New and Improved Bounds for the Minimum Set Cover Problem

Rishi Saket[1] and Maxim Sviridenko[2]

[1] IBM T. J. Watson Research Center, NY, USA.
[2] Department of Computer Science, University of Warwick, UK.

Abstract. We study the relationship between the approximation factor for the Set-Cover problem and the parameters Δ : the maximum cardinality of any subset, and k : the maximum number of subsets containing any element of the ground set. We show an LP rounding based approximation of $(k-1)(1-e^{-\frac{\ln \Delta}{k-1}})+1$, which is substantially better than the classical algorithms in the range $k \approx \ln \Delta$, and also improves on related previous works [19, 22]. For the interesting case when $k = \theta(\log \Delta)$ we also exhibit an integrality gap which essentially matches our approximation algorithm. We also prove a hardness of approximation factor of $\Omega\left(\frac{\log \Delta}{(\log \log \Delta)^2}\right)$ when $k = \theta(\log \Delta)$. This is the first study of the hardness factor specifically for this range of k and Δ, and improves on the only other such result implicitly proved in [18].

1 Introduction

We consider the classical minimum set cover problem. We are given the ground set $\{1, \ldots, n\} = [n]$ and m subsets $S_j \subseteq [n]$ for $j = 1, \ldots, m$. Each set S_j has an associated non-negative weight w_j. The goal is to choose a collection of sets indexed by $\mathcal{C} \subseteq \{1, \ldots, m\} = [m]$ such that $[n] = \cup_{j \in \mathcal{C}} S_j$ and minimize $\sum_{j \in \mathcal{C}} w_j$.

There are two additional parameters associated with the problem. Let $\Delta = \max_{j \in [m]} |S_j|$ be the maximal cardinality of a set in the instance. For each element $i \in [n]$, let $k_i = |\{S_j : i \in S_j, j \in [m]\}|$ be the number of sets in the instance containing the element $i \in [n]$ and let $k = \max_{i \in [n]} k_i$.

There are two types of classical approximation algorithms for the minimum set cover problem. The natural greedy algorithm has performance guarantee $\ln \Delta + 1$ [20, 12, 5]. Another well-known type of algorithms has performance guarantee k [4, 10]. Both performance guarantees are asymptotically the best possible under natural complexity assumptions [7, 6, 17] specifically for the regime where Δ is not bounded by a constant, although for constant Δ Halperin's [9] algorithm has a performance guarantee strictly better than k. Nevertheless, assuming that Δ is not bounded, if one defines the performance ratio $\rho(k)$ as a function of parameter k the classical approximation algorithms provide us with performance guarantee $\rho(k) = \min\{k, \ln \Delta + 1\}$ (see Figure 1). The function $\rho(k)$ is not smooth at the point $k = \ln \Delta + 1$, which indicates that performance guarantee of classical algorithms is not best possible, at least in regime when $k \approx \ln \Delta + 1$.

Our Results. In this paper we study the relationship between the approximation factor for Set-Cover in terms of k and Δ. We prove the following results.

Approximation Algorithm. In this paper we design a simple LP rounding based approximation algorithm with performance guarantee $(k-1)(1-e^{-\frac{\ln \Delta}{k-1}})+1$ which asymptotically matches the performance guarantee of known (and best possible) approximation algorithms when $k \ll \ln \Delta$ or $k \gg \ln \Delta$ in the regime where Δ is unbounded. In particular, when $k = \ln \Delta + 1$, our algorithm has performance guarantee $(1 - e^{-1}) \ln \Delta + 1$. For a comparison of the performance of our algorithm with $\rho(k)$, refer to Figure 2. Our approximation algorithm and its analysis are presented in Section 2.

Previous results in this direction are due to Krivelevich [19] and Okun [22]. Using our notations Krivelevich [19] designed an approximation algorithm with performance guarantee $\max\{k-1, (k-1)(1-e^{-\frac{\ln \Delta}{k-1}})+1\}$ for the case when all subsets have cardinality Δ and all elements of the ground set belong to exactly k sets. Okun [22] designed an approximation algorithm that works in the regime when $(1-e^{-1})k \leq \ln \Delta$ with performance guarantee smaller than k but strictly worse than ours.

Integrality Gap. For the interesting regime where $k = \theta(\log \Delta)$ we show an LP integrality gap of $k(1 - e^{-\frac{\ln \Delta}{k}} - \delta)$ for any constant $\delta > 0$, essentially matching our LP rounding upper bound. Our construction is probabilistic and is given in Section 3.

