Analysis of Approximation Algorithms for k-Set Cover Using Factor-Revealing Linear Programs*

Stavros Athanassopoulos, Ioannis Caragiannis, and Christos Kaklamanis

Research Academic Computer Technology Institute &
Department of Computer Engineering and Informatics
University of Patras, 26500 Rio, Greece

Abstract. We present new combinatorial approximation algorithms for k-set cover. Previous approaches are based on extending the greedy algorithm by efficiently handling small sets. The new algorithms further extend them by utilizing the natural idea of computing large packings of elements into sets of large size. Our results improve the previously best approximation bounds for the k-set cover problem for all values of $k \geq 6$. The analysis technique could be of independent interest; the upper bound on the approximation factor is obtained by bounding the objective value of a *factor-revealing* linear program.

1 Introduction

Set cover is a fundamental combinatorial optimization problem with many applications. Instances of the problem consist of a set of elements V and a collection S of subsets of V and the objective is to select a subset of S of minimum cardinality so that every element is covered, i.e., it is contained in at least one of the selected sets.

The natural greedy algorithm starts with an empty solution and augments it until all elements are covered by selecting a set that contains the maximum number of elements that are not contained in any of the previously selected sets. Denoting by n the number of elements, it is known by the seminal papers of Johnson [10], Lovasz [13], and Chvátal [1] that the greedy algorithm has approximation ratio H_n. The tighter analysis of Slavík [14] improves the upper bound on the approximation ratio to $\ln n - \ln\ln n + O(1)$. Asymptotically, these bounds are tight due to a famous inapproximability result of Feige [3] which states that there is no $(1 - \epsilon)\ln n$-approximation algorithm for set cover unless all problems in NP have deterministic algorithms running in subexponential time $O\left(n^{\mathrm{polylog}(n)}\right)$.

An interesting variant is k-set cover where every set of S has size at most k. Without loss of generality, we may assume that S is closed under subsets. In this case, the greedy algorithm can be equivalently expressed as follows:

Greedy phases: For $i = k$ down to 1 do:
> Choose a maximal collection of disjoint i-sets.

An i-set is a set that contains exactly i previously uncovered elements and a collection T of disjoint i-sets is called maximal if any other i-set intersects some of the sets in T.

A tight bound of H_k on the approximation ratio of the greedy algorithm is well known in this case. Since the problem has many applications for particular values of k and due to its interest from a complexity-theoretic viewpoint, designing algorithms with improved second order terms in their approximation ratio has received much attention. Currently, there are algorithms with approximation ratio $H_k - c$ where c is a constant. Goldschmidt et al. [4] were the first to present a modified greedy algorithm with $c = 1/6$. This value was improved to $1/3$ by Halldórsson [6] and to $1/2$ by Duh and Fürer [2]. Recently, Levin [12] further improved the constant to $98/195 \approx 0.5026$ for $k \geq 4$. On the negative side, Trevisan [15] has shown that, unless subexponential-time deterministic algorithms for NP exist, no polynomial-time algorithm has an approximation ratio of $\ln k - \Omega(\ln \ln k)$.

The main idea that has been used in order to improve the performance of the greedy algorithm is to efficiently handle small sets. The algorithm of Goldschmidt et al. [4] uses a matching computation to accommodate as many elements as possible in sets of size 2 when no set of size at least 3 contains new elements. The algorithms of Halldórsson [5,6] and Duh and Fürer [2] handle efficiently sets of size 3. The algorithm of [2] is based on a semi-local optimization technique. Levin's improvement [12] extends the algorithm of [2] by efficiently handling of sets of size 4.

A natural but completely different idea is to replace the phases of the greedy algorithm associated with large sets with set-packing phases that also aim to maximize the number of new elements covered by large maximal sets. The approximation factor is not getting worse (due to the maximality condition) while it has been left as an open problem in [12] whether it leads to any improvement in the approximation bound. This is the aim of the current paper: we show that by substituting the greedy phases in the algorithms of Duh and Fürer with packing phases, we obtain improved approximation bounds for every $k \geq 6$ which approaches $H_k - 0.5902$ for large values of k.

In particular, we will use algorithms for the k-set packing problem which is defined as follows. An instance of k-set packing consists of a set of elements V and a collection S of subsets of V each containing exactly k elements. The objective is to select as many as possible disjoint sets from S. When $k = 2$, the problem is equivalent to maximum matching in graphs and, hence, it is solvable in polynomial time. For $k \geq 3$, the problem is APX-hard [11]; the best known inapproximability bound for large k is asymptotically $O\left(\frac{\log k}{k}\right)$ [7]. Note that any maximal collection of disjoint subsets yields a $1/k$-approximate solution. The best known algorithms have approximation ratio $\frac{2-\epsilon}{k}$ for any $\epsilon > 0$ [8] and are based on local search; these are the algorithms used by the packing phases of our algorithms.

