer k and the objective is to test whether there exists an assignment to its variables that satisfy at least $(\nu(F) + k)$ clauses of Φ. Here, $\nu(F)$ denotes the size of maximum matching in the variable clause incidence graph of Φ. Since it is the algorithm for $(m - k)$-HITTING SET that dominates the running time of the algorithm for $(\nu(F) + k)$-SAT in [4], we get the following result for $(\nu(F) + k)$-SAT using the Corollary 5.

▶ **Corollary 6.** $(\nu(F) + k)$-SAT *can be solved in time* $\mathcal{O}(4^k(m + n)^{\mathcal{O}(1)})$.

3 Bounded Universe Kernel for $(n - k)$-TEST COVER

We start by recalling the problem definition. Given a universe $\mathcal{U}$, we shall call a set $T \subseteq \mathcal{U}$ a *test*. A test T is said to *separate* $u, v \in \mathcal{U}$ if $|T \cap \{u, v\}| = 1$. A collection of tests $\mathcal{T}$ is a *test cover* for $\mathcal{U}$, if for any $u, v \in \mathcal{U}$, there exists a test $T \in \mathcal{T}$ that separates u and v. Given a universe of size n and a collection of tests $\mathcal{S}$, the $(n - k)$-TEST COVER problem is the problem of deciding whether $\mathcal{S}$ contains a test cover of size at most $n - k$. We denote an instance of the problem by $(\mathcal{U}, \mathcal{S}, k)$.

3.1 Starting with the Greedy Algorithm

Given a collection of tests $\mathcal{T}$, define the equivalence relation $R_{\mathcal{T}}$ on $\mathcal{U}$ as follows: $u R_{\mathcal{T}} v$ if there is no test in $\mathcal{T}$ that separates u and v. The equivalence classes of $R_{\mathcal{T}}$ are said to be the *classes induced by* $\mathcal{T}$ *in* $\mathcal{U}$. It is easy to see that the classes induced by $\mathcal{T}$ form a partition

of $\mathcal{U}$. Note that if $\mathcal{T}$ is a test cover, then $\mathcal{T}$ induces exactly n classes in $\mathcal{U}$. We denote by $\mathcal{C}_{\mathcal{U}}(\mathcal{T})$ the set of classes induced by $\mathcal{T}$ in $\mathcal{U}$. When the universe $\mathcal{U}$ is clear from the context, we abbreviate this to just $\mathcal{C}(\mathcal{T})$.

For any test $T \subseteq \mathcal{U}$, we shall define $\overline{T}$ to be $\mathcal{U} \setminus T$. The following observation follows directly from the fact for any test T, both T and $\overline{T}$ separate exactly the same pairs of elements in $\mathcal{U}$.

▶ **Observation 1.** *For any collection of tests* $\mathcal{T}$ *and* $T \in \mathcal{T}$, $\mathcal{C}(\mathcal{T}) = \mathcal{C}((\mathcal{T} \setminus \{T\}) \cup \overline{T})$.

A collection of tests $\mathcal{T}$ is said to be a *k-mini test cover* if $|\mathcal{C}(\mathcal{T})| \geq |\mathcal{T}| + k$ and $|\mathcal{T}| \leq 2k$. We first describe the **Greedy-mini-test**$(\mathcal{S}, k)$ algorithm from [5]. Like before, this algorithm constructs a collection $\mathcal{T}$ of tests from $\mathcal{S}$ which has some special properties.

Greedy-mini-test$(\mathcal{S}, k)$
Start with $\mathcal{T} = \emptyset$.
Repeat until there are no more tests in $\mathcal{S} \setminus \mathcal{T}$ that can be added to $\mathcal{T}$ or until $|\mathcal{T}| \geq 2k - 2$:
       If there exists $T \in \mathcal{S} \setminus \mathcal{T}$ such that $|\mathcal{C}(\mathcal{T} \cup \{T\})| \geq |\mathcal{C}(\mathcal{T})| + 2$, then $\mathcal{T} = \mathcal{T} \cup \{T\}$.
       Else if there exist $T_1, T_2 \in \mathcal{S} \setminus \mathcal{T}$, such that $|\mathcal{C}(\mathcal{T} \cup \{T_1, T_2\})| \geq |\mathcal{C}(\mathcal{T})| + 3$, then
       $\mathcal{T} = \mathcal{T} \cup \{T_1, T_2\}$.