Hardness of Approximation. In this work we obtain a lower bound of $\Omega\left(\frac{\log \Delta}{(\log \log \Delta)^2}\right)$ when $k = \theta(\log \Delta)$, where Δ is a polynomial in n. In previous work, Feige [7] had shown that in the regime where $k = \Omega(\Delta^\gamma)$ for some constant $\gamma > 0$, it is hard to approximate Set-Cover to within a factor of $(1-\varepsilon)\ln \Delta$. As mentioned before, this essentially matches the $\ln \Delta + 1$ greedy algorithm. A slightly weaker lower bound of $\Omega(\log \Delta)$ was obtained by Lund and Yannakakis [21] for $k = \Omega((\log \Delta)^c)$, where $c > 1$ is a large constant, and for $k = 2^{\log^{1-\varepsilon} \Delta}$ by Raz and Safra [23] and by Alon, Moshkovitz and Safra [1]. On the other hand, for small values of k the known hardness factors are linear in k. For constant k, assuming the Unique Games Conjecture [14] it is NP-hard to approximate within a factor of $k - \varepsilon$ [17, 3]. In [6] it was shown that for superconstant $k = O((\log \log \Delta)^{1/c})$ the hardness factor is $k - 1 - \varepsilon$, and for $k = O((\log \Delta)^{1/c})$ it is $k/2 - \varepsilon$. In all these hardness reductions (except for that of [17, 3]) Δ is a polynomial in the size of the ground set n. It should be noted that these hardness results did not explicitly state the dependence between k, Δ and n, and these relations can be inferred from the respective hardness reductions.

However, the interesting case when $k = \theta(\log \Delta)$ remained unexplored till the work of Khot and Saket [18] who studied the problem of minimizing the size of DNF expression of a boolean function given its truth table. In their work [18], they implicitly obtain a hardness factor (for $k = \theta(\log \Delta)$, Δ polynomial in n) of $\Omega(\log^{1-\varepsilon} \Delta)$ although [18] do not explicitly mention this in their work.

Our stronger lower bound of $\Omega\left(\frac{\log \Delta}{(\log\log \Delta)^2}\right)$ is obtained by revisiting the Probabilistically Checkable Proof (PCP) construction of [18] using different parameters while avoiding some of the complications of their reduction, and is presented in Section 4. This still leaves open the possibility that when $k = \ln \Delta + 1$, our approximation of $(1 - e^{-1}) \ln \Delta + 1$ may not be optimal. Conversely, it may be possible to improve the hardness factor to match the algorithmic bound. We include, in Section 4.4, a brief discussion on some of the limitations of current PCP techniques to improving the hardness factor. The hardness result of this paper along with previous ones for various regimes of k are summarized in Figure 3.

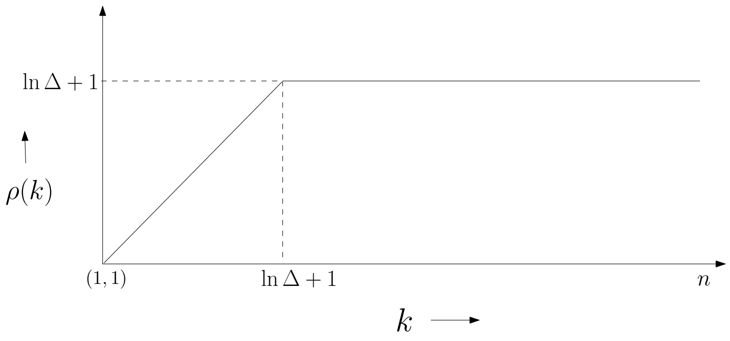

Fig. 1. Approximation Factor by Classical Algorithms.

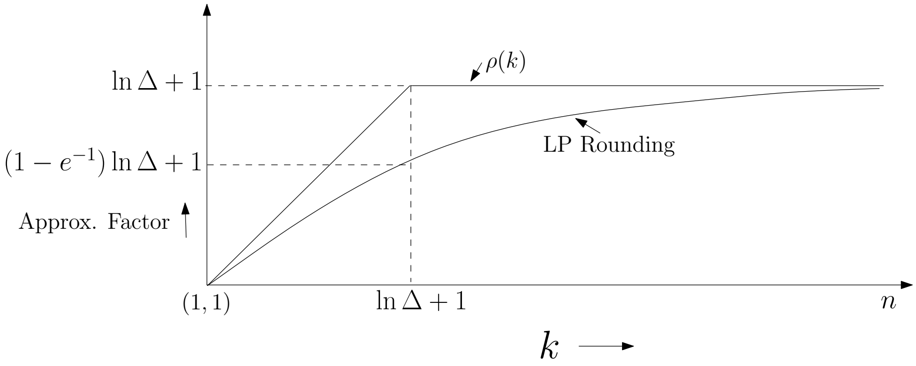

Fig. 2. Comparison of $\rho(k)$ with the LP Rounding Approximation for growing parameter Δ.