Our analysis is based on the concept of factor-revealing LPs which has been introduced in a different context in [9] for the analysis of approximation algorithms for facility location. We show that the approximation factor is upper-bounded by the maximum objective value of a factor-revealing linear program. Hence, no explicit reasoning about the structure of the solution computed by the algorithms is required. Instead, the performance guarantees of the several phases are used as black boxes in the definition of the constraints of the LP.

The rest of the paper is structured as follows. We present the algorithms of Duh and Fürer [2] as well as our modifications in Section 2. The analysis technique is discussed in Section 3 where we present the factor-revealing LP lemmas. Then, in Section 4, we present a part of the proofs of our main results; proofs that have been omitted from this extended abstract will appear in the final version of the paper. We conclude in Section 5.

2   Algorithm Description

In this section we present the algorithms considered in this paper. We start by giving an overview of the related results in [2]; then, we present our algorithms and main statements.

Besides the greedy algorithm, local search algorithms have been used for the k-cover problem, in particular for small values of k. Pure local search starts with any cover and works in steps. In each step, the current solution is improved by replacing a constant number of sets with a (hopefully smaller) number of other sets in order to obtain a new cover. Duh and Fürer introduced the technique of semi-local optimization which extends pure local search. In terms of 3-set cover, the main idea behind semi-local optimization is that once the sets of size 3 have been selected, computing the minimum number of sets of size 2 and 1 in order to complete the covering can be done in polynomial time by a matching computation. Hence, a semi-local (s, t)-improvement step for 3-set cover consists of the deletion of up to t 3-sets from the current cover and the insertion of up to s 3-sets and the minimum necessary 2-sets and 1-sets that complete the cover. The quality of an improvement is defined by the total number of sets in the cover while in case of two covers of the same size, the one with the smallest number of 1-sets is preferable. Semi-local optimization for $k \geq 4$ is much similar; local improvements are now defined on sets of size at least 3 while 2-sets and 1-sets are globally changed. The analysis of [2] shows that the best choice of the parameters (s, t) is $(2, 1)$.

Theorem 1 (Duh and Fürer [2]). *Consider an instance of k-set cover whose optimal solution has a_i i-sets. Then, the semi-local $(2, 1)$-optimization algorithm has cost at most $a_1 + a_2 + \sum_{i=3}^{k} \frac{i+1}{3} a_i$.*

Proof (outline). The proof of [2] proceeds as follows. Let b_i be the number of i-sets in the solution. First observe that $\sum_{i=1}^{k} i b_i = \sum_{i=1}^{k} i a_i$. Then, the following two properties of semi-local $(2, 1)$-optimization are proved: $b_1 \leq a_1$ and $b_1 + b_2 \leq \sum_{i=1}^{k} a_i$. The theorem follows by summing the three inequalities.   □

This algorithm has been used as a basis for the following algorithms that approximate k-set cover. We will call them $\mathsf{GSLI}_{k,\ell}$ and $\mathsf{GRSLI}_{k,\ell}$, respectively.

>*Algorithm* $\mathsf{GSLI}_{k,\ell}$.
>**Greedy phases:** For $i = k$ down to $\ell + 1$ do:
>>Choose a maximal collection of i-sets.
>**Semi-local optimization phase:** Run the semi-local $(2, 1)$-optimization
>>algorithm on the remaining instance.

Theorem 2 (Duh and Fürer [2]). *Algorithm* $\mathsf{GSLI}_{k,4}$ *has approximation ratio* $H_k - 5/12$.

>*Algorithm* $\mathsf{GRSLI}_{k,\ell}$
>**Greedy phases:** For $i = k$ down to $\ell + 1$ do:
>>Choose a maximal collection of i-sets.
>**Restricted phases:** For $i = \ell$ down to 4 do:
>>Choose a maximal collection of disjoint i-sets so that the choice of
>>these i-sets does not increase the number of 1-sets in the final solution.
>**Semi-local optimization phase:** Run the semi-local optimization algorithm on the remaining instance.

Theorem 3 (Duh and Fürer [2]). *Algorithm* $\mathsf{GRSLI}_{k,5}$ *has approximation ratio* $H_k - 1/2$.

We will modify the algorithms above by replacing each greedy phase with a packing phase for handling sets of not very small size. We use the local search algorithms of Hurkens and Schrijver [8] in each packing phase. The modified algorithms are called $\mathsf{PSLI}_{k,\ell}$ and $\mathsf{PRSLI}_{k,\ell}$, respectively.