Given a subcollection $\mathcal{T} \subseteq \mathcal{S}$, we say that $(\mathcal{S}, \mathcal{T})$ is *saturated* if for any test $T \in \mathcal{S} \setminus \mathcal{T}$, $|\mathcal{C}(\mathcal{T} \cup \{T\})| \leq |\mathcal{C}(\mathcal{T})| + 1$ and for any pair of tests $T_1, T_2 \in \mathcal{S} \setminus \mathcal{T}$, $|\mathcal{C}(\mathcal{T} \cup \{T_1, T_2\})| \leq |\mathcal{C}(\mathcal{T})| + 2$. Note that if $\mathcal{T}$ was a collection of tests chosen by **Greedy-mini-test**(S, k), then $(\mathcal{S}, \mathcal{T})$ is saturated.

▶ **Observation 2.** *Let* $(\mathcal{S}, \mathcal{T})$ *be saturated. Let* $T \in \mathcal{S} \setminus \mathcal{T}$ *and* $\tilde{S} = (\mathcal{S} \cup \{\mathcal{U} \setminus T\}) \setminus \{T\}$. *Then,* $(\tilde{S}, \mathcal{T})$ *is also saturated.*

Observation 2 follows directly from Observation 1. The following lemmas, that appeared in [5], can be seen to have proofs similar to those of Lemmas 1 and 2.

▶ **Lemma 7** ([5]). *If a collection of tests* $\mathcal{S}$ *is such that* $|\mathcal{C}(\mathcal{S})| \geq |\mathcal{S}| + k$, *then* $\mathcal{S}$ *contains a k-mini test cover.*

▶ **Lemma 8** ([5]). *A collection* $\mathcal{S}$ *of tests contains a test cover of size at most* $n - k$ *if and only if it contains a k-mini test cover.*

Let $\mathcal{T} \subseteq \mathcal{S}$ be a collection of tests. Now we say that any test $T \in \mathcal{S} \setminus \mathcal{T}$ intersects a class R in $\mathcal{C}(\mathcal{T})$ *trivially* if $T \cap R = R$ or $T \cap R = \emptyset$. Otherwise, we say that the intersection of T with R is *non-trivial*. If T has a non-trivial intersection with R, then consider the two sets $A = R \cap T$ and $B = R \setminus T$. It is easy to see that no two elements u, v such that $u, v \in A$ or $u, v \in B$ are separated by any test in $\mathcal{T} \cup \{T\}$ and that any two elements u, v such that $u \in A$ and $v \in B$ are separated by $T \in \mathcal{T} \cup \{T\}$, implying that A and B are two classes in $\mathcal{C}(\mathcal{T} \cup \{T\})$. Note that $R = A \cup B$. We therefore say that the class R of $\mathcal{C}(\mathcal{T})$ is *split* into the classes A and B of $\mathcal{C}(\mathcal{T} \cup \{T\})$.

▶ **Observation 3.** *For any collection of tests* $\mathcal{T}$, *and a test* $T \in \mathcal{S} \setminus \mathcal{T}$, *any class* R *of* $\mathcal{C}(\mathcal{T})$ *is split into at most two classes of* $\mathcal{C}(\mathcal{T} \cup \{T\})$. *In particular, the classes of* $\mathcal{C}(\mathcal{T})$ *which have trivial intersections with* $\mathcal{T}$ *are also classes of* $\mathcal{C}(\mathcal{T} \cup \{T\})$, *whereas the classes of* $\mathcal{C}(\mathcal{T})$ *with non-trivial intersections with* T *are split into two classes of* $\mathcal{C}(\mathcal{T} \cup \{T\})$.