Range of k	Hardness Factor	Complexity Assumption	Reference
k: arbitrarily large const.	$k-\varepsilon$	Unique Games Conj. [14]	[17, 3]
$k \leq O((\log\log \Delta)^{1/c})$	$k-1-\varepsilon$	NP$\not\subseteq$ DTIME($n^{O(\log\log n)}$)	[6]
$k \leq O((\log \Delta)^{1/c})$	$k/2-\varepsilon$	NP$\not\subseteq$ DTIME($n^{O(\log\log n)}$)	[6]
$k=\theta(\log \Delta)$	$\Omega(\log^{1-\varepsilon}\Delta)$	NP$\not\subseteq$ DTIME($n^{\mathrm{poly}(\log n)}$)	Implicit in [18]
$k=\theta(\log \Delta)$	$\Omega\left(\frac{\log \Delta}{(\log\log \Delta)^2}\right)$	NP$\not\subseteq$ DTIME($n^{\mathrm{poly}(\log n)}$)	This work.
$k=\Omega((\log \Delta)^c)$	$\Omega(\log \Delta)$	NP$\not\subseteq$ DTIME($n^{O(\log\log n)}$)	[21, 15]
$k=\Omega(2^{\log^{1-\varepsilon}\Delta})$	$\Omega(\log \Delta)$	P $\neq$ NP	[23, 1]
$k=\Omega(\Delta^\gamma)$	$(1-\varepsilon)\ln \Delta$	NP$\not\subseteq$ DTIME($n^{O(\log\log n)}$)	[7]

Fig. 3. Summary of known NP-hardness factors for Set-Cover with different ranges of k.

2 Approximation Algorithm

Consider the following linear programming relaxation of the minimum set cover problem:

$$\min \sum_{j\in[m]} w_j x_j, \tag{1}$$

$$\sum_{j:i\in S_j} x_j \geq 1,\ \forall i \in [n], \tag{2}$$

$$x_j \geq 0,\ \forall j \in [m]. \tag{3}$$

Our approximation algorithm solves linear programming relaxation on the first step. Let LP^* be the optimal value of the linear programming relaxation and $x_j^*, j \in [m]$ be the optimal fractional solution found by the LP solver. We define $p_j = \min\{1, \alpha k \cdot x_j^*\}$ where $\alpha = 1 - e^{-\frac{\ln \Delta}{k-1}}$. Our approximation algorithm defines a partial cover by choosing to add the set S_j to the cover with probability p_j and not choosing it with probability $1-p_j$ independently at random. Let R_1 be the indices of sets chosen by our random procedure. Let I^r be the set of the elements of the ground set that do not belong to any of the sets chosen by the random procedure, i.e. the set of uncovered elements. Each element in I^r chooses the cheapest set in our instance that covers it. Let R_2 be the set of indices of such sets covering I^r. Our algorithm outputs $R_1 \cup R_2$ as the final solution.

Theorem 1. *The expected value of the approximate solution output by our algorithm is at most* $((k-1)(1-e^{-\frac{\ln \Delta}{k-1}})+1)LP^*$.

Proof. By linearity of expectation, the expected value of the sets indexed by R_1 is $\sum_{j\in[m]} w_j p_j \leq k(1-e^{-\frac{\ln \Delta}{k-1}})LP^*$.

Assume that each element $i \in [n]$ of the ground set chooses the cheapest set that covers that element. Let j_i be the index of such a set and $W = \sum_{i\in[n]} w_{j_i}$ be the upper bound on the weight of chosen sets. Then by utilizing the constraints (2) we obtain

$$W = \sum_{i\in[n]} w_{j_i} \le \sum_{i\in[n]} w_{j_i} \sum_{j:i\in S_j} x_j^* \le \sum_{i\in[n]} \sum_{j:i\in S_j} w_j x_j^* \le \Delta \cdot LP^*.$$

Now, we estimate $Pr[i \in I^r]$ above. If $p_j = 1$ for at least one set such that $i \in S_j$ then $Pr[i \in I^r] = 0$. Otherwise, $p_j = \alpha k \cdot x_j^*$ for all sets S_j such that $i \in S_j$ and

$$Pr[i \in I^r] = \prod_{j|i\in S_j} (1 - p_j) \le \left(1 - \frac{\sum_{j|i\in S_j} p_j}{k_i}\right)^{k_i} \le \left(1 - \frac{\sum_{j|i\in S_j} p_j}{k}\right)^{k}$$
$$= \left(1 - \frac{\sum_{j|i\in S_j} \alpha k \cdot x_j^*}{k}\right)^{k} \le (1-\alpha)^k = \frac{1}{\Delta^{k/(k-1)}}.$$

Therefore, by linearity of expectation, the expected weight of the sets in R_2 can be estimated above by $W/\Delta^{k/(k-1)} \le LP^*/\Delta^{1/(k-1)}$. Overall, the expected cost of the approximate solution is upper bounded above by

$$\left(k(1 - e^{-\frac{\ln \Delta}{k-1}}) + \frac{1}{\Delta^{1/(k-1)}}\right) LP^* = \left((k-1)(1 - e^{-\frac{\ln \Delta}{k-1}}) + 1\right) LP^*$$

3 Integrality Gap

The integrality gap of a linear programming relaxation for the specific instance of a minimization problem is the ratio between the minimum value integral solution of the relaxation (in the numerator) and the minimum value of the fractional solution (in the denominator).

