Parameterizations of Test Cover with Bounded Test Sizes

R. Crowston, G. Gutin, M. Jones and G. Muciaccia

Royal Holloway, University of London
Egham, Surrey, TW20 0EX, UK
{robert|gutin|markj|G.Muciaccia}@cs.rhul.ac.uk

A. Yeo

University of Johannesburg
Auckland Park, 2006 South Africa
andersyeo@gmail.com

October 31, 2018

Abstract

In the TEST COVER problem we are given a hypergraph $H = (V, \mathcal{E})$ with $|V| = n, |\mathcal{E}| = m$, and we assume that $\mathcal{E}$ is a *test cover*, i.e. for every pair of vertices x_i, x_j, there exists an edge $e \in \mathcal{E}$ such that $|\{x_i, x_j\} \cap e| = 1$. The objective is to find a minimum subset of $\mathcal{E}$ which is a test cover. The problem is used for identification across many areas, and is NP-complete. From a parameterized complexity standpoint, many natural parameterizations of TEST COVER are either $W[1]$-complete or have no polynomial kernel unless $coNP \subseteq NP/poly$, and thus are unlikely to be solveable efficiently.

However, in practice the size of the edges is often bounded. In this paper we study the parameterized complexity of TEST-r-COVER, the restriction of TEST COVER in which each edge contains at most $r \geq 2$ vertices. In contrast to the unbounded case, we show that the following below-bound parameterizations of TEST-r-COVER are fixed-parameter tractable with a polynomial kernel: (1) Decide whether there exists a test cover of size $n - k$, and (2) decide whether there exists a test cover of size $m - k$, where k is the parameter. In addition, we prove a new lower bound $\lceil \frac{2(n-1)}{r+1} \rceil$ on the minimum size of a test cover when the size of each edge is bounded by r. TEST-r-COVER parameterized above this bound is unlikely to be fixed-parameter tractable; in fact, we show that it is para-NP-complete, as it is NP-hard to decide whether an instance of TEST-r-COVER has a test cover of size exactly $\frac{2(n-1)}{r+1}$.

1

1   Introduction

The input to the TEST COVER problem consists of a hypergraph $H = (V, \mathcal{E})$, with vertex set $V = \{x_1, \ldots, x_n\}$ and edge set $\mathcal{E} = \{e_1, \ldots, e_m\}$[1]. We say that an edge e_q *separates* a pair of vertices x_i, x_j if $|\{x_i, x_j\} \cap e_q| = 1$. A subcollection $\mathcal{T} \subseteq \mathcal{E}$ is a *test cover* if each pair of distinct vertices x_i, x_j is separated by an edge in $\mathcal{T}$. The objective is to find a test cover of minimum size, if one exists. Since it is easy to decide, in polynomial time, whether the collection $\mathcal{E}$ itself is a test cover, henceforth we will assume that $\mathcal{E}$ is a test cover.

TEST COVER arises naturally in the following general setting of identification problems: Given a set of items (which corresponds to the set of vertices) and a set of binary attributes that may or may not occur in each item, the aim is to find the minimum size subset of attributes (corresponding to a minimum test cover) such that each item can be uniquely identified from the list of attributes it has from this subset. TEST COVER arises in fault analysis, medical diagnostics, pattern recognition and biological identification (see, e.g., [17, 18, 22]).

The TEST COVER problem has also been studied extensively from an algorithmic viewpoint. The problem is NP-hard, as was shown by Garey and Johnson [14]. There is an $O(\log n)$-approximation algorithm for the problem [22] and there is no $o(\log n)$-approximation algorithm unless P=NP [17]. Often, in practice, the sizes of all edges are bounded from above by a constant $r \geq 2$ [9, 17]. For such cases the problem remains NP-hard, even when $r = 2$, but the approximation guarantee can be improved to $O(\log r)$ [17].

Initially, research in parameterized algorithmics[2] considered mainly standard parameterizations, where the parameter is the size of the solution. In the last decade, this situation has changed: many papers deal with structural parameters such as treewidth or parameterizations above and below tight bounds. Research on parameterizations above and below tight bounds was initiated by Mahajan and Raman [20], but the systematic study of such parameterizations started only after the publication of [21]. This resulted in new methods and approaches, solving most of the open problems stated in [20] and [21], see, e.g., [1, 4, 6, 7, 15].

The complexity of the following four parameterizations of TEST COVER were first studied in [5, 16], in all of which k is the parameter:

TESTCOVER(k): Is there a test cover with at most k edges?

TESTCOVER($n - k$): Is there a test cover with at most $n - k$ edges?

TESTCOVER($m - k$): Is there a test cover with at most $m - k$ edges?

TESTCOVER($\log n + k$): Is there a test cover with at most $\log n + k$ edges?

Whilst TESTCOVER(k) is a standard parameterization, TESTCOVER($n-k$) and TESTCOVER($m-k$) are parameterizations below tight upper bounds: clearly m is a tight upper bound on the minimum size of a test cover and it is not hard to

---

[1]Notice that, in the literature, the vertices are also called *items* and the edges *tests*. We provide basic terminology and notation on hypergraphs in Section 2.

[2]We provide basic notions on parameterized algorithmics in the end of this section.

prove that $n-1$ is another tight upper bound [3] (see also Corollary 1). Finally, TESTCOVER($\log n + k$) is a parameterization above a tight lower bound $\lceil \log n \rceil$ on the minimum size of a test cover [17].

It is clear that TESTCOVER(k) is fixed-parameter tractable; it follows immediately from the tight lower bound $\lceil \log n \rceil$. It is proved in [5] that TESTCOVER($n-k$) is fixed-parameter tractable. The proof of this result is quite nontrivial and yields an algorithm whose runtime is $f(k)(n+m)^{O(1)}$, where $f(k)$ is a function growing impractically fast. It is also proved in [5] that TESTCOVER($m-k$) and TESTCOVER($\log n + k$) are W[1]-hard. Since TESTCOVER(k) and TESTCOVER($n-k$) are fixed-parameter tractable, it is natural to ask whether they admit polynomial-size kernels. The authors of [16] answered this question by proving that, unless $coNP \subseteq NP/poly$, neither TESTCOVER(k) nor TESTCOVER($n-k$) admits a polynomial-size kernel.

Thus, papers [5, 16] demonstrated that the four parameterizations of TEST COVER are not easy to solve: even those of them which are fixed-parameter tractable do not admit polynomial-size kernels. This gives another theoretical explanation of the fact that TEST COVER is not easy to solve in practice [8, 12]. Thus, it is natural to study important special cases of TEST COVER. It turns out that often the sizes of all edges are bounded from above by a constant r: De Bontridder *et al.* [9] observed that "this is the common restriction" for the problem and provide, as an example, protein identification. A question is whether the parameterizations of TEST COVER become easier in this case. Already [16] indicated that this can be true by proving that TESTCOVER(k) does admit a polynomial-size kernel if r is a constant. We will denote the special cases of the four parameterizations by TEST-r-COVER(k), TEST-r-COVER($n-k$), TEST-r-COVER($m-k$), and TEST-r-COVER($\log n + k$). Since the case $r=1$ is trivial, henceforth we will assume $r \geq 2$.

In this paper we will prove that not only is TEST-r-COVER($m-k$) fixed-parameter tractable, but TEST-r-COVER($m-k$) and TEST-r-COVER($n-k$) also admit polynomial-size kernels. There is no interest in studying TEST-r-COVER($\log n + k$) as $\lceil \log n \rceil$ is no longer a tight lower bound. Instead, we prove that $\lceil \frac{2(n-1)}{r+1} \rceil$ is a tight lower bound and study the parameterization TEST-r-COVER($\frac{2(n-1)}{r+1} + k$): Is there a test cover with at most $\frac{2(n-1)}{r+1} + k$ edges, where k is the parameter? We prove that it is NP-hard even to decide whether there is a test cover of size $\frac{2(n-1)}{r+1}$. It follows that TEST-r-COVER($\frac{2(n-1)}{r+1} + k$) is para-NP-complete. Note that lower bounds play a key role in designing branch-and-bound algorithms for TEST COVER [8, 12] and so our new lower bound may be of interest for practical solutions of TEST-r-COVER.

The rest of the paper is organized as follows. In the remainder of this section we provide necessary basic terminology on parameterized algorithmics. In Section 2, we provide necessary terminology on hypergraphs and prove some preliminary results. In Section 3, we show that $\lceil \frac{2(n-1)}{r+1} \rceil$ is a tight lower bound and that TEST-r-COVER($\frac{2(n-1)}{r+1} + k$) is para-NP-complete. In Section 4, we give a short proof that TEST-r-COVER($m-k$) is fixed-parameter tractable. In Sections 5 and 6, we prove that TEST-r-COVER($m-k$) and TEST-r-COVER($n-$

k) admit polynomial-size kernels. In Section 7, we discuss open problems.