3.2   Outer Phase of the Kernelization Algorithm

Run **Greedy-mini-test**$(\mathcal{S}, k)$. Let $\mathcal{T}$ be the tests that were chosen by the greedy algorithm. Clearly, $(\mathcal{S}, \mathcal{T})$ is saturated. If $\mathcal{T}$ is a k-mini test cover, we are done. Suppose that $\mathcal{T}$ is

not a k-mini test cover. Observe that in the first iteration of the algorithm, two tests get added together, creating four classes. After that each iteration adds at least $\frac{3}{2}x$ classes where x is the number of tests added to $\mathcal{T}$ in that iteration. Therefore, we can conclude that $|\mathcal{C}(\mathcal{T})| \geq \frac{3}{2}(|\mathcal{T}| - 2) + 4 = |\mathcal{T}| + \frac{|\mathcal{T}|+2}{2}$. This shows that when $|\mathcal{T}| \geq 2k - 2$, $\mathcal{T}$ is a k-mini test cover. Therefore, as $\mathcal{T}$ is not a k-mini test cover, we have $|\mathcal{T}| \leq 2k - 3$ and $|\mathcal{C}(\mathcal{T})| \leq 3k - 4$.

We shall call the classes in $\mathcal{C}(\mathcal{T})$ **regions**.

▶ **Observation 4.** $(*)$ [1] *Any test T can have non-trivial intersection with at most one region.*

Given a region R, a strict subset X of R is said to be a ***chunk*** in R if there exists a test $T \in \mathcal{S}$ such that $R \cap T = X$. Let $T \in \mathcal{S} \setminus \mathcal{T}$ and $R \in \mathcal{C}(\mathcal{T})$. Let $X = R \cap T$ be a chunk in a region R such that $|X| > |R|/2$. We construct a new collection $\tilde{S}$ of tests by replacing T with $\mathcal{U} \setminus T$ in $\mathcal{S}$. By Observation 2, $(\tilde{S}, \mathcal{T})$ is saturated. Now, we set $\mathcal{S} = \tilde{S}$. We repeat this procedure for every such $T \in \mathcal{S} \setminus \mathcal{T}$. After this is done, $(\mathcal{S}, \mathcal{T})$ is still saturated and therefore satisfies Observation 4 and moreover, $\mathcal{S}$ has the property that for any test $T \in \mathcal{S} \setminus \mathcal{T}$ and for any region $R \in \mathcal{C}(\mathcal{T})$, $|T \cap R| \leq |R|/2$.

▶ **Observation 5.** *Given any two chunks $X_1 = R \cap T_1$ and $X_2 = R \cap T_2$ where $T_1, T_2 \in \mathcal{S} \setminus \mathcal{T}$, then either one of them is a subset of the other or the two are disjoint.*

Proof. Suppose for the sake of contradiction that $X_1 \cap X_2 \neq \emptyset$ and $X_1 \setminus X_2 \neq \emptyset$ and $X_2 \setminus X_1 \neq \emptyset$. Note that, by the argument given above, we can assume that $|X_1| \leq |R|/2$ and $|X_2| \leq |R|/2$. Since $X_1 \cap X_2 \neq \emptyset$, this means that $X_1 \cup X_2 \neq R$, or in other words, $R \setminus (X_1 \cup X_2) \neq \emptyset$. This, together with the fact that $X_1 \cap X_2 \neq \emptyset$, $X_1 \setminus X_2 \neq \emptyset$ and $X_2 \setminus X_1 \neq \emptyset$ implies that $|\mathcal{C}(\mathcal{T} \cup \{T_1, T_2\})| = |\mathcal{C}(\mathcal{T})| + 3$. This contradicts the fact that $(\mathcal{S}, \mathcal{T})$ was saturated.   ◀

3.3   A step of the kernelization algorithm applied to a region

Let $R_1, R_2, \ldots, R_t$ be the regions in $\mathcal{C}(\mathcal{T})$. We now restrict our attention to a region R and define the set of indices $I = \{i : 1 \leq i \leq t \text{ and } R_i \neq R\}$. We shall denote by $P' = (I \times I) \setminus \{(i, i) : i \in I\}$ the set of pairs (i, j) from I with $i \neq j$. We define the set P as $P = P' \cup \{(0, 0), (-1, -1)\}$.