Consider the following instance of the minimum set cover problem. We are given a ground set of n elements and $m = n^\varepsilon$ sets. We fix an arbitrary constant $c > 0$ and consider the regime when $k = c \cdot \ln n$. Each element $i \in [n]$ independently at random chooses k sets out of possible m sets, i.e. this element chooses one combination of k sets out of possible $\binom{m}{k}$ variants uniformly at random. Each set S_j for $j \in [m]$ consists of elements that chose that set. Let $\mathcal{I}_\varepsilon$ be the resulting random instance of the minimum set cover problem. Note that the parameter $\Delta \le n$. We prove the following theorem showing that for all values of $c > 0$ the instance $\mathcal{I}_\varepsilon$ is, with high probability, the desired integrality gap example.

Theorem 2. *For any constants $c > 0$ and $\delta > 0$, there exists a constant $\varepsilon > 0$ such that the integrality gap of the linear programming relaxation (1)-(3) for the instance $\mathcal{I}_\varepsilon$ is at least $k(1 - e^{-1/c} - \delta)$ with high probability for large enough n.*

Proof. First, we note the fractional solution $x'_j = 1/k$ for all $j \in [m]$ is feasible. Indeed, each element is covered by exactly k sets in the instance $\mathcal{I}_\varepsilon$. Therefore, $\sum_{j:i\in S_j} x'_j = 1$ for each element $i \in [n]$. We obtain $LP^* \le m/k$.

We will assume that the constants $c, \delta > 0$ and m are such that the number $(e^{-1/c} + \delta)m$ is an integer. We now fix an arbitrary collection of sets indexed by $\mathcal{C} \subseteq [m]$ such that $|\mathcal{C}| = (1 - e^{-1/c} - \delta)m$. We will estimate the probability that this integral solution is infeasible, i.e. there exist an element $i \in [n]$ which is left uncovered by the sets in this collection in the instance $\mathcal{I}_\varepsilon$.

The probability that a fixed element $i \in [n]$ is not covered by the sets indexed by $\mathcal{C}$ is exactly

$$\frac{\binom{(e^{-1/c}+\delta)m}{k}}{\binom{m}{k}} = \prod_{i=0}^{k-1} \frac{(e^{-1/c}+\delta)m - i}{m-i} \geq (e^{-1/c} + \delta/2)^k = \frac{(1 + e^{1/c}\delta/2)^{c \ln n}}{n}$$
$$= n^{-(1-F_{c,\delta})},$$

where the inequality holds for large enough m since $k << m$ and $F_{c,\delta} = c\ln(1 + \delta e^{1/c}/2)$ is a constant depending on c and δ. We assume that δ is small enough that $F_{c,\delta} \in (0, 1)$.

Since each element chooses its sets independently, the probability that all n elements are covered by the sets indexed by $\mathcal{C}$ is at most

$$\left(1 - n^{-(1-F_{c,\delta})}\right)^n \leq e^{-n^{F_{c,\delta}}}.$$

The total number of choices for the index set $\mathcal{C}$ is at most $2^m = 2^{n^\varepsilon}$. Therefore, by the union bound, the probability that there exists a feasible index set $\mathcal{C}$ is at most

$$e^{-n^{F_{c,\delta}}} 2^{n^\varepsilon} \leq e^{n^\varepsilon - n^{F_{c,\delta}}}.$$

If we choose $\varepsilon = F_{c,\delta}/2$ then probability that there exists a feasible solution becomes negligibly small for large values of n. Therefore, with probability at least $1 - e^{n^\varepsilon - n^{F_{c,\delta}}}$ one needs to choose at least $(1 - e^{-1/c} - \delta)m$ sets into any feasible integral solution. This implies the claimed bound on the integrality gap.

4 Hardness of Approximation

In this section we shall derive an inapproximability result for the minimum set cover problem when $k = \theta(\log \Delta)$. Our reduction utilizes a PCP verifier constructed by Khot and Saket [18] who used it to prove a nearly optimal hardness result for minimizing the size of DNF expressions for a boolean function given its truth table, which is itself a special case of minimum set cover. We slightly modify the parameters of the verifier constructed in [18] to construct an instance of maximum constraint satisfaction problem (CSP) with some specific properties. This is then combined with a reduction – similar to that of Holmerin [11] for vertex cover – to obtain an instance of Set-Cover. In Section 4.1 we define the constraint satisfaction problem and state a hardness result for it, a proof of which is given in Section 4.3. The hardness reduction to Set-Cover is given in Section 4.2.

In the rest of this section, for convenience, we shall use notations (such as k, n) in contexts different to the previous sections.

4.1 A Hardness Result for Constraint Satisfaction

In this section we shall describe a result on the hardness of a variant of maximum constraints satisfaction problem (as defined below), which shall be useful in our reduction for the Set-Cover problem.

Definition 1. *An instance of Max-CSP-Reg(t, k) with N constraints, with parameters t, k as functions of N consists of a set of variables V, a label set $[k]$ and a set of t-variable constraints E, with $|E| = N$. The constraints are non-trivial, i.e. there is at least one satisfying labeling for every constraint. Additionally, each variable occurs in the same number of constraints. The goal is to assign labels to each variable in V to satisfy as many constraints in E as possible.*

The following hardness result for Max-CSP-Reg follows from the results in [16] and [18], and a formal proof is presented in Section 4.3.