Basics on Parameterized Complexity. A parameterized problem Π can be considered as a set of pairs (I, k) where I is the *problem instance* and k (usually a nonnegative integer) is the *parameter*. Π is called *fixed-parameter tractable (fpt)* if membership of (I, k) in Π can be decided by an algorithm of runtime $O(f(k)|I|^c)$, where $|I|$ is the size of I, $f(k)$ is an arbitrary function of the parameter k only, and c is a constant independent from k and I. Let Π and Π' be parameterized problems with parameters k and k', respectively. An *fpt-reduction R from Π to Π'* is a many-to-one transformation from Π to Π' that maps an instance (I, k) of Π to an instance (I', k') of Π', such that (i) $(I, k) \in \Pi$ if and only if $(I', k') \in \Pi'$ with $k' \leq g(k)$ for a fixed function g, and (ii) R is of complexity $O(f(k)|I|^c)$.

Π is in *para-NP* if membership of (I, k) in Π can be decided by a nondeterministic Turing machine in time $O(f(k)|I|^c)$, where $|I|$ is the size of I, $f(k)$ is an arbitrary function of the parameter k only, and c is a constant independent from k and I. A parameterized problem Π' is *para-NP-complete* if it is in para-NP and for any parameterized problem Π in para-NP there is an fpt-reduction from Π to Π'. It is well-known that a parameterized problem Π belonging to para-NP is para-NP-complete if we can reduce an NP-complete problem to the subproblem of Π when the parameter is equal to some constant [13].

Given a parameterized problem Π, a *kernelization of* Π is a polynomial-time algorithm that maps an instance (I, k) to an instance (I', k') (the *kernel*) such that (i) $(I, k) \in \Pi$ if and only if $(I', k') \in \Pi$, and (ii) $|I'| + k' \leq g(k)$ for some function g of k only. We call $g(k)$ the *size* of the kernel. It is well-known [11, 13] that a decidable parameterized problem Π is fixed-parameter tractable if and only if it has a kernel. Polynomial-size kernels are of main interest, due to applications [11, 13, 23], but unfortunately not all fixed-parameter problems have such kernels unless coNP⊆NP/poly, see, e.g., [2, 10].

2 Additional terminology, notation and preliminaries

We use usual hypergraph terminology. Let $H = (V, \mathcal{E})$ be a hypergraph. The degree of a vertex x is the cardinality of $\{e \in \mathcal{E} : x \in e\}$. Given $X \subseteq V$, define the neighborhood $N_1(X) = \{u \in V \setminus X : \exists e \in \mathcal{E}\ \exists x \in X \text{ such that } \{x, u\} \subseteq e\}$, $N_1[X] = N_1(X) \cup X$ and $N_j[X] = N_1[N_{j-1}[X]]$. Given $\mathcal{F} \subseteq \mathcal{E}$, define the neighborhood $N_1(\mathcal{F}) = \{e \in \mathcal{E} \setminus \mathcal{F} : \exists f \in \mathcal{F},\ f \cap e \neq \emptyset\}$, $N_1[\mathcal{F}] = N_1(\mathcal{F}) \cup \mathcal{F}$ and $N_j[\mathcal{F}] = N_1[N_{j-1}[\mathcal{F}]]$. For $\mathcal{F} \subseteq \mathcal{E}$, $H[\mathcal{F}]$ is the hypergraph with vertex set $V(H[\mathcal{F}]) = \{x \in V : \exists e \in \mathcal{F} \text{ such that } x \in e\}$ and edge set $\mathcal{E}(H[\mathcal{F}]) = \mathcal{F}$. Similarly, given a set of vertices $X \subseteq V$, $H[X]$ is the hypergraph with vertex set $V(H[X]) = X$ and edge set $\mathcal{E}(H[X]) = \{e \cap X : e \in \mathcal{E}, e \cap X \neq \emptyset\}$.

We reuse some terminology from [5]. For a subset $\mathcal{T} \subseteq \mathcal{E}$, the *classes induced by $\mathcal{T}$* are the maximal subsets C_i of V such that no pair of vertices in C_i is separated by an edge in $\mathcal{T}$. Observe that the classes induced by $\mathcal{T}$ form a

partition of V.

We say that $\mathcal{E}$ *separates* $X \subseteq V$ and $Y \subseteq V$, $X \cap Y = \emptyset$, if for every pair (x, y), with $x \in X$ and $y \in Y$, x and y are separated by an edge of $\mathcal{E}$. We say that $\mathcal{E}$ *isolates* $X \subseteq V$ if it separates X and $V \setminus X$. We say an edge e *cuts* $X \subseteq V$ if $X \cap e \neq \emptyset$ and $X \setminus e \neq \emptyset$.

For an integer t, we use $[t]$ to denote the set $\{1, 2, \ldots, t\}$.

Lemma 1. *If $\mathcal{E}$ induces $t \geq 2$ classes in a hypergraph $H = (V, \mathcal{E})$ and $i \in [t-1]$ then there is a subset $\mathcal{F}$ of $\mathcal{E}$ with i edges that induces at least $i + 1$ classes.*

Proof. By induction on $i \in [t-1]$. To see that the lemma holds for $i = 1$ set $\mathcal{F} = \{e\}$, where e is any edge of $\mathcal{E}$ with less than $|V|$ vertices. Let $\mathcal{F}$ be a subset of $\mathcal{E}$ with $i - 1$ edges that induces at least i classes, let x, y be vertices separated by $\mathcal{E}$ not separated by $\mathcal{F}$, and let e be an edge separating x and y. It remains to observe that $\mathcal{F} \cup \{e\}$ induces at least $i + 1$ classes. $\square$

Observe that $\mathcal{E}$ is a test cover if and only if it induces n classes. This and the lemma above imply the following:

Corollary 1. *[3] If $\mathcal{E}$ is a test cover in a hypergraph $H = (V, \mathcal{E})$ then there is a subset $\mathcal{F}$ of $\mathcal{E}$ with at most $n - 1$ edges which is also a test cover.*

Corollary 2. *In a hypergraph $H = (V, \mathcal{E})$, let $X, Y \subseteq V$ be such that $X \cap Y = \emptyset$ and $\mathcal{E}$ separates X and Y. Then there is a subset $\mathcal{F}$ of $\mathcal{E}$ that separates X and Y and has at most $t_X + t_Y - 1$ edges, where t_X (t_Y) is the number of classes induced by $\mathcal{E}$ that intersect X (Y).*

Proof. Apply Lemma 1 to $H[X \cup Y]$ and then extend the obtained edges of $H[X \cup Y]$ beyond $X \cup Y$ such that they correspond to edges of H. $\square$

3 TEST-r-COVER($\frac{2(n-1)}{r+1} + k$)

Proposition 3. *If the size of every edge is at most r, then every test cover has at least $\lceil \frac{2(n-1)}{r+1} \rceil$ edges. This lower bound on the size of a test cover is tight. Furthermore, for a test cover of size exactly the lower bound, every vertex is contained in at most two edges.*

Proof. We first prove that $\lceil \frac{2(n-1)}{r+1} \rceil$ is indeed a lower bound. Given a test cover of size m', observe that at most one vertex may be contained in no edges. For each of the m' edges, at most one vertex is contained in only that edge and every other vertex is contained in at least two edges. Hence $n \leq 1 + m' + \frac{m'(r-1)}{2}$, which implies $m' \geq \frac{2(n-1)}{r+1}$. Observe that no vertex being contained in three edges is a necessary condition for the bound to be tight.