We say that a chunk X in R is an (i, j)-***chunk***, where $(i, j) \notin \{(0, 0), (-1, -1)\}$, if there exists a test T with $R \cap T = X$ and $T \cap R_i = R_i$ and $T \cap R_j = \emptyset$. We call a chunk X in R a $(0, 0)$-***chunk*** if there exists a test T with $R \cap T = X$ and $T \cap R_i = R_i$, for all $R_i \neq R$. And we call a chunk X in R a $(-1, -1)$-***chunk*** if there exists a test T with $R \cap T = X$ and $T \cap R_i = \emptyset$, for all $R_i \neq R$. We now start a marking procedure in which we mark some of the chunks in R with pairs from P. The marking is done in two phases.

Phase I.   In this phase, we use only the labels from $P \setminus \{(-1, -1)\}$. Each chunk is considered for marking at most once. A chunk that has already been considered for marking is called ***checked***. We choose for checking a chunk such that every chunk contained in it is both checked and unmarked. We mark this chunk with a label "(i, j)" such that this chunk is an (i, j)-chunk and there are not already $6k + 3$ chunks marked with the label "(i, j)". If such a label does not exist, we leave this chunk unmarked. In either case, this chunk has now become a checked chunk. This procedure goes on until there are no more chunks to check or until each label in $P \setminus \{(-1, -1)\}$ has been marked on $6k + 3$ chunks.

---

[1]   Due to lack of space, proofs of results marked $(*)$ are omitted.

Note that a chunk will be marked with at most one pair. Also note that no chunk marked with some pair "(i, j)" will strictly contain a marked chunk. The chunks that have been marked after this phase of the marking procedure shall be called **solid chunks**. In the rest of the paper, we use the term ***irrelevant chunk*** to refer to a chunk that does not contain any solid chunk. Note that by definition, irrelevant chunks are unmarked and it shall shortly be clear that they remain unmarked even after the second phase of marking.

▶ **Observation 6.** ($*$) *If X is an irrelevant (i, j)-chunk, then there exist $6k + 3$ solid chunks that are marked "(i, j)" each of which is disjoint from X.*

Phase II. A ***wrapper*** is a chunk that strictly contains some solid chunk. We say that two wrappers belong to the same ***level*** if they contain the same solid chunks. Note that by Observation 5, all wrappers of the same level form a chain under inclusion. We now start the second phase of marking chunks. In this phase, we mark wrappers in each level separately. In each level, for each pair (i, j) from P, we mark with "(i, j)" the $2k + 1$ smallest and $2k + 1$ largest unmarked (i, j)-wrappers in the chain formed by (i, j)-wrappers in that level. If for some pair (i, j), there are no $4k + 2$ unmarked (i, j)-wrappers in that level, we mark all the unmarked (i, j)-wrappers in that level. Note that there can be at most one pair marked on any wrapper.

3.3.1 Pruning the Universe and the Family with respect to a region

From $\mathcal{S}$, delete every test T such that $T \cap R$ is an unmarked chunk. Let $\mathcal{S}' \subseteq \mathcal{S}$ denote the resulting collection of tests. Observe that $\mathcal{T} \subseteq \mathcal{S}'$.

Note that the chunks that remain in R are all marked chunks. Now, from $\mathcal{U}$, we delete elements as follows. For every solid chunk, we retain one element in it and delete all other elements. For every deleted element v in a solid chunk, we define $rep(v)$ to be the vertex that is retained from that chunk.

▶ **Claim 1.** *If v is an element that was deleted from a solid chunk X, then there is no test in $\mathcal{S}'$ that separates v and $rep(v)$.*

Proof. Let x and y be any two elements in X. Suppose there is a test $T \in \mathcal{S}'$ such that $x \in T$ and $y \notin T$, then it means that $T \cap R$ is a chunk that contains x and not y. Therefore, $T \cap R$ is a strict subset of X (because of Observation 5). But observe that $T \cap R$ is a marked chunk. This contradicts the fact that X was a solid chunk. Therefore there is no test $T \in \mathcal{S}'$ that separates any two elements of X. Thus, we can infer that there is no test in $\mathcal{S}'$ which separates v and $rep(v)$. ◀

For a wrapper W that is not the smallest in its level, we shall define $rep(W)$ to be an element (chosen arbitrarily) in $W \setminus W'$, where W' is the largest wrapper of the same level that is strictly contained in it.