Theorem 3. *Given an instance $\mathcal{A}$ of Max-CSP-Reg(t, k) with variable set V and set of constraints E, where $|E| = N$, $tk = \omega(\log N)$ and $t = \theta((\log k)^2)$, there is no polynomial time algorithm to distinguish between the following two cases:*

YES CASE: There is a set $V' \subset V$ of variables of size at most $|V|/(k^3)$ and a labeling $\sigma^* : V \setminus V' \mapsto [k]$ such that,

1. *(Strong Completeness) σ^* satisfies all constraints in E induced by $V \setminus V'$.*
2. *(Extendability) For any constraint $e \in E$ (possibly containing variables from V'), there is a labeling σ'_e to variables in $e \cap V'$ such that σ^* extended by σ'_e satisfies constraint e.*

NO CASE: Any labeling σ to the variables of V satisfies at most $k^{-t+O(\sqrt{t})}$ (soundness) fraction of the constraints,

unless NP $\subseteq$ DTIME$(n^{\mathrm{poly}(\log n)})$.

In the next subsection we shall give a reduction from Max-CSP-Reg to an instance of Set Cover to prove our hardness result.

4.2 Reduction to Set-Cover

Now we give a reduction from the instance $\mathcal{A}$ of Max-CSP-Reg(t, k) given in Theorem 3 to an instance $\mathcal{I}$ of Set-Cover. As before we have V as the variable set of $\mathcal{A}$ and E as the set of t-variable constraints, where $E = |N|$, $kt = \omega(\log N)$ and $t = \theta((\log k)^2)$. Before describing the instance $\mathcal{I}$ of Set-Cover, we need to construct the following objects.

For every variable v, define a set $L(v) := \{(v, i) \mid i \in [k]\}$, which is just the set of all labels for that variable. Let $e \in E$ be any constraint over variables $v_1, \ldots, v_t$. Define $\tilde{L}(e) := \cup_{i=1}^{t} L(v_i)$. Clearly, $|\tilde{L}(e)| = tk$ for all $e \in E$.

Let $T(e)$ be set of all labelings τ to $v_1, \ldots, v_t$ that satisfy e. Since the constraints are non-trivial, $T(e) \neq \emptyset$ for all $e \in E$. We say that a subset $S \subseteq \tilde{L}(e)$ is "good" if for all $\tau \in T(e)$, there is an $i \in \{1, \ldots, t\}$ such that $(v_i, \tau(v_i)) \in S$.

In other words, every assignment to the variables $v_1, \ldots, v_t$, that satisfies e, has a variable-label pair from S. As an illustration, suppose e is a constraint over vertices $v_1, \ldots, v_t$ such that assigning the label $1 \in [k]$ to each of $v_1, \ldots v_t$ satisfies e. Then any good subset $S \subseteq \tilde{L}(e)$ must contain at least one pair $(v_i, 1)$ for some $i \in \{1, \ldots, t\}$, and this should similarly hold for any satisfying assignment to $v_1, \ldots, v_t$ which satisfies e.Let $G(e)$ to be the set of all such "good" subsets S for the constraint $e \in E$. With these definitions we now describe the ground set G and the set of subsets $\mathcal{C}$ for our instance $\mathcal{I}$ of Set-Cover.

Ground set G. The ground set is defined as $G := \cup_e G(e)$, where the union is over all constraints $e \in E$.

Set of subsets $\mathcal{C}$. Every possible variable-label pair (v, i), there is a subset $C(v, i)$ which contains all elements from G (i.e. "good" subsets of $\tilde{L}(e)$ for all constraints e) that contain (v, i). This finishes the construction of our Set-Cover instance.

Note that every element of the ground set G can be covered by at most tk subsets from $\mathcal{C}$ and that for every constraint e, $|G(e)| \leq 2^{tk}$ and therefore $|G| \leq 2^{kt}N$. Also, since $kt = \omega(\log N)$, we obtain that $\log |G| = O(kt)$. We now analyze the YES and NO cases of $\mathcal{A}$.

YES Case. In the YES case there is a subset V' of the variables V and a labeling τ^* to $V \setminus V'$ as given in Theorem 3. We construct a cover $\mathcal{H}^*$ for the instance $\mathcal{I}$ as follows. For all variables v in $V \setminus V'$ we choose the subset $C(v, \tau^*(v))$. Additionally, for all variables v' in V' we choose *all* subsets $C(v', i)$ for all $i \in [k]$.

Let us first confirm that $\mathcal{H}^*$ indeed covers all elements of the ground set G. Consider a constraint e over variables $v_1, \ldots, v_t$. Let us first consider the case when e does not contain any variable from V'. By construction of $G(e)$, we have that for every $S \in G(e)$, there is an $i \in \{1, \ldots, t\}$ such that $(v_i, \tau^*(v_i)) \in S$. Thus $G(e)$ is covered by $\mathcal{H}^*$. Now consider the case that e contains some variables from V'. In this case, by the Extendability property of Theorem 3, τ^* can be extended by choosing labels to variables in $e \cap V'$ so that the constraint e is satisfied. Since $\mathcal{H}^*$ contains all subsets $C(v', i), i \in [k]$, for all $v' \in e \cap V'$, this implies that it covers all elements in $G(e)$. Thus, $\mathcal{H}^*$ is a valid set cover.