To see that this bound is tight, consider a set $V = \{x_{i,j} : i, j \in [r]\}$ of vertices, and a set $\mathcal{E} = \{e_q : q \in [r-1]\} \cup \{e'_s : s \in [r-1]\}$ of edges, where $e_q = \{x_{q,j} : j \in [r]\}$, $e'_s = \{x_{i,s} : i \in [r]\}$ (see Figure 1). Since $n = r^2$, we have $|\mathcal{E}| = 2(r-1) = \frac{2(n-1)}{r+1}$. Consider two vertices $x_{i,j}$ and $x_{i',j'}$. If $i \neq i'$, then

5

$x_{i,j}$ and $x_{i',j'}$ are separated by $e_{\min\{i,i'\}}$ and if $j \neq j'$, then $x_{i,j}$ and $x_{i',j'}$ are separated by $e'_{\min\{j,j'\}}$. Thus, $\mathcal{E}$ is a test cover of minimum possible size. By using multiple copies of this construction (except for $x_{r,r}$), one can see that the bound is tight for arbitrarily large n. $\square$
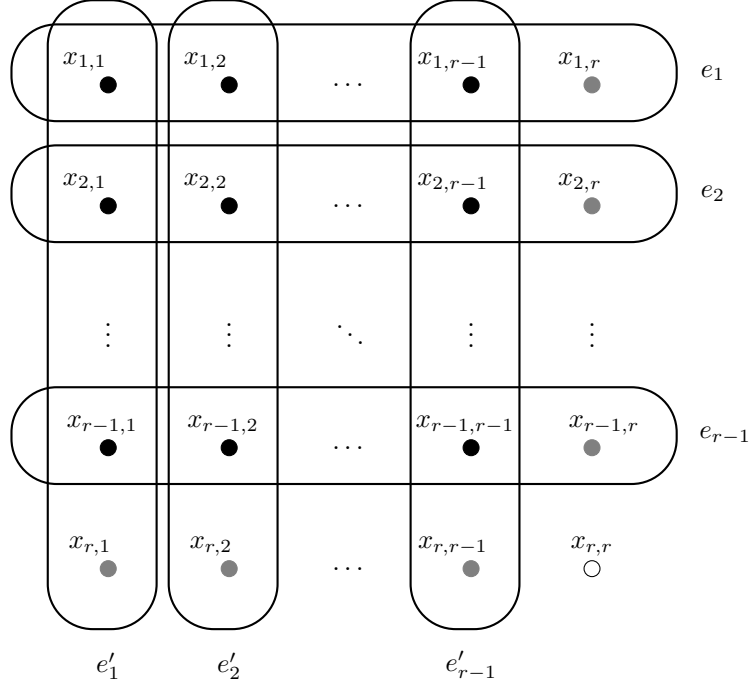

Figure 1: Illustration of the hypergraph $(V, \mathcal{E})$ in Proposition 3. Vertices of degree two are labelled in black, vertices of degree one in gray and the vertex of degree zero in white.

From this lower bound, a kernel for TEST-r-COVER(k), with less vertices than in [16], immediately follows:

Corollary 4. TEST-r-COVER(k) *admits a kernel with at most $k(r+1)/2 + 1$ vertices.*

It follows from the proof of Proposition 3 that for a test cover of size $\frac{2(n-1)}{r+1}$, there must be exactly one vertex contained in no edges, each edge must contain one vertex of degree 1, and every other vertex must be of degree 2.

To prove the next lemma we give a reduction from the r-DIMENSIONAL MATCHING problem, which is one of Karp's 21 NP-complete problems [19], for each $r \geq 3$. To state the problem, we will give the following definitions. A collection $\mathcal{E}$ of edges of a hypergraph H is called a *matching* if no two edges of $\mathcal{E}$ have a common vertex. A matching is *perfect* if every vertex of H belongs to an

edge of the matching. A hypergraph H is *r-partite r-uniform* if the vertex set $V(H)$ of H can be partitioned into r sets $X_1, \ldots, X_r$ such that each edge of H has exactly one vertex from each X_i. In *r-Dimensional Matching*, given an r-partite r-uniform hypergraph H, the aim is to decide whether H has a perfect matching.

Lemma 2. Test-r-Cover($\frac{2(n-1)}{r+1} + k$) *is NP-hard for $k = 0$ and $r \geq 3$.*

Proof. We give a reduction from r-Dimensional Matching to Test-r-Cover($\frac{2(n-1)}{r+1} + k$) for $k = 0$. We assume we have an instance of r-Dimensional Matching given by an r-partite r-uniform hypergraph G with vertex set $V(G) = \{x_{i,j} : i \in [r], j \in [n']\}$, where $X_i = \{x_{i,j} : j \in [n']\}$ form the partition of $V(G)$, i.e. each edge has exactly one vertex in each X_i. (We may assume all X_i have the same size, as otherwise there is no perfect matching.) The edge set of G will be denoted by $\mathcal{E}(G)$. Additionally, we may assume that n' is divisible by $r - 1$.

We now give the construction of $(V, \mathcal{E})$, the Test-r-Cover instance. Let the vertex set $V = V(G) \cup Y$, and the edge set $\mathcal{E} = \mathcal{E}(G) \cup \mathcal{E}'$, where Y and $\mathcal{E}'$ are defined as follows:

$$Y = \{y_{i,j} : i \in [r-1], j \in \{(r-1), 2(r-1), ..., n'\}\} \cup \{y_0\}$$

$$\mathcal{E}' = \{e_{i,p} : i \in [r-1], p \in \{(r-1), 2(r-1), ..., n'\}\}$$

where $e_{i,p} = \{x_{i,p-r+2}, x_{i,p-r+3}, \ldots, x_{i,p}, y_{i,p}\}$ (see Figure 2). Note that $n = |V| = rn' + n' + 1$.

We first show that if $\mathcal{E}'' \subseteq \mathcal{E}(G)$ is a perfect matching in G, then $\mathcal{E}' \cup \mathcal{E}''$ is a test cover of size $\frac{2(n-1)}{r+1}$. For any pair $x_{i,j}, x_{i',j'} \in V(G)$, if $i = i'$ then $x_{i,j}, x_{i',j'}$ appear in different edges in $\mathcal{E}''$ and are separated, if $i < i'$ they are separated by $e_{i,(r-1)\lceil \frac{j}{r-1} \rceil}$, and if $i > i'$ they are separated by $e_{i',(r-1)\lceil \frac{j'}{r-1} \rceil}$. Any pair $y_{i,j}, y_{i',j'} \in Y$ is separated by $e_{i,j}$. As $\mathcal{E}'$ covers $V(G)$, every $x_{i,j} \in V(G)$ is separated from every $y_{i',j'} \in Y$. Finally, y_0 is separated from every other vertex as it is the only vertex not covered by $\mathcal{E}' \cup \mathcal{E}''$. Thus, $\mathcal{E}' \cup \mathcal{E}''$ is a test cover. Since $|\mathcal{E}' \cup \mathcal{E}''| = 2n'$ and $n = rn' + n' + 1$, we have that $\mathcal{E}' \cup \mathcal{E}''$ is a test cover of size $\frac{2(n-1)}{r+1}$.

It remains to show that if $(V, \mathcal{E})$ has a test cover of size $\frac{2(n-1)}{r+1}$, then G has a perfect matching. Firstly, observe that to separate y_0 from $y_{i,p}$, every edge $e_{i,p}$ must be in the test cover. Hence $\mathcal{E}'$ is contained in any test cover for $(V, \mathcal{E})$. So a test cover is a set $\mathcal{E}' \cup \mathcal{E}''$, where $\mathcal{E}'' \subseteq \mathcal{E}(G)$. Next, observe that for every vertex $x_{i,j}$, there must exist an edge in $\mathcal{E}''$ containing $x_{i,j}$, as otherwise there would be no edge separating $x_{i,j}$ from $y_{i,(r-1)\lceil \frac{j}{r-1} \rceil}$.

We may also observe that no two edges in $\mathcal{E}''$ intersect. Suppose two edges intersected, but not in partite set X_r. Then there is another edge in $\mathcal{E}'$ also intersecting these edges at the same vertex. But we know from Proposition 3 that in a test cover of size $\frac{2(n-1)}{r+1}$, at most two edges intersect at any vertex, so this cannot happen. Suppose instead, that edges e and e' intersect in partite set X_r. Then e also intersects an edge in $\mathcal{E}'$ in each of the other partitions, so

7

every vertex in e is of degree 2. But we know that in a test cover of size $\frac{2(n-1)}{r+1}$, one vertex in each edge has degree 1, so this case is also not possible. Hence, a test cover with $\frac{2(n-1)}{r+1}$ edges for $(V, \mathcal{E})$ would give a perfect matching in G. $\quad\square$
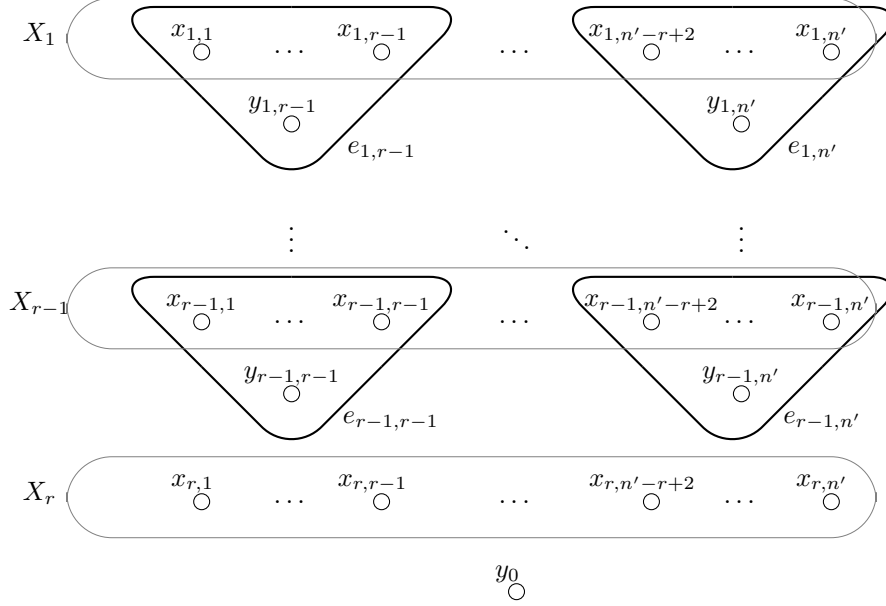

Figure 2: Illustration of the edge set $\mathcal{E}'$ in Lemma 2. Edges of $\mathcal{E}'$ are in black. The sets $X_1, \ldots X_r$ are in gray. Edges of $\mathcal{E}(G)$ (not depicted) contain one vertex from each of $X_1, \ldots X_r$.