For every marked wrapper M that is not the smallest marked wrapper in its level, let M' be the largest marked wrapper that is strictly contained in it. We delete every element in $M \setminus M'$ except $rep(M)$. Again, for every element $v \in M \setminus M'$ that gets deleted, we let $rep(v) = rep(M)$. (Note that for $v \in M \setminus M'$, we have $rep(v) \in M \setminus M'$.)

▶ **Claim 2.** ($*$) *If v is an element that was deleted from $M \setminus M'$, then there is no test $T \in \mathcal{S}'$ that separates v and $rep(v)$.*

For a marked wrapper M that is the smallest marked wrapper in its level, let M' be the set of all elements in M that is not contained in any marked subchunk of M. If M' is not

empty, we delete every element of M' except one. As before, for every deleted element v in M', we let $rep(v)$ be the element in M' that is not deleted.

▶ **Claim 3.** (∗) *If v is an element that was deleted from M', then there is no test $T \in \mathcal{S}'$ that separates v and $rep(v)$.*

Let R' be the set of elements in R that are not inside any marked chunk. If R' is not empty, delete every element from R' except one. Here again, for every deleted element v, we denote by $rep(v)$ the element in R' that is retained.

▶ **Claim 4.** (∗) *If v is an element that was deleted from the R', then there is no test $T \in \mathcal{S}'$ that separates v and $rep(v)$.*

> We denote by $\mathcal{U}' \subseteq \mathcal{U}$ the set of elements that do not get deleted.

▶ **Lemma 9.** *If $v \in \mathcal{U} \setminus \mathcal{U}'$, then there is no test in $\mathcal{S}'$ that separates v and $rep(v)$.*

Proof. The proof is immediate from the above claims.   ◀

▶ **Corollary 10.** *There is no class in $\mathcal{C}(\mathcal{S}')$ that consists only of elements from $\mathcal{U} \setminus \mathcal{U}'$.*

Proof. Suppose there is a class $D \in \mathcal{C}(\mathcal{S}')$ such that $D \subseteq \mathcal{U} \setminus \mathcal{U}'$. Consider any element $v \in D$. By Lemma 9, there is no test in $\mathcal{S}'$ that separates v and $rep(v)$. This implies that $rep(v) \in D$. As $rep(v) \in \mathcal{U}'$, this contradicts the assumption that $D \subseteq \mathcal{U} \setminus \mathcal{U}'$.   ◀

Note that if $\mathcal{Z} \subseteq \mathcal{S}'$, then every class in $\mathcal{C}(\mathcal{Z})$ is a union of some classes from $\mathcal{C}(\mathcal{S}')$. This immediately gives us the following corollary.

▶ **Corollary 11.** *If $\mathcal{Z} \subseteq \mathcal{S}'$, then there is no class in $\mathcal{C}(\mathcal{Z})$ that consists only of elements from $\mathcal{U} \setminus \mathcal{U}'$.*

3.3.2   Proof of correctness and the size bound on $\mathcal{U}' \cap R$

Now, we shall show that our kernelization algorithm, given an instance $(\mathcal{U}, \mathcal{S}, k)$ of $(n - k)$-Test Cover produces an equivalent instance $(\mathcal{U}', \mathcal{S}', k)$ of $(n-k)$-Test Cover. In particular, we shall show that there is a k-mini test cover for $(\mathcal{U}, \mathcal{S}, k)$ if and only if there is a k-mini test cover for $(\mathcal{U}', \mathcal{S}', k)$. Note that any k-mini test cover for $(\mathcal{U}', \mathcal{S}', k)$ is also a k-mini test cover for $(\mathcal{U}, \mathcal{S}, k)$ and therefore the "if" direction is clear. To show the other direction, we shall show that whenever there is a k-mini test cover for $(\mathcal{U}, \mathcal{S}, k)$, there is also a k-mini test cover for $(\mathcal{U}', \mathcal{S}', k)$. This is proved in Lemma 14.