A local search algorithm for set packing uses a constant parameter p (informally, this is an upper bound on the number of local improvements performed at each step) and, starting with an empty packing Π, repeatedly updates Π by replacing any set of $s < p$ sets of Π with $s+1$ sets so that feasibility is maintained and until no replacement is possible. Clearly, the algorithm runs in polynomial time. It has been analyzed in [8] (see also [5] for related investigations).

Theorem 4 (Hurkens and Schrijver [8]). *The local search t-set packing algorithm that performs at most p local improvements at each step has approximation ratio* $\rho_t \geq \frac{2(t-1)^r - t}{t(t-1)^r - t}$, *if* $p = 2r - 1$ *and* $\rho_t \geq \frac{2(t-1)^r - 2}{t(t-1)^r - 2}$, *if* $p = 2r$.

As a corollary, for any constant $\epsilon > 0$, we obtain a $\frac{2-\epsilon}{t}$-approximation algorithm for t-set packing by using $p = O(\log_t 1/\epsilon)$ local improvements. Our algorithms $\mathsf{PSLI}_{k,\ell}$ and $\mathsf{PRSLI}_{k,\ell}$ simply replace each of the greedy phases of the algorithms $\mathsf{GSLI}_{k,\ell}$ and $\mathsf{GRSLI}_{k,\ell}$, respectively, with the following packing phase:

>**Packing phases:** For $i = k$ down to $\ell + 1$ do:
>>Select a maximal collection of disjoint i-sets using a $\frac{2-\epsilon}{i}$-approximation
>>local search i-set packing algorithm.

Our first main result (Theorem 5) is a statement on the performance of algorithm $\mathsf{PSLI}_{k,\ell}$ (for $\ell = 4$ which is the best choice).

Theorem 5. *For any constant $\epsilon > 0$, algorithm $\mathsf{PSLI}_{k,4}$ has approximation ratio at most $H_{k/2} + \frac{1}{6} + \epsilon$ for even $k \geq 6$, at most $2H_k - H_{\frac{k-1}{2}} - \frac{11}{9} + \epsilon$ for $k \in \{5, 7, 9, 11, 13\}$, and at most $H_{\frac{k-1}{2}} + \frac{1}{6} + \frac{2}{k} - \frac{1}{k-1} + \epsilon$ for odd $k \geq 15$.*

Algorithm $\mathsf{PSLI}_{k,4}$ outperforms algorithm $\mathsf{GSLI}_{k,4}$ for $k \geq 5$, algorithm $\mathsf{GRSLI}_{k,5}$ for $k \geq 19$, as well as the improvement of Levin [12] for $k \geq 21$. For large values of k, the approximation bound approaches $H_k - c$ with $c = \ln 2 - 1/6 \approx 0.5264$. Algorithm $\mathsf{PSLI}_{k,\ell}$ has been mainly included here in order to introduce the analysis technique. As we will see in the next section, the factor-revealing LP is simpler in this case. Algorithm $\mathsf{PRSLI}_{k,\ell}$ is even better; its performance (for $\ell = 5$) is stated in the following.

Theorem 6. *For any constant $\epsilon > 0$, algorithm $\mathsf{PRSLI}_{k,5}$ has approximation ratio at most $2H_k - H_{\frac{k-1}{2}} - \frac{77}{60} + \epsilon$ for odd $k \geq 7$, at most $\frac{461}{240} + \epsilon$ for $k = 6$, and at most $2H_k - H_{k/2} - \frac{77}{60} + \frac{2}{k} - \frac{1}{k-1} + \epsilon$ for even $k \geq 8$.*

Algorithm $\mathsf{PRSLI}_{k,5}$ achieves better approximation ratio than the algorithm of Levin [12] for every $k \geq 6$. For example, the approximation ratio of $\frac{461}{240} \approx 1.9208$ for 6-set cover improves the previous bound of $H_6 - \frac{98}{195} \approx 1.9474$. For large values of k, the approximation bound approaches $H_k - c$ with $c = 77/60 - \ln 2 \approx 0.5902$. See Table 1 for a comparison between the algorithms discussed in this section.