Lemma 3. TEST-2-COVER$(2(n-1)/3 + k)$ *is NP-complete for $k = 0$.*

Proof. We will show NP-completeness of our problem by reduction from the P_3-packing problem, which asks, given a graph G with n vertices, n divisible by 3, whether G has $n/3$ vertex-disjoint copies of P_3. The problem is NP-complete [14]. Consider the graph H obtained from G by adding to it an isolated vertex x. We will view H as an instance of TEST-2-COVER and we will prove that G is a YES-instance of the P_3-packing problem if and only if H has a test cover of size $2n/3$. If G contains a set of edges F forming $n/3$ disjoint copies of P_3, then observe that F forms a test cover in H of size $2n/3$. Now assume that H has a test cover F of size $2n/3$. We will prove that G is a YES-instance of the P_3-packing problem. Observe that every vertex of G has positive degree in $G[F]$. Let n' be the number of vertices of degree 1 in $G[F]$. We know that no vertex can have degree larger than 2, as otherwise F must have more than $2n/3$ edges. Since $|F| = 2n/3$, we have $n' + 2(n - n') = 2|F| = 4n/3$ and so $n' = 2n/3$. Finally, F cannot have any isolated edges as it is a test cover. Thus, F is a collection of $n/3$ vertex-disjoint copies of P_3. $\quad\square$

The next theorem follows from the straightforward fact that TEST-r-COVER($\frac{2(n-1)}{r+1} +$ k) is in para-NP and the previous two lemmas.

Theorem 5. TEST-r-COVER($\frac{2(n-1)}{r+1} + k$) *is para-NP-complete for each fixed* $r \geq 2$.

4   A short FPT proof for TEST-r-COVER($m - k$)

In this section, we show that TEST-r-COVER($m-k$) is fixed-parameter tractable. The proof is short but does not lead to a polynomial kernel; this requires more work, and is the subject of the next section.

Theorem 6. *There is an algorithm for* TEST-r-COVER($m - k$) *that runs in time* $(r^2 + 1)^k (n + m)^{O(1)}$.

Proof. We first need to guess whether there will be a vertex not contained in any edge in the solution, and if so, which vertex it will be. If there already exists a vertex y_0 not in any edge in $\mathcal{E}$, then we are done. Otherwise, either pick a vertex x or guess that every vertex in V will be covered by the solution. If a vertex x is picked, delete all the edges containing x, and reduce k by the number of deleted edges. If it is guessed that every vertex in V will be covered by the solution, add a new vertex y_0 which is not in any edge. Observe that this does not change the solution to the problem. By doing this we have split the problem into $n + 1$ separate instances, with each instance containing an isolated vertex. Thus we may now assume that there exists a vertex y_0 which is not contained in any edge in $\mathcal{E}$.

Consider an edge $e \in \mathcal{E}$, and suppose that $\mathcal{E} \setminus \{e\}$ is a test cover. Let $\mathcal{B}_0$ be a minimal set of edges in $\mathcal{E} \setminus \{e\}$ which covers e. Note that such a set must exist, as otherwise x is not separated from y_0 in $\mathcal{E} \setminus \{e\}$ for some $x \in e$, and so $\mathcal{E} \setminus \{e\}$ is not a test cover. Furthermore, we may assume $|\mathcal{B}_0| \leq |e| \leq r$. Now for each $b \in \mathcal{B}_0$, let $\mathcal{B}_b$ be a minimal set of edges in $\mathcal{E} \setminus \{e\}$ separating every vertex in $b \setminus e$ from every vertex in $b \cap e$. By Corollary 2, we may assume that $|\mathcal{B}_b| \leq r - 1$.

Now let $\mathcal{B} = \mathcal{B}_0 \cup (\bigcup_{b \in \mathcal{B}_0} \mathcal{B}_b)$, and observe that $\mathcal{B}$ isolates the vertex set of e. Thus, in any solution with minimum number of edges, at least one edge from $\mathcal{B} \cup \{e\}$ will be missing. Note that $|\mathcal{B}| \leq r + r(r - 1) = r^2$.

We now describe a depth-bounded search tree algorithm for TEST-r-COVER($m - k$). If $\mathcal{E}$ is not a test cover, return NO. Otherwise if $k = 0$ return YES. Otherwise, for each edge $e \in \mathcal{E}$ check whether $\mathcal{E} \setminus \{e\}$ is a test cover. If for all $e \in \mathcal{E}$, $\mathcal{E} \setminus \{e\}$ is not a test cover, then a test cover must contain all m edges and so we return NO. Otherwise, let e be an edge such that $\mathcal{E} \setminus \{e\}$ is a test cover, and construct the set $\mathcal{B}$ as described above. Then we may assume one of $\mathcal{B} \cup \{e\}$ is not in the solution. Thus we may pick one edge from $\mathcal{B} \cup \{e\}$, delete it, and reduce k by 1. So we split into $r^2 + 1$ instances with reduced parameter.

We therefore have a search tree with at most $(r^2 + 1)^k$ leaves. As every internal node has at least 2 children, the total number of nodes is at most

9

$2(r^2 + 1)^k - 1$. Note also that guessing the isolated vertex at the start split the problem into $n+1$ instances, so there are at most $(n+1)(2(r^2+1)^k-1)$ nodes to compute in total. As each node in the tree takes polynomial time to compute, we have an algorithm with total running time $(r^2 + 1)^k (n + m)^{O(1)}$. ☐

5 Polynomial Kernel for TEST-r-COVER$(m - k)$

In this section, we show that TEST-r-COVER$(m-k)$ admits a polynomial kernel. It will be useful to consider a slight generalization of TEST-r-COVER$(m - k)$, which we call SUBSET-TEST-r-COVER$(m - k)$. In this problem, we are given a special subset $\mathcal{B} \subseteq \mathcal{E}$ of edges which are required to be in the solution. For convenience we will say these edges are *colored black*.

We begin with the following reduction rules, which must be applied whenever possible:

Reduction Rule 1. Given a vertex x of degree 1 and a black edge b which contains only x, delete b from $\mathcal{B}$ and $\mathcal{E}$, delete x and leave k the same.

Reduction Rule 2. Given a black edge b, if there exists any other edge e such that $b \subset e$, then replace e with $e \setminus b$. If there exists a black edge b' such that b cuts b' and b' cuts b, delete b and b' and add black edges $b \setminus b'$, $b' \setminus b$ and $b \cap b'$. Leave k the same.

Lemma 4. *Let $(V, \mathcal{E}', \mathcal{B}', k)$ be an instance of SUBSET-TEST-r-COVER$(m - k)$ derived from $(V, \mathcal{E}, \mathcal{B}, k)$ by an application of Rule 1 or 2. Then $(V, \mathcal{E}', \mathcal{B}', k)$ is a YES-instance if and only if $(V, \mathcal{E}, \mathcal{B}, k)$ is a YES-instance.*

Proof. We will show for each rule that for any t, $(V, \mathcal{E}, \mathcal{B}, k)$ has a solution of size $|\mathcal{E}| - k$ if and only if $(V, \mathcal{E}', \mathcal{B}', k)$ has a solution of size $|\mathcal{E}'| - k$.

Rule 1: Suppose $(V, \mathcal{E}', \mathcal{B}', k)$ is a YES-instance, with solution $\mathcal{T}'$. Then observe that $\mathcal{T} = \mathcal{T}' \cup \{b\}$ is a solution for $(V, \mathcal{E}, \mathcal{B}, k)$. Conversely, if $\mathcal{T}$ is a solution for $(V, \mathcal{E}, \mathcal{B}, k)$ then $\mathcal{T}$ must contain b, and $\mathcal{T} \setminus \{b\}$ is a solution for $(V, \mathcal{E}', \mathcal{B}', k)$.