First, we show how, when given a k-mini test cover of $(\mathcal{U}, \mathcal{S}, k)$ that contains some deleted tests (i.e., some tests from $\mathcal{S} \setminus \mathcal{S}'$), we can obtain a new k-mini test cover of $(\mathcal{U}, \mathcal{S}, k)$ that contains a lesser number of deleted tests. Lemma 12 shows how this can be done when $\mathcal{Z}$ contains at least one deleted test whose chunk in R was an irrelevant chunk. Then, Lemma 13 shows how this can be done when $\mathcal{Z}$ contains no deleted test whose chunk in R is irrelevant (i.e., for every deleted test in $\mathcal{Z}$, its chunk in R is an unmarked wrapper).

▶ **Lemma 12.** (∗) *Let $\mathcal{Z}$ be a k-mini test cover for $(\mathcal{U}, \mathcal{S}, k)$. Let $T \in \mathcal{Z} \setminus \mathcal{S}'$ such that there is no other test $T' \in \mathcal{Z} \setminus \mathcal{S}'$ having $T' \cap R \subset T \cap R$. Also let $T \cap R$ be an irrelevant chunk (that is, $T \cap R$ is a chunk that was not marked and also did not contain any marked chunk). Then there is a collection $\mathcal{F} \subseteq \mathcal{S}'$ such that $(\mathcal{Z} \setminus \{T\}) \cup \mathcal{F}$ contains a k-mini test cover for $(\mathcal{U}, \mathcal{S}, k)$.*

▶ **Lemma 13.** (∗) *Let $\mathcal{Z}$ be a k-mini test cover of $(\mathcal{U}, \mathcal{S}, k)$ such that for each $T' \in \mathcal{Z} \setminus \mathcal{S}'$, $T' \cap R$ is a wrapper. Let $T \in \mathcal{Z} \setminus \mathcal{S}'$. Then there is a collection $\mathcal{F} \subseteq \mathcal{S}'$ such that $(\mathcal{Z} \setminus \{T\}) \cup \mathcal{F}$ contains a k-mini test cover for $(\mathcal{U}, \mathcal{S}', k)$.*

▶ **Lemma 14.** *If there is a k-mini test cover $\mathcal{Z} \subseteq \mathcal{S}$ of $\mathcal{U}$, then there is a k-mini test cover $\mathcal{Z}' \subseteq \mathcal{S}'$ of $\mathcal{U}'$.*

Proof. Since there is a k-mini test cover $\mathcal{Z}$ of $\mathcal{U}$, there must be a k-mini test cover $\mathcal{Z}'$ of $\mathcal{U}$ which contains the least number of tests from $\mathcal{S} \setminus \mathcal{S}'$. We claim that $\mathcal{Z}'$ is the required k-mini test cover of $\mathcal{U}'$ that contains no test from $\mathcal{S} \setminus \mathcal{S}'$. First, let us suppose for the sake of contradiction that $\mathcal{Z}' \cap (\mathcal{S} \setminus \mathcal{S}') \neq \emptyset$. Consider the case when there exists a test $T \in \mathcal{Z}' \cap (\mathcal{S} \setminus \mathcal{S}')$ such that $T \cap R$ is an irrelevant chunk. Then there exists a $T' \in \mathcal{Z}' \cap (\mathcal{S} \setminus \mathcal{S}')$ such that $T' \cap R$ is an irrelevant chunk and there is no $Z \in \mathcal{Z}' \cap (\mathcal{S} \setminus \mathcal{S}')$ with $Z \cap R \subset T' \cap R$ (recall that any chunk contained in an irrelevant chunk is also irrelevant). Then by Lemma 12, we have a k-mini test cover of $\mathcal{U}$ with lesser number of tests from $\mathcal{S} \setminus \mathcal{S}'$ than $\mathcal{Z}'$, which is a contradiction. Therefore, we can assume that there is no test in $T \in \mathcal{Z}' \cap (\mathcal{S} \setminus \mathcal{S}')$ such that $T \cap R$ is an irrelevant chunk. Now consider any $T \in \mathcal{Z}' \cap (\mathcal{S} \setminus \mathcal{S}')$. We know that $T \cap R$ is a wrapper. Then by Lemma 13, we can find a k-mini test cover of $\mathcal{U}$ that has lesser number of tests from $\mathcal{S} \setminus \mathcal{S}'$ than $\mathcal{Z}'$, which is again a contradiction. Therefore, we have $\mathcal{Z}' \cap (\mathcal{S} \setminus \mathcal{S}') = \emptyset$ or in other words, $\mathcal{Z}' \subseteq \mathcal{S}'$. Now, from Corollary 11, we know that $|\mathcal{C}_{\mathcal{U}'}(\mathcal{Z}')| = |\mathcal{C}_{\mathcal{U}}(\mathcal{Z}')|$. Thus, $\mathcal{Z}'$ is a k-mini test cover of $\mathcal{U}'$ as well. ◀