Table 1. Comparison of the approximation ratio of the algorithms $\mathsf{GSLI}_{k,4}$, $\mathsf{PSLI}_{k,4}$, $\mathsf{GRSLI}_{k,5}$, the algorithm in [12] and algorithm $\mathsf{PRSLI}_{k,5}$ for several values of k

k	$\mathsf{GSLI}_{k,4}$ [2]	$\mathsf{PSLI}_{k,4}$	$\mathsf{GRSLI}_{k,5}$ [2]	[12]	$\mathsf{PRSLI}_{k,5}$
3	1.3333	1.3333	1.3333	1.3333	1.3333
4	1.6667	1.6667	1.5833	1.5808	1.5833
5	1.8667	1.8444	1.7833	1.7801	1.7833
6	2.0333	2	1.95	1.9474	1.9208
7	2.1762	2.1429	2.0929	2.0903	2.0690
8	2.3012	2.25	2.2179	2.2153	2.1762
9	2.4123	2.3524	2.3290	2.3264	2.2917
10	2.5123	2.45	2.4290	2.4264	2.3802
19	3.1311	3.0453	3.0477	3.0452	2.9832
20	3.1811	3.0956	3.0977	3.0952	3.0305
21	3.2287	3.1409	3.1454	3.1428	3.0784
22	3.2741	3.1865	3.1908	3.1882	3.1217
50	4.0825	3.9826	3.9992	3.9966	3.9187
75	4.4847	4.3814	4.4014	4.3988	4.3178
100	4.7707	4.6659	4.6874	4.6848	4.6021
large k	$H_k - 0.4167$	$H_k - 0.5264$	$H_k - 0.5$	$H_k - 0.5026$	$H_k - 0.5902$

3   Analysis Through Factor-Revealing LPs

Our proofs on the approximation guarantee of our algorithms essentially follow by computing upper bounds on the objective value of factor-revealing linear programs whose constraints capture simple invariants maintained in the phases of the algorithms.

Consider an instance (V, S) of k-set cover. For any phase of the algorithms associated with i ($i = \ell, ..., k$ for algorithm $\mathsf{PSLI}_{k,\ell}$ and $i = 3, ..., k$ for algorithm $\mathsf{PRSLI}_{k,\ell}$), consider the instance (V_i, S_i) where V_i contains the elements in V that have not been covered in previous phases and S_i contains the sets of S which contain only elements in V_i. Denote by $\mathcal{OPT}^i$ an optimal solution of instance (V_i, S_i); we also denote the optimal solution $\mathcal{OPT}^k$ of $(V_k, S_k) = (V, S)$ by $\mathcal{OPT}$. Since S is closed under subsets, without loss of generality, we may assume that $\mathcal{OPT}^i$ contains disjoint sets. Furthermore, it is clear that $|\mathcal{OPT}^{i-1}| \leq |\mathcal{OPT}^i|$ for $i \leq k$, i.e., $|\mathcal{OPT}^i| \leq |\mathcal{OPT}|$.

For a phase of algorithm $\mathsf{PSLI}_{k,\ell}$ or $\mathsf{PRSLI}_{k,\ell}$ associated with i, denote by $a_{i,j}$ the ratio of the number of j-sets in $\mathcal{OPT}^i$ over $|\mathcal{OPT}|$. Since $|\mathcal{OPT}^i| \leq |\mathcal{OPT}|$, we obtain that

$$\sum_{j=1}^{i} a_{i,j} \leq 1. \tag{1}$$

The i-set packing algorithm executed on packing phase associated with i includes in i-sets the elements in $V_i \backslash V_{i-1}$. Since $V_{i-1} \subseteq V_i$, their number is

$$|V_i \backslash V_{i-1}| = |V_i| - |V_{i-1}| = \left(\sum_{j=1}^{i} j a_{i,j} - \sum_{j=1}^{i-1} j a_{i-1,j} \right) |\mathcal{OPT}|. \tag{2}$$

Denote by ρ_i the approximation ratio of the i-set packing algorithm executed on the phase associated with i. Since at the beginning of the packing phase associated with i, there exist at least $a_{i,i}|\mathcal{OPT}|$ i-sets, the i-set packing algorithm computes at least $\rho_i a_{i,i}|\mathcal{OPT}|$ i-sets, i.e., covering at least $i\rho_i a_{i,i}|\mathcal{OPT}|$ elements from sets in $\mathcal{OPT}^i$. Hence, $|V_i \backslash V_{i-1}| \geq i\rho_i a_{i,i}|\mathcal{OPT}|$, and (2) yields

$$\sum_{j=1}^{i-1} j a_{i-1,j} - \sum_{j=1}^{i-1} j a_{i,j} - i(1 - \rho_i)a_{i,i} \leq 0. \tag{3}$$