Rule 2: First consider the case when $b \subset e$ for some other edge e. It is sufficient to show that for any $\mathcal{T} \subseteq \mathcal{E}$ containing e and b, $\mathcal{T}$ is a test cover if and only if $(\mathcal{T} \setminus \{e\}) \cup \{e \setminus b\}$ is a test cover. To see this, observe that for any $x \in e, y \notin e$, x and y are separated either by b or $e \setminus b$, and for any $x \in e \setminus b$, $y \notin e \setminus b$, x and y are separated either by b or e.

Now consider the case when b, b' are intersecting black edges. Similar to the previous case, if x and y are separated by one of b, b' then they are also separated by at least one of $b \setminus b'$, $b' \setminus b$, $b \cap b'$, and if they are separated by one of $b \setminus b'$, $b' \setminus b$, $b \cap b'$ then they are also separated by at least one of b, b'. ☐

Lemma 5. *Let $(V, \mathcal{E}, \mathcal{B}, k)$ be an instance irreducible by Rules 1 and 2. The instance can be reduced, in polynomial time, to an equivalent instance such that every vertex has degree at most kr^2.*

Proof. Assume that there exists a vertex x with degree in $\mathcal{E}$ greater than kr^2. We will be able to produce an equivalent instance in which either k or the degree of x is reduced. Clearly this reduction can only take place a polynomial number of times, so in polynomial time we will reduce to an instance in which every vertex has degree bounded by kr^2.

We produce a special set $\widetilde{X}$, such that $\widetilde{X}$ is still isolated when at most k edges are deleted, according to the following algorithm.

Set $\widetilde{\mathcal{E}} = \mathcal{E}$, $i = 1$, $X = \{x\}$, $j = 1$;
while $i \leq k + 1$ **do**
    if $\widetilde{\mathcal{E}}$ *isolates* X **then**
        Let e_i be an edge containing X, and construct a set $E_i \subseteq \widetilde{\mathcal{E}}$ such that $E_i \cup \{e_i\}$ isolates X and $|E_i| \leq r - 1$;
        Set $\widetilde{\mathcal{E}} = \widetilde{\mathcal{E}} \setminus (E_i \cup \{e_i\})$;
        Set $i = i + 1$;
    else
        Let X' be the class induced by $\widetilde{\mathcal{E}}$ containing X;
        Set $X = X', i = 1, j = j + 1$;
    end
end
Set $\widetilde{X} = X$.

Observe that throughout the algorithm, by construction any edge in $\widetilde{\mathcal{E}}$ which contains x also contains X as a subset, $|X| \geq j$ and $\widetilde{\mathcal{E}} \subseteq \mathcal{E}$. Notice that if the algorithm ever sets $j = r + 1$, then at that point at most kr^2 edges have been deleted from $\widetilde{\mathcal{E}}$, and as $|X| \geq r + 1$, no remaining edges in $\widetilde{\mathcal{E}}$ contain x. But this is a contradiction as the degree of x is greater than kr^2. Therefore we may assume the algorithm never reaches $j = r + 1$. Hence the algorithm must terminate for some $j \leq r$.

We now show that we can always find e_i and E_i for $i \leq k + 1$. Since x has degree greater than kr^2 and at most $kr(r - 1)$ edges are removed from $\widetilde{\mathcal{E}}$ earlier in the algorithm, we can always find an edge e_i containing x and therefore containing X. To see that E_i can be constructed using at most $r - 1$ edges, apply Corollary 2 to X and $e_i \setminus X$.

Now consider the set $\widetilde{X}$ formed by the algorithm. When the algorithm terminated, e_i and E_i were found for all $i \leq k + 1$. If we remove k arbitrary edges from $\mathcal{E}$, it is still possible to find i such that no edges in $E_i \cup \{e_i\}$ have been deleted. This means that as long as we delete at most k edges, $\widetilde{X}$ is still isolated. Therefore if $\widetilde{X}$ is an edge in $\mathcal{E}$, delete it and reduce k by 1. If $\widetilde{X}$ is not an edge in $\mathcal{E}$, add a new black edge $\widetilde{X}$ to $\mathcal{E}$ and $\mathcal{B}$, keeping k the same, and apply Rule 1 and 2. Observe that since $\widetilde{X}$ is properly contained in at least two edges, before adding the black edges, this will decrease the degree of every vertex in $\widetilde{X}$. $\qquad\square$

Now assume that $(V, \mathcal{E}, \mathcal{B}, k)$ is reduced by Rules 1 and 2 and that every vertex has degree at most kr^2. We will color the uncolored edges in $\mathcal{E}$ as follows.

For every edge e which is not black, if $\mathcal{E} \setminus \{e\}$ is not a test cover, color e black, adding it to $\mathcal{B}$ (and apply Rules 1 and 2). If $\mathcal{E} \setminus \{e\}$ is a test cover and e contains a degree one vertex, color e orange. Otherwise, color e green.

Remark 1. Notice that an edge is colored orange only if there is no isolated vertex.

Lemma 6. *If G is a set of green edges such that, for every pair $g_1, g_2 \in G$, $N_1[g_1] \cap N_1[g_2]$ is empty, then $\mathcal{E} \setminus G$ is a test cover.*

Proof. We proceed by induction on $|G|$. If $|G| = 1$ this is obviously true. If $|G| = j + 1$, delete the first j edges and consider the last one, denoted g. The only problem that could occur removing g is that a vertex $x \in g$ may no longer be separated from another vertex y. If y is not in one of the edges in $G \setminus \{g\}$, then x and y are not separated even by $\mathcal{E} \setminus \{g\}$, which is a contradiction since g is green. Therefore, denote by g' the edge in G which contains y. The degree of y is at least 2, hence there exists an edge different from g' which contains y; this edge cannot contain x too, or $N_1[g] \cap N_1[g']$ would not be empty. This ensures that x and y are separated by $\mathcal{E} \setminus G$. $\square$

Reduction Rule 3. Given an orange edge o, if $N_2[o]$ contains no green edges, delete o and decrease k by 1. (Notice that this creates an isolated vertex, which means that every other orange edge will become black.)

Lemma 7. *Let $(V, \mathcal{E}', \mathcal{B}', k-1)$ be an instance of* SUBSET-TEST-r-COVER$(m-k)$ *derived from $(V, \mathcal{E}, \mathcal{B}, k)$ by an application of Rule 3. Then $(V, \mathcal{E}', \mathcal{B}', k-1)$ is a* YES-*instance if and only if $(V, \mathcal{E}, \mathcal{B}, k)$ is a* YES-*instance.*

Proof. Let $(V, \mathcal{E}, \mathcal{B}, k)$ be a YES-instance, and suppose there is an orange edge o such that $N_2[o]$ contains no green edges. It is sufficient to prove that there exists a solution that does not include o.

Suppose $\mathcal{T} \subseteq \mathcal{E}$ is a solution and suppose it contains o. If there is a vertex x which is not contained in any edge of $\mathcal{T}$, consider an edge $e \in \mathcal{E}$ which contains it (which exists by Remark 1). If the isolated vertex x does not exist, take any edge e from $\mathcal{E} \setminus \mathcal{T}$.

Consider $\mathcal{T}' = (\mathcal{T} \setminus \{o\}) \cup \{e\}$. We claim that $\mathcal{T}'$ is still a test cover.

First of all, note that if there is an orange edge $o' \in \mathcal{E} \setminus \mathcal{T}$, this edge must be e: removing o' creates an isolated vertex and o' is the only edge containing that vertex, which means that o' is the edge that we add to make $\mathcal{T}'$. Therefore, we may assume $(N_2[o] \setminus \{o\}) \subseteq \mathcal{T}'$.

Now, the only problem that can occur removing o is that a vertex $x \in o$ is no longer separated from a vertex $y \in V \setminus o$. Vertices x and y must be separated by some other edge in $\mathcal{E} \setminus \mathcal{T}'$ as o is not black; furthermore this edge must contain y and not x, as any other edge containing x is in $(N_2[o] \setminus \{o\}) \subseteq \mathcal{T}'$. Let this edge be $\widetilde{e}$.