▶ **Lemma 15.** $|\mathcal{U}' \cap R| = \mathcal{O}(k^6)$.

Proof. In Phase I, we mark at most $(6k + 3)|P| = \mathcal{O}(k^3)$ chunks. In Phase II, there at most $\mathcal{O}(k^3)$ different levels and in each level, we mark at most $2(2k + 1)|P| = \mathcal{O}(k^3)$ wrappers. Thus there are totally $\mathcal{O}(k^6)$ marked wrappers. Therefore, the total number of marked chunks in R is $\mathcal{O}(k^6)$. Since each element of $\mathcal{U}' \cap R$, except the one representative element that is not contained in any chunk in R, is associated with a chunk (a solid chunk or a marked wrapper), and since this association is one-to-one, $|\mathcal{U}' \cap R| = \mathcal{O}(k^6)$. ◀

3.4 Final Result and Algorithms for $(n - k)$-Test Cover

We summarize the kernelization algorithm in the following theorem.

▶ **Theorem 16.** *There is an algorithm that, given an instance $(\mathcal{U}, \mathcal{S}, k)$ of $(n - k)$-Test Cover, runs in polynomial time and returns an equivalent instance $(\mathcal{U}', \mathcal{S}', k)$ such that $|\mathcal{U}'| = \mathcal{O}(k^7)$ and $|\mathcal{S}'| = \mathcal{O}(2^{3k} \cdot k^7)$.*

Proof. We run the outer phase of the algorithm only once. Let $R_1, R_2, \ldots, R_t$ be the regions in $\mathcal{C}(\mathcal{T})$ after the outer phase is done. Now we do the marking and pruning procedure iteratively on every region in $R_1, R_2, \ldots, R_t$. In each iteration, the universe size decreases. In particular, after the marking and pruning procedure is done on a region R_i, by Lemma 15, the number of elements in R_i becomes $\mathcal{O}(k^6)$. Let $\mathcal{U}'$ denote the set of elements that remain undeleted from $\mathcal{U}$ after the marking and pruning has been done on all the regions. Since there are at most $3k$ regions, $|\mathcal{U}'| = \mathcal{O}(k^7)$.

We shall now count the number of tests $\mathcal{S}'$. There can be at most 2^{3k} tests that have non-trivial intersection with no region. Given a particular region, the number of tests in $\mathcal{S}'$ that have a non-trivial intersection with that region can be bounded as follows. Each such test is associated with exactly one chunk in that region. Since, in $(\mathcal{U}', \mathcal{S}', k)$, there are at most $\mathcal{O}(k^6)$ chunks in each region, and because there are at most $3k$ regions in total, there can be at most 2^{3k} tests in $\mathcal{S}'$ which have the same chunk in this region. So there are at most $\mathcal{O}(2^{3k} \cdot k^6)$ tests that have non-trivial intersection with one particular region. Since there are at most $3k$ regions, the total number of tests in $\mathcal{S}'$ is at most $\mathcal{O}(2^{3k} \cdot k^7)$.