Now, $\mathcal{H}^*$ chooses one subset for each variable in $V \setminus V'$, and k subsets for all variables in V'. Therefore we have, $|\mathcal{H}^*| = |V \setminus V'| + k|V'| \leq |V|(1 + k^{-2})$, using the bound in Theorem 3 that $|V'|/|V| = O(k^{-3})$.

NO Case. In the NO case let $\mathcal{H} \subseteq \mathcal{C}$ be any cover. We shall prove that it cannot be small. For any variable v, let $H(v)$ be the set of variable-label pairs (v, i) where $i \in [k]$ such that $C(v, i)$ is in $\mathcal{H}$. Consider a constraint e over variables $v_1, \ldots, v_t$. Let $\tilde{H}(e) := \cup_{i=1}^{t} H(v_i)$. It can be seen that there must be a choice of variable-label pairs $(v_i, j_i) \in H(v_i)$ for each $1 \leq i \leq t$ which constitutes a satisfying assignment to e. In other words $\tilde{H}(e)$ must *contain* a satisfying assignment to e. If not, then $\tilde{L}(e) \setminus \tilde{H}(e) \in G(e)$ is "good", and is not covered by $\mathcal{H}$. Note that this also implies that $H(v)$ is non-empty for every variable v.

The above analysis suggests a randomized way to assign labels to each variable based on the cover $\mathcal{H}$. For every variable choose a label uniformly at random from the labels corresponding to the set $H(v)$. For any constraint e over variables $v_1, \ldots, v_t$, let p_e be the probability that it is satisfied. Then, $p_e \geq \frac{1}{\prod_{i=1}^{t} |H(v_i)|}$. In expectation, the number of constraints satisfied is $\sum_{e \in E} p_e$. This quantity has to be at most the soundness of the instance $\mathcal{A}$ in the NO case, i.e. $\sum_{e \in E} p_e \leq |E| k^{-t+O(\sqrt{t})}$. This implies by Markov's Inequality, that for at least half of the constraints $e \in E$ over variables $v_1, \ldots, v_t$, we have $\frac{1}{\prod_{i=1}^{t} |H(v_i)|} \leq p_e \leq 2k^{-t+O(\sqrt{t})}$, and thus,

$$\frac{\sum_{i=1}^{t} |H(v_i)|}{t} \geq \left(\prod_{i=1}^{t} |H(v_i)| \right)^{\frac{1}{t}} \geq \frac{1}{(2k^{-t+O(\sqrt{t})})^{\frac{1}{t}}}. \tag{4}$$

Since each variable in V occurs in the same number of constraints, we have the following, $(|\mathcal{H}|/|V|) = (1/|V|) \sum_{v \in V} |H(v)| = \mathrm{E}_{e \in E} \left[\frac{\sum_{i=1}^{t} |H(v_i)|}{t} \right]$, where the inner summation in the final expression is over the variables $v_1, \ldots v_t$ of the constraint e in the outer expectation. Combining the above with the fact that Equation (4) is satisfied for at least half of the coonstraints we obtain,

$$|\mathcal{H}| \geq |V| \left(\frac{1}{2} \right) \left(\frac{1}{(2k^{-t+O(\sqrt{t})})^{\frac{1}{t}}} \right) \geq |V| \Omega \left(k^{1-O\left(\frac{1}{\sqrt{t}}\right)} \right).$$

Therefore we obtain a hardness factor of $\Omega \left(k^{1-O\left(\frac{1}{\sqrt{t}}\right)} \right)$. Since $t = \theta((\log k)^2)$, the hardness factor is $\Omega(k)$. Let d be the upper bound on the number of subsets that can cover any element in G. From our construction and previous calculations we have that $d = O(kt)$, and $\log |G| = O(kt)$, where $t = \theta((\log k)^2)$. Therefore, we obtain a hardness of approximation factor of $\Omega \left(\frac{\log |G|}{(\log \log |G|)^2} \right)$. By adding a dummy subset of $|G|$ new dummy elements to the ground set we can ensure that $\Delta = \Omega(|G|)$, and by adding another dummy element and $\log |G|$ additional dummy singleton sets containing that element, we can ensure that $d = \theta(\log |G|)$. This implies a hardness factor of $\Omega \left(\frac{\log \Delta}{(\log \log \Delta)^2} \right)$, which holds under the assumption that NP$\not\subseteq$ DTIME$(n^{\mathrm{poly}(\log n)})$.

4.3 Proof of Theorem 3

We begin by stating the following theorem proved by Khot and Ponnuswami [16] on the hardness of approximating Max-3LIN : the problem of satisfying as many of a system of three variable linear equations over $\mathbb{F}_2$.

Theorem 4. *[16] Given a 7-regular instance $\mathcal{A}$ of Max-3LIN over $\mathbb{F}_2$ on n variables, unless* NP $\subseteq$ DTIME$(2^{O(\log^2 N)})$, *there is no polynomial time algorithm to distinguish between the following two cases,*

YES CASE. There is an assignment to the variables of $\mathcal{A}$ that satisfies $1 - 2^{-\Omega(\sqrt{\log n})}$ fraction of the equations (completeness).