If x is the degree 1 vertex, then it is now the only isolated vertex and therefore it is separated from any other vertex. If x is a vertex of degree at least 2, then there is an edge (different from o) containing it; moreover, this edge cannot

12

contain any vertices of $\widetilde{e}$, because in this case $\widetilde{e} \in N_2[o]$. Hence, even deleting o, every vertex $x \in o$ is still separated from any other vertex, which ensures that $\mathcal{T}'$ is a test cover. $\qquad\square$

Lemma 8. *Let $(V, \mathcal{E}, \mathcal{B}, k)$ be an instance obtained after applying Rules 1, 2 and 3, and let x be a non-isolated vertex. Then $x \in V(N_3[g])$ for some green edge g.*

Proof. If x is contained in a green edge, we are done. If x is contained in an orange edge o, Lemma 7 implies that there exists a green edge g in $N_2[o]$, which means that $x \in V(N_2[g]) \subseteq V(N_3[g])$. If, finally, x is contained in a black edge b, this edge must intersect one other edge, that can be either green or orange (due to Rules 1 and 2). In both cases, $x \in V(N_3[g])$ for some green edge g. $\qquad\square$

Theorem 7. *There is a kernel for* SUBSET-TEST-r-COVER$(m-k)$ *with* $|V| \leq (k-1)k^5r^{16} + 1$ *and* $|\mathcal{E}| \leq (k-1)k^5r^{16} + k$. *This gives a kernel for* TEST-r-COVER$(m-k)$ *with* $|V| \leq 5(k-1)k^5r^{16} + 4k + 1$ *and* $|\mathcal{E}| \leq 3(k-1)k^5r^{16} + 3k$.

Proof. Let $(V, \mathcal{E}, \mathcal{B}, k)$ be an instance irreducible by Rules 1, 2 and 3. Construct greedily a set G of green edges which satisfy the hypothesis of Lemma 6. If $|G| \geq k$, we answer YES using Lemma 6. Otherwise, $|G| \leq k-1$ and every green edge which is not in G must be in $N_2[G]$. By Lemma 8, this means that every vertex (except the one of degree zero, if it exists) must be in $V(N_3[N_2[G]]) = V(N_5[G])$.

However, $|V(N_5[G])| \leq r|N_5[G]|$ and, given $\mathcal{F} \subseteq \mathcal{E}$, $|N_1[\mathcal{F}]| \leq |\mathcal{F}|(r)(kr^2)$ (by Lemma 5), which means that $|N_5[G]| \leq |G|(kr^3)^5$. To sum up, $|V(N_5[G])| \leq (k-1)k^5r^{16}$, which gives us the required bound on the number of vertices.

To bound the number of edges, we show that there is a solution of size at most $|V|$. First let $\mathcal{T}$ be the set of black edges. By Rule 2, the black edges are disjoint and therefore $|\mathcal{T}| \leq |V|$ and $\mathcal{T}$ induces at least $|\mathcal{T}|$ classes. Now if $\mathcal{T}$ is not a test cover, add an edge to $\mathcal{T}$ that increases the number of induced classes. Then eventually we have that $|\mathcal{T}| \leq |V|$ and $\mathcal{T}$ induces $|V|$ classes as required. Therefore if $|\mathcal{E}| - k \geq |V|$, the answer is YES. Hence $|\mathcal{E}| \leq |V| + k - 1 \leq (k-1)k^5r^{16} + k$, proving the kernel for SUBSET-TEST-r-COVER$(m-k)$.

We now prove the kernel for TEST-r-COVER$(m-k)$. First transform an instance $(V, \mathcal{E}, k)$ into an equivalent instance $(V, \mathcal{E}, \mathcal{B}, k)$ of SUBSET-TEST-r-COVER$(m-k)$, by letting $\mathcal{B} = \emptyset$. Then reduce this instance to a kernel $(V', \mathcal{E}', \mathcal{B}', k')$. Now we reduce $(V', \mathcal{E}', \mathcal{B}', k')$ to an instance $(V'', \mathcal{E}'', k'')$ of TEST-r-COVER$(m-k)$, completing the proof. If an edge b is colored black, and $\mathcal{E} \setminus \{b\}$ is not a test cover, then uncolor b. Otherwise, observe that by construction, b must have been created during the algorithm of Lemma 5, and in such a case b was contained within $k'' + 1$ other edges. Therefore b contains at most $r-1$ vertices. We will replace b with a small gadget that is 'equivalent' to b.

To make the gadget, add vertices x_1, x_2, x_3, x_4 to the instance, replace b with $b \cup \{x_1\}$ and add edges $e' = \{x_1, x_2, x_3\}$ and $e'' = \{x_3, x_4\}$. Observe edge e' is necessary to separate x_3 from x_4, e'' is necessary to separate x_2 from x_3 and

$b \cup \{x_1\}$ is necessary to separate x_1 from x_2. Hence all three edges must be in a test cover. Hence, in any test cover solution for the new instance, if we replace the gadget with b, we obtain a test cover solution for the original instance. Furthermore, such a solution will contain b. In this sense, the problems are equivalent.

Finally observe that for each original black edge we added four new vertices and two new edges. Hence $|V''| \leq |V'| + 4|\mathcal{E}'| \leq 5(k-1)k^5 r^{16} + 4k + 1$ and $|\mathcal{E}''| \leq |\mathcal{E}'| + 2|\mathcal{E}'| \leq 3(k-1)k^5 r^{16} + 3k$. $\qquad\square$

6 Polynomial kernel for TEST-r-COVER$(n-k)$

In this section, we show that TEST-r-COVER$(n-k)$ admits a kernel which is polynomial in the number of vertices. We may assume that $k \geq 2$ as the answer to TEST-r-COVER$(n-k)$ for $k \leq 1$ is always positive.

We reuse some terminology from [5]. We say $\mathcal{T} \subseteq \mathcal{E}$ is a k-mini test cover if $|\mathcal{T}| \leq 2k$ and the number of classes induced by $\mathcal{T}$ is at least $|\mathcal{T}| + k$.

The following result is a consequence of Lemma 1 and Theorem 2 of [5].

Lemma 9. *Assume $\mathcal{E}$ is a test cover of V. Then the following are equivalent:*

1. *$\mathcal{E}$ contains a test cover of size at most $n - k$.*

2. *There exists $\mathcal{T} \subseteq \mathcal{E}$ such that the number of classes induced by $\mathcal{T}$ is at least $|\mathcal{T}| + k$.*

3. *$\mathcal{E}$ contains a k-mini test cover.*

The following result follows from Lemma 1 and the proof of Theorem 2 in [5].

Lemma 10. *In polynomial time, we may either find a k-mini test cover, or find an $\mathcal{F} \subseteq \mathcal{E}$ such that:*

1. *$|\mathcal{F}| < 2k$*

2. *$\mathcal{F}$ induces less than $|\mathcal{F}| + k$ classes.*

3. *Each edge in $\mathcal{E}$ cuts at most one class induced by $\mathcal{F}$.*

4. *For any $e, e' \in \mathcal{E}$ and any class K induced by $\mathcal{F}$, at least one of $(e \cap e') \cap K$, $(e \backslash e') \cap K$, $(e \backslash e') \cap K$ and $K \backslash (e \cup e')$ is empty.*

Now assume we have found such an $\mathcal{F}$ and let $C_1, \ldots, C_l, G$ be the classes induced by $\mathcal{F}$ (where G is the class of vertices which are not contained in any edge of $\mathcal{F}$ and $l < 3k$). Let $\mathcal{C}$ be the set of classes $C_1, \ldots, C_l$, and let C be the set of vertices contained in such classes. Let $\mathcal{G}$ be the set of edges that intersect G. For each edge $e \in \mathcal{G}$, we say $e \cap G$ is the G-*portion* of e. A subset Γ of G is a *component* if Γ is the G-portion of an edge $e \in \mathcal{G}$ and $\Gamma \not\subset e' \cap G$ for all edges $e' \in \mathcal{G}$. Notice that the number of vertices in C is bounded by $(2k-1)r$ because these vertices are contained in edges of $\mathcal{F}$ and every edge contains at most r vertices. Also notice that every component of G has at most r vertices.

Theorem 8. *Given an instance $(V, \mathcal{E}, k)$, it is possible to reduce it in polynomial time to an equivalent instance with at most $18k^3r$ vertices and $(18k^3r)^r$ edges.*

Proof. We may assume that $|V| > (7k+2)r$ as otherwise our instance is already a kernel (every edge has at most r vertices and so the number of edges is at most $((7k+2)r)^r$). Observe that if there exists an edge e such that $|e \cap G| \geq |G|/2$ then $|G| \leq 2|e| \leq 2r$, and we conclude that $|V| \leq |C| + 2r \leq (2k+1)r$, a contradiction. Therefore, $|e \cap G| < |G|/2$, and so by Part 4 of Lemma 10 if X, Y are different G-portions then either $X \subset Y$, $Y \subset X$ or $X \cap Y = \emptyset$.