It is easy to see that the marking and pruning procedure can be done in polynomial time and hence the algorithm runs in polynomial time. ◀

▶ **Theorem 17.** *There is an algorithm that, given an instance $(\mathcal{U}, \mathcal{S}, k)$ of $(n - k)$-Test Cover, runs in time $\mathcal{O}(2^{\mathcal{O}(k^2)}(m + n)^{\mathcal{O}(1)})$ and either returns a test cover of size at most $n - k$ or correctly concludes that no such test cover exists.*

Proof. After generating an equivalent instance of size $\mathcal{O}(2^{3k} \cdot k^7)$ using the kernelization algorithm which runs in time $\mathcal{O}(k^{\mathcal{O}(1)}(m+n)^{\mathcal{O}(1)})$, we can check whether this is a Yes instance by checking for the existence of a k-mini test cover. This can be determined by considering all possible collections of tests containing at most $2k$ tests. There are $\mathcal{O}((2^{3k} \cdot k^7)^{2k})$ such collections and each of them can be checked in time $\mathcal{O}((m + n)^{\mathcal{O}(1)})$, which completes the proof.   ◀

4   Conclusion

In this paper we studied the $(n - k)$-Set Cover and $(n - k)$-Test Cover problems and obtained partially polynomial kernels for both these problems by designing reduction rules that bound the universe size. We showed the utility of these rules and kernels by obtaining a linear time FPT algorithm with an improved dependence on the parameter for $(n - k)$-Set Cover and $(m - k)$-Hitting Set. Furthermore, we also obtained an FPT algorithm for $(\nu(F) + k)$-SAT which improves over the previous best algorithm with respect to the dependence on the parameter, as well as the first single exponential FPT algorithm for $(n - k)$-Test Cover. We believe that in general, the existing upper bounds for other problems can also be improved by using partially polynomial kernels, and in particular, we believe that our reduction rules could be applicable to other covering problems as well.

References

1   Nadja Betzler, Jiong Guo, Christian Komusiewicz, and Rolf Niedermeier. Average parameterization and partial kernelization for computing medians. *J. Comput. Syst. Sci.*, 77(4):774–789, 2011.

2   Hans L. Bodlaender, Rodney G. Downey, Michael R. Fellows, and Danny Hermelin. On problems without polynomial kernels. *J. Comput. Syst. Sci.*, 75(8):423–434, 2009.

3   Robert Crowston, Gregory Gutin, Mark Jones, Gabriele Muciaccia, and Anders Yeo. Parameterizations of test cover with bounded test sizes. *CoRR*, abs/1209.6528, 2012.

4   Robert Crowston, Gregory Gutin, Mark Jones, Venkatesh Raman, Saket Saurabh, and Anders Yeo. Fixed-parameter tractability of satisfying beyond the number of variables. *Algorithmica*, pages , to appear, 2013.

5   Robert Crowston, Gregory Gutin, Mark Jones, Saket Saurabh, and Anders Yeo. Parameterized study of the test cover problem. In *MFCS*, volume 7464, pages 283–295, 2012.

6   Fedor V. Fomin, Serge Gaspers, Saket Saurabh, and Alexey A. Stepanov. On two techniques of combining branching and treewidth. *Algorithmica*, 54(2):181–207, 2009.

7   Lance Fortnow and Rahul Santhanam. Infeasibility of instance compression and succinct pcps for np. *J. Comput. Syst. Sci.*, 77(1):91–106, 2011.

8   Gregory Gutin, Mark Jones, and Anders Yeo. Kernels for below-upper-bound parameterizations of the hitting set and directed dominating set problems. *Theor. Comput. Sci.*, 412(41):5744–5751, 2011.

9   Gregory Gutin, Gabriele Muciaccia, and Anders Yeo. (non-)existence of polynomial kernels for the test cover problem. *Inf. Process. Lett.*, 113(4):123–126, 2013.

10   Jörg Flum and Martin Grohe. *Parameterized Complexity Theory.* Springer-Verlag, 2006.