NO CASE. No assignment to the variables of $\mathcal{A}$ satisfies more than $1 - \Omega(\log^{-3} n)$ fraction of the equations (soundness).

We shall combine the above result of [16] with the following "inner verifier" constructed by Khot and Saket [18]. A similar combination was done in [16] itself, however our construction is more optimized and we also use slightly different parameters.

Theorem 5. *Given an instance $\mathcal{A}$ of Max-3LIN over n variables with completeness $1 - c(n)$ and soundness $1 - s(n)$, for parameters m, r, k, ℓ and t there is a verifier V_{lin} which expects a proof Π where each position in the proof is expected to be labeled from $[k] = [2^r]$ such that,*

1. *V_{lin} uses $m \log n + O(\ell m r)$ random bits.*
2. *V_{lin} queries $t := \ell^2 + 2\ell$ positions from the proof.*
3. *If the instance $\mathcal{A}$ is a YES instance then there is a set Γ consisting of at most $mc(n)$ fraction (by the probability that V_{lin} queries any of them) of the positions in the proof, and an assignment τ^* to all the positions of the proof except those in Γ such that,*
   a. *(Strong Completeness) The verifier accepts on τ^* whenever none of the positions in Γ are queried.*
   b. *(Extendability) For any constraint q of the verifier which (possibly) queries positions from Γ, there is an assignment τ_q to the positions in Γ queried in q, such that τ^* extended by τ_q satisfies the constraint q.*
4. *If the instance $\mathcal{A}$ is a NO instance then the probability that the verifier accepts is at most $k^{-\ell^2} + \delta$, for $\delta^2 = (1 - s(n)^\kappa)^{(m/(\kappa r))}(k-1)^{\ell^2}$, for some universal constant κ.*

We first *regularize* the above inner verifier as follows. Let p be any position in the proof Π, and let R_p be the set of all random strings on which the verifier V_{lin} queries p. Replicate p with $|R_p|$ copies one for each string in R_p, for each position p. The new verifier simply chooses an element in R_p at random for each position p in the original query. Clearly, the new verifier queries each position with equal probability. It can also be seen that this does not change the completeness or the soundness of the verifier, and the strong completeness and extendability properties hold as well. The following lemma formalizes the modification to Theorem 5 that we can make.

Lemma 1. *The properties of the verifier V_{lin} in Theorem 4 hold with the following modifications : (i) The verifier V_{lin} queries each position with equal probability, and (ii) The number of random bits used by the verifier is $t(m \log n + O(\ell m r))$.*

In the combination of the above verifier with the Max-3Lin instance of [16] with n variables we have the completeness $c(n) = 2^{-\Omega(\sqrt{\log n})}$, and soundness $s(n) = \Omega(\log^3 n)$. We set the rest of the parameters as follows: take $m = \theta(\log^{3\kappa+3} n)$ and $r = \theta(\log \log n)$ such that $k = \theta(\log^{6\kappa+10} n)$. Additionally, we set $\ell = \theta(\log \log n)$. Now let Q be the set of all queries that the verifier makes. Clearly $\log |Q| = t(m \log n + O(\ell m r)) = O(\log^{3\kappa+5} n) = o(k)$. Furthermore, the fraction of positions in the subset Γ (as defined in Theorem 5) is

$mc(n) \leq \theta(\log^{3\kappa+3} n) \cdot 2^{-\Omega(\sqrt{\log n})} = o(k^{-3})$. Since $s(n) = \Omega(\log^{-3} n)$, we have $\delta^2 = (1 - s(n)^{\kappa})^{(m/(\kappa r))}(k-1)^{\ell^2} = 2^{-\Omega(\log^2 n)}(k-1)^{\ell^2}$. Therefore, the soundness $(k^{-\ell^2} + \delta) = k^{-t+O(\sqrt{t})}$.

It is easy to see that with this setting of the parameters, the PCP verifier obtained in the above combination is an instance of Max-CSP-Reg(t, k) with variable set V identical to the set of positions Π, the set of constraints E identical to the set of queries Q, and the subset V' same as Γ such that the properties of Theorem 3 hold. This completes the proof of Theorem 3.

4.4 Limitations to improving the hardness factor

In Section 4 we have shown a hardness factor of $\Omega\left(\frac{\log \Delta}{(\log\log \Delta)^2}\right)$ for Set-Cover where every subset has at most Δ elements, and each element of the ground set is in at most $\theta(\log \Delta)$ subsets. The two parts of this result are : a reduction to the Set-Cover problem from the Max-CSP problem; and the construction of a hard instance of Max-CSP with appropriate alphabet size, arity and hardness factor.