Apply the following algorithm:

Step 1: For each pair (C_i, C_j) $(i \neq j)$ in turn, mark $2k$ unmarked components of G which contain the G-portion of an edge containing C_i and having empty intersection with C_j (mark these edges too). If there are less than $2k$ such components, mark them all. Let $E_{i,j}$ denote the set of marked edges.

For each C_i in turn, mark $2k+1$ unmarked components of G which contain the G-portion of an edge containing C_i (mark these edges too). If there are less than $2k+1$ such components, mark them all. Let E_i denote the set of marked edges.

Step 2: Delete every edge in $\mathcal{G}$ whose G-portion is not contained in a marked component of G. Delete every vertex which is not contained in any edge anymore, except one vertex y (if it exists).

Let $\mathcal{E}'$ be the set of edges which have not been deleted by this algorithm. Notice that the number of marked components in G is at most $(3k-1)(3k-2)(2k) + (3k-1)(2k+1) < (3k-1)^2(2k+1) = 18k^3 - 3k^2 - 4k + 1 < 18k^3 - 2k$ (here we use the assumption that $k \geq 1$). Let G' be the set of vertices of G which have not been deleted by the algorithm. Notice that $|G'| \leq (18k^3 - 2k)r$.

In the instance which is produced, $|V'| = |C| + |G'| \leq 18k^3r$, and since each edge contains at most r vertices, $|\mathcal{E}'| \leq (18k^3r)^r$. Hence it is sufficient to show that $(V', \mathcal{E}', k)$ admits a k-mini test cover if and only if $(V, \mathcal{E}, k)$ admits one.

Obviously, if $(V', \mathcal{E}', k)$ admits a k-mini test cover, this is a k-mini test cover for $(V, \mathcal{E}, k)$ too. For the other direction, suppose $\mathcal{T}$ is a k-mini test cover for $(V, \mathcal{E}, k)$ such that $\mathcal{T} \setminus \mathcal{E}'$ is as small as possible. For the sake of contradiction, suppose that $\mathcal{T}$ contains at least one edge e in $\mathcal{T} \setminus \mathcal{E}'$. We claim that it is possible to construct a set $\mathcal{T}'''$ which induces at least $|\mathcal{T}'''| + k$ classes, such that $\mathcal{T}''' \setminus \mathcal{E}' = (\mathcal{T} \setminus \mathcal{E}') \setminus \{e\}$. By applying Lemma 9 to the hypergraph H' with edge set $\mathcal{T}'''$, and vertex set formed by identifying the vertices in each class induced by $\mathcal{T}'''$, observe that there exists a k-mini test cover in H'. Observe that the edges of this k-mini test cover in H' also form a k-mini test cover in the original instance, and this k-mini test cover is a subcollection of $\mathcal{T}'''$. Since this k-mini test cover contains fewer edges from $\mathcal{E} \setminus \mathcal{E}'$ than $\mathcal{T}$ does, we have a contradiction.

Start with $\mathcal{T}' = \mathcal{T} \setminus \{e\}$. Since e is not in $\mathcal{E}'$, e must be in $\mathcal{G}$, and the G-portion of e must not be contained in any marked component. Furthermore, for each $C_i, C_j \in \mathcal{C}$ with $C_i \subseteq e$ and $e \cap C_j = \emptyset$ we note that $E_{i,j}$ must contain $2k$ edges, as otherwise e would be in $\mathcal{E}'$. Similarly, for each C_i contained in e

we note that E_i must contain $2k + 1$ edges. For any i, j such that $|E_{i,j}| = 2k$, let $e_{i,j}$ be an edge in $E_{i,j}$ whose G-portion is disjoint from any edge in $\mathcal{T}'$. This must exist as $|\mathcal{T}'| \leq 2k - 1$.

For any i such that $|E_i| = 2k + 1$ let e_i, e_i' be edges in E_i whose G-postions are disjoint from any edge in $\mathcal{T}'$. These edges must exist as $|\mathcal{T}'| \leq 2k - 1$.

Let C_0^* be the class induced by $\mathcal{T}'$ that consists of all vertices not in any edge in $\mathcal{T}'$ (which exists by Claim C below). We will need the following claims.

Claim A: There is at most one class C_G^* induced by $\mathcal{T}'$, such that $G \cap (C_G^* \cap e) \neq \emptyset$ and $G \cap (C_G^* \setminus e) \neq \emptyset$.

Proof of Claim A: For the sake of contradiction assume that there are two such classes C_G' and C_G''. This implies that there exist vertices $x' \in G \cap (C_G' \cap e)$, $y' \in G \cap (C_G' \setminus e)$, $x'' \in G \cap (C_G'' \cap e)$ and $y'' \in G \cap (C_G'' \setminus e)$. Some edge $e' \in \mathcal{T}'$ separates C_G' and C_G''. Note that adding e' and e to $\mathcal{F}$ separates x', y', x'' and y'' into different classes, contradicting Part 4 of Lemma 10. This contradiction complete the proof of Claim A.

Claim B: For each edge e' that cuts G and every C_i we have $C_i \subseteq e'$ or $C_i \cap e' = \emptyset$. In particular, $C_i \subseteq e$ or $C_i \cap e = \emptyset$.

Proof of Claim B: If Claim B is false then there exist $x \in C_i \cap e'$ and $y \in C_i \setminus e'$. So adding e' to $\mathcal{F}$ cuts G (as G is not a subset of any edge and e' contains vertices from G) and C_i, a contradiction to Part 3 in Lemma 10.

Claim C: C_0^* exists and $|G \cap C_0^*| \geq (3k + 2)r$.

Proof of Claim C: If C_0^* does not exist then every vertex of V belongs to some edge in $\mathcal{T}'$, which implies that $|V| \leq 2kr$, so C_0^* does exist. If $|G \cap C_0^*| < (3k + 2)r$, then the following holds and we have a contradiction to the assumption on $|V|$ in the beginning of the proof:

$$|V| \leq 2kr + |C_0^*| \leq 2kr + (|C| + |G \cap C_0^*|) < 2kr + 2kr + (3k + 2)r = (7k + 2)r.$$

By Claim A there exists at most one class, say C_G^*, induced by $\mathcal{T}'$ that is cut by e and only contains vertices from G. Let $C_1^*, \ldots, C_t^*$ be all classes induced by $\mathcal{T}'$, different from C_G^* and C_0^*, that are cut by e. Note that $t \leq 3k$, as $\mathcal{T}$ is a k-mini test cover. Each C_s^*, $(1 \leq s \leq t)$, must be contained in an edge, say e_s^*, in $\mathcal{T}'$ and contain vertices from C, by the definitions on C_G^* and C_0^*. We are going to create a collection of edges $\mathcal{T}''$ such that each C_s^* is cut by an edge in $\mathcal{T}''$ and also $\mathcal{T}''$ induces $|\mathcal{T}''|$ extra classes in C_0^*. Initially let $\mathcal{T}'' = \emptyset$. For each $s \in [t]$ in turn, consider the following two cases.

Case 1: For some $i \neq j$, e contains C_i but not C_j and $C_i \cap C_s^* \neq \emptyset$, $C_j \cap C_s^* \neq \emptyset$.

In this case observe that $|E_{i,j}| = 2k$, as otherwise e would be marked. Then add the edge $e_{i,j}$ to $\mathcal{T}''$, if $e_{i,j}$ is not in $\mathcal{T}''$ already. Note that $e_{i,j}$ separates C_i from C_j and therefore cuts C_s^*, and also creates an extra class in C_0^*, as desired.

16

Case 2: Case 1 does not hold. That is, $C \cap C_s^* \subseteq e$ or $(C \cap C_s^*) \cap e = \emptyset$.

Recall that there exists a C_i such that C_s^* contains vertices from C_i and, since e cuts C_s^*, we have $C_s^* \cap G \neq \emptyset$. Suppose e does not contain C_i. Then $(C \cap C_s^*) \cap e = \emptyset$ and, since e cuts C_s^*, it must contain vertices from $C_s^* \cap G \subseteq e_s^* \cap G$. Then e_s^* cuts G and the G-portion of e_s^* is in the same component as the G-portion of e, and therefore e_s^* is an unmarked edge. Furthermore since e_s^* cuts G it does not cut C_i, and therefore $C_i \subseteq e_s^*$. Thus, we have that either e or e_s^* is an unmarked edge containing C_i, and therefore $|E_i| = 2k + 1$. Then add e_i to $\mathcal{T}''$, if e_i is not already in $\mathcal{T}''$. Observe that e_i cuts C_s^* as it contains vertices in $C_i \cap C_s^*$ but no vertex from $C_s^* \cap G$, and e_i creates an extra class in C_0^*, as required.

This completes Case 1 and Case 2. Note that the edges in $\mathcal{T}''$ all have vertex disjoint G-portions, as they are in distinct $E_{i,j}$'s and E_i's. We now consider Case (i) and Case (ii) below, which will complete the proof.

Case (i): C_G^* does not exist or is equal to C_0^* or e does not cut C_0^* or does not cut C_G^*.

In this case e cuts at most $t + 1$ classes induced by $\mathcal{T}'$. Note that if we add the edges from $\mathcal{T}''$ to $\mathcal{T}'$, each edge in $\mathcal{T}'$ increase the number of classes in C_0^* by at least one. Also note that for every $s \in [t]$ some edge in $\mathcal{T}''$ cuts C_s^*. So let $\mathcal{T}''' = \mathcal{T}' \cup \mathcal{T}'' = (\mathcal{T} \setminus e) \cup \mathcal{T}''$. Removing e from $\mathcal{T}$ decreases the number of classes by at most $t + 1$ and adding $\mathcal{T}''$ increases the number of classes by at least $t + |\mathcal{T}''|$. So by increasing the number of edges by $|\mathcal{T}''| - 1$ we have increased the number of classes by at least $|\mathcal{T}''| - 1$ and therefore we still have at least k more classes than edges.

Case (ii): Case (i) does not hold. That is, C_G^* exists and is distinct from C_0^* and e cuts both C_G^* and C_0^*.

By Claim A we note that e either contains all of $C_0^* \cap G$ or none of $C_0^* \cap G$. By Claim C e must contain none of $C_0^* \cap G$. As e cuts C_0^* we must have $C \cap e \cap C_0^* \neq \emptyset$. Therefore there exists C_i such that e contains vertices from $C_i \cap C_0^*$, and so $|E_i| = 2k + 1$. Add e_i and e_i' to $\mathcal{T}''$ (unless e_i is already in $\mathcal{T}'$, in which case just add e_i' to $\mathcal{T}''$). Observe that the G-portions of e_i and e_i' are vertex disjoint by construction, and so the G-portions of all edges in $\mathcal{T}''$ are still vertex disjoint.

Note that adding e_i and e_i' to $\mathcal{T}'$ creates three new classes in C_0^* (C_0^* now being split into the class $C_0^* \cap e \cap e'$ which contains vertices from C_i, the G-portion of e_i, the G-portion of e_i' and the class of vertices not in any edge). Adding each other edge from $\mathcal{T}''$ to $\mathcal{T}'$ increases the number of classes in C_0^* by one (as by Claim C we note that some vertex in $G \cap C_0^*$ is not contained in any edge in $\mathcal{T}''$). Also note that for every $s \in [t]$ some edge in $\mathcal{T}''$ cuts C_s^*.

So let $\mathcal{T}''' = \mathcal{T}' \cup \mathcal{T}'' = (\mathcal{T} \setminus e) \cup \mathcal{T}''$. Removing e from $\mathcal{T}$ decreases the number of classes by $t + 2$ and adding $\mathcal{T}''$ increases the number of classes by at least $t + |\mathcal{T}''| + 1$. So by increasing the number of edges by $|\mathcal{T}''| - 1$ we have increased the number of classes by at least $|\mathcal{T}''| - 1$ and therefore we still have at least k more classes than edges. $\square$

7 Discussion

The two main results proved in this paper are the existence of polynomial-size kernels for TEST-r-COVER($m-k$) and TEST-r-COVER($n-k$). In fact, our result for TESTCOVER($m-k$) is stronger: TESTCOVER($m-k$) has a polynomial-size kernel for the parameter $k+r$. It would be interesting to find out whether TESTCOVER($n-k$) has a polynomial-size kernel for the parameter $k+r$. In addition, Theorem 6 gives an algorithm for TESTCOVER($m-k$) with running time $(f(r))^{O(k)}(n+m)^{O(1)}$; it would be interesting to see if an algorithm with similar running time can be found for TESTCOVER($n-k$).

8 Acknowledgments

We are grateful to Manu Basavaraju and Mathew Francis for carefully reading an earlier version of this paper and informing us of a subtle flaw in Theorem 8 which led us to changing the proof substantially.

References

[1] N. Alon, G. Gutin, E. J. Kim, S. Szeider, and A. Yeo, Solving MAX-r-SAT above a tight lower bound. Algorithmica 61 (2011), 638–655.

[2] H.L. Bodlaender, R.G. Downey, M.R. Fellows, and D. Hermelin, On problems without polynomial kernels, *J. Comput. Syst. Sci.*, 75(8): 423–434, 2009.

[3] J.A. Bondy, Induced subsets. *J. Combin. Th., Ser. B* 12 (1972), 201–202.

[4] R. Crowston, M. Fellows, G. Gutin, M. Jones, F. Rosamond, S. Thomassé, and A. Yeo, Simultaneously Satisfying Linear Equations Over $\mathbb{F}_2$: MaxLin2 and Max-r-Lin2 Parameterized Above Average. *Proc. FSTTCS 2011*, LIPICS Vol. 13, 229–240.

[5] R. Crowston, G. Gutin, M. Jones, S. Saurabh, and A. Yeo, Parameterized Study of the Test Cover Problem. In *MFCS 2012*, Lect. Notes Comput. Sci. 7464 (2012), 283-295.

[6] R. Crowston, G. Gutin, M. Jones, and A. Yeo, A new lower bound on the maximum number of satisfied clauses in Max-SAT and its algorithmic applications. Algorithmica 64 (2012), 56–68.

[7] R. Crowston, M. Jones, and M. Mnich, Max-Cut Parameterized above the Edwards-Erdős Bound, In *ICALP 2012*, Lect. Notes Comput. Sci. 7391 (2012) 242–253.

[8] K.M.J. De Bontridder, B.J. Lageweg, J.K. Lenstra, J.B. Orlin, and L. Stougie, Branch and bound algorithms for the test-cover problem. In *ESA 2002*, Lect. Notes Comput. Sci. 2461 (2002), 223–233.

[9] K.M.J. De Bontridder, B.V. Halldórsson, M.M. Halldórsson, C.A.J. Hurkens, J.K. Lenstra, R. Ravi, and L. Stougie, Approximation algorithms for the test cover problem, Math. Programming-B 98 (13) (2003), 477–491.

[10] M. Dom, D. Lokshtanov, and S. Saurabh, Incompressibility though Colors and IDs, In *36th ICALP, Part I*, Lect. Notes Comput. Sci. 5555: 378–389, 2009.

[11] R. G. Downey and M.R. Fellows, Parameterized Complexity, Springer, 1999.

[12] T. Fahle and K. Tiemann, A faster branch-and-bound algorithm for the test-cover problem based on set-covering techniques, J. Exp. Algorithmics 11 (2007), 2.2.

[13] J. Flum and M. Grohe, Parameterized Complexity Theory, Springer Verlag, 2006.

[14] M.R. Garey and D.S. Johnson, Computers and Intractability. A guide to the theory of NP-completeness. CA, Freeman, 1979.

[15] G. Gutin, E.J. Kim, M. Mnich, and A. Yeo, Betweenness Parameterized Above Tight Lower Bound. J. Comput. Syst. Sci. 76 (2010), 872–878.

[16] G. Gutin, G. Muciaccia, and A. Yeo, (Non-)existence of Polynomial Kernels for the Test Cover Problem. Inform. Proc. Lett. 113 (2013), 123–126

[17] B.V. Halldórsson, M.M. Halldórsson, and R. Ravi, On the approximability of the Minimum Test Collection problem. In *ESA 2001*, Lect. Notes Comput. Sci. 2161 (2001), 158–169.

[18] B.V. Halldórsson, J.S. Minden, and R. Ravi, PIER: Protein identification by epitope recognition. In *Currents in Computational Molecular Biology 2001*, 109–110, 2001.

[19] R.M. Karp. Reducibility among combinatorial problems, In R.E. Miller and J.W. Thatcher, editors, Complexity of Computer Computations, pp. 85103. Plenum Press, 1972.

[20] M. Mahajan and V. Raman, Parameterizing above guaranteed values: MaxSat and MaxCut. J. Algorithms 31(2) (1999), 335–354.

[21] M. Mahajan, V. Raman, and S. Sikdar, Parameterizing above or below guaranteed values. J. Comput. Sys. Sci. 75(2):137–153, 2009.

[22] B.M.E. Moret and H.D. Shapiro, On minimizing a set of tests, SIAM J. Scientific & Statistical Comput. 6 (1985), 983–1003.

[23] R. Niedermeier, Invitation to Fixed-Parameter Algorithms, Oxford University Press, 2006.