The second step is accomplished by running the t-query PCP test of Samorodnitsky and Trevisan [24] on a Hadamard Code based encoding of 3SAT introduced by Khot [13], which reduces the blowup of the PCP compared to the alphabet size. The t-query PCP test of [24] on an alphabet $[q]$ has a soundness of $q^{-t+O(\sqrt{t})}$. Notably, a better soundness of $q^t/O(qt)$ is achieved by more efficient PCP tests given in [8, 2]. However these tests do not combine with the Hadamard Code encoding of [13] and instead are used along with the Long Code encoding of Unique Games, which leads to a large blowup of the PCP compared to the alphabet size. Another issue with the efficient tests of [8, 2] is that t needs to be at least q^2, which will lead to weaker bounds for the canonical reduction to Set-Cover.

Thus, improving the hardness factor proved in Section 4 is connected to the question of designing efficient PCPs for Max-CSP problems over large alphabet, which in itself is a significant line of research. The current PCP techniques seem to fall short of yielding a tight bound for Set-Cover when $k = \theta(\log \Delta)$ and resolving the gap between the hardness result and the algorithmic upper bound remains an interesting open question.

Acknowledgements

We would like to thank Uriel Feige for helpful and insightful discussions on the topic.

References

1. N. Alon, D. Moshkovitz and M. Safra, Algorithmic construction of sets for k-restrictions. The ACM Transactions on Algorithms 2(2) (2006), pp. 153-177.

2. P. Austrin and E. Mossel, Approximation Resistant Predicates from Pairwise Independence. Computational Complexity 18 (2009), no. 2, pp. 249-271.
3. N. Bansal and S. Khot, Inapproximability of hypergraph vertex cover and applications to scheduling problems. In Proc. ICALP (2010), pp. 250-261.
4. R. Bar-Yehuda and S. Even, A linear-time approximation algorithm for the weighted vertex cover problem. J. Algorithms 2 (1981), no. 2, pp. 198-203.
5. V. Chvatal, A greedy heuristic for the set-covering problem. Math. Oper. Res. 4 (1979), no. 3, pp. 233-235.
6. I. Dinur, V. Guruswami, S. Khot and O. Regev, A new multilayered PCP and the hardness of hypergraph vertex cover. SIAM J. Comput. 34 (2005), no. 5, pp. 1129-1146.
7. U. Feige, A threshold of $\ln n$ for approximating set cover. J. ACM 45 (1998), no. 4, pp. 634-652.
8. V. Guruswami and P. Raghavendra, Constraint Satisfaction over a Non-Boolean Domain: Approximation Algorithms and Unique-Games Hardness. In Proc. APPROX-RANDOM (2008), pp. 77-90.
9. E. Halperin, Improved Approximation Algorithms for the Vertex Cover Problem in Graphs and Hypergraphs. SIAM J. Comput. 31(2002), no. 5, pp. 1608-1623.
10. D. Hochbaum, Approximation algorithms for the set covering and vertex cover problems. SIAM J. Comput. 11 (1982), no. 3, pp. 555-556.
11. J. Holmerin, Improved Inapproximability Results for Vertex Cover on k-Uniform Hypergraphs. In Proc. ICALP (2002), pp. 1005-1016.
12. D. Johnson, Approximation algorithms for combinatorial problems. J. Comput. Syst. Sci. 9 (1974), pp. 256-278.
13. S. Khot. Improved inaproximability results for maxclique, chromatic number and approximate graph coloring. In Proc. FOCS (2001), pp. 600-609.
14. S. Khot, On the power of unique 2-prover 1-round games. In Proc. STOC (2002), pp. 767-775.
15. S. Khot, Online lecture notes for Probabilistically Checkable Proofs and Hardness of Approximation, Lecture 3 (scribed by Deeparnab Chakrabarty), available at www.cs.nyu.edu/ khot/pcp-lecnotes/lec3.ps.
16. S. Khot and A. Ponnuswami, Better Inapproximability Results for MaxClique, Chromatic Number and Min-3Lin-Deletion. In Proc. ICALP (2006), pp. 226-237.
17. S. Khot and O. Regev, Vertex cover might be hard to approximate to within $2-\varepsilon$. J. Comput. System Sci. 74 (2008), no. 3, pp. 335-349.
18. S. Khot and R. Saket, Hardness of Minimizing and Learning DNF Expressions. In Proc. FOCS (2008), pp. 231-240.
19. M. Krivelevich, Approximate set covering in uniform hypergraphs. J. Algorithms 25 (1997), no. 1, pp. 118-143.
20. L. Lovasz, On the ratio of the optimal integral and fractional covers. Disc. Math. 13 (1975), pp. 383-390.
21. C. Lund and M. Yannakakis, On the hardness of approximating minimization problems. J. ACM 31 (1994), no. 5, pp. 960-981.
22. M. Okun, On the approximation of the vertex cover problem in hypergraphs. Discrete Optimization 2 (2005), no. 1, pp. 101-111.
23. R. Raz and M. Safra, A sub-constant error-probability low-degree test, and a subconstant error-probability PCP characterization of NP. In Proc. STOC (2007), pp. 475-484.
24. A. Samorodnitsky and L. Trevisan, A PCP characterization of NP with optimal amortized query complexity. In Proc. STOC (2000), pp. 191-199.