So far, we have defined all constraints for the factor-revealing LP of algorithm $\mathsf{PSLI}_{k,\ell}$. Next, we bound from above the number of sets computed by algorithm $\mathsf{PSLI}_{k,\ell}$ as follows. Let t_i be the number of i-sets computed by the i-set packing algorithm executed at the packing phase associated with $i \geq \ell + 1$. Clearly,

$$t_i = \frac{1}{i}|V_i \backslash V_{i-1}| = \left(\frac{1}{i} \sum_{j=1}^{i} j a_{i,j} - \frac{1}{i} \sum_{j=1}^{i-1} j a_{i-1,j} \right) |\mathcal{OPT}|. \tag{4}$$

By Theorem 1, we have that

$$
t_\ell \leq \left(a_{\ell,1} + a_{\ell,2} + \sum_{j=3}^{\ell} \frac{j+1}{3} a_{\ell,j} \right) |\mathcal{OPT}|. \tag{5}
$$

Hence, by (4) and (5), it follows that the approximation guarantee of algorithm $\mathsf{PSLI}_{k,\ell}$ is

$$
\frac{\sum_{i=4}^{k} t_i}{|\mathcal{OPT}|} \leq \sum_{i=\ell+1}^{k} \frac{1}{i} \left(\sum_{j=1}^{i} j a_{i,j} - \sum_{j=1}^{i-1} j a_{i-1,j} \right) + a_{\ell,1} + a_{\ell,2} + \sum_{j=3}^{\ell} \frac{j+1}{3} a_{\ell,j}
$$

$$
= \frac{1}{k} \sum_{j=1}^{k} j a_{k,j} + \sum_{i=\ell+1}^{k-1} \frac{1}{i(i+1)} \sum_{j=1}^{i} j a_{i,j} + \frac{\ell}{\ell+1} a_{\ell,1}
$$

$$
+ \frac{\ell-1}{\ell+1} a_{\ell,2} + \sum_{j=3}^{\ell} \left(\frac{j+1}{3} - \frac{j}{\ell+1} \right) a_{\ell,j} \tag{6}
$$

Hence, an upper bound on the approximation ratio of algorithm $\mathsf{PSLI}_{k,\ell}$ follows by maximizing the right part of (6) subject to the constraints (1) for $i = \ell, ..., k$ and (3) for $i = \ell + 1, ..., k$ with variables $a_{i,j} \geq 0$ for $i = \ell, ..., k$ and $j = 1, ..., i$. Formally, we have proved the following statement.

Lemma 1. *The approximation ratio of algorithm* $\mathsf{PSLI}_{k,\ell}$ *when a* ρ_i-*approximation i-set packing algorithm is used at phase i for* $i = \ell + 1, ..., k$ *is upperbounded by the maximum objective value of the following linear program:*

$$
\text{maximize } \frac{1}{k} \sum_{j=1}^{k} j a_{k,j} + \sum_{i=\ell+1}^{k-1} \frac{1}{i(i+1)} \sum_{j=1}^{i} j a_{i,j} + \frac{\ell}{\ell+1} a_{\ell,1}
$$

$$
+ \frac{\ell-1}{\ell+1} a_{\ell,2} + \sum_{j=3}^{\ell} \left(\frac{j+1}{3} - \frac{j}{\ell+1} \right) a_{\ell,j}
$$

$$
\text{subject to } \sum_{j=1}^{i} a_{i,j} \leq 1, i = \ell, ..., k
$$

$$
\sum_{j=1}^{i-1} j a_{i-1,j} - \sum_{j=1}^{i-1} j a_{i,j} - i(1-\rho_i) a_{i,i} \leq 0, i = \ell+1, ..., k
$$

$$
a_{i,j} \geq 0, i = \ell, ..., k, j = 1, ..., i
$$

Each packing or restricted phase of algorithm $\mathsf{PRSLI}_{k,\ell}$ satisfies (3); a restricted phase associated with $i = 4, ..., \ell$ computes a maximal i-set packing and, hence, $\rho_i = 1/i$ in this case.

In addition, the restricted phases impose extra constraints. Denote by b_1, b_2, and b_3 the ratio of the number of 1-sets, 2-sets, and 3-sets computed by the

semi-local optimization phase over $|\mathcal{OPT}|$, respectively. The restricted phases guarantee that the number of the 1-sets in the final solution does not increase, and, hence,

$$b_1 \leq a_{i,1}, \text{ for } i = 3, ..., \ell. \tag{7}$$

Following the proof of Theorem 1, we obtain $b_1 + b_2 \leq a_{3,1} + a_{3,2} + a_{3,3}$ while it is clear that $b_1 + 2b_2 + 3b_3 = a_{3,1} + 2a_{3,2} + 3a_{3,3}$. We obtain that the number t_3 of sets computed during the semi-local optimization phase of algorithm $\mathsf{PRSLI}_{k,\ell}$ is

$$
\begin{aligned}
t_3 &= (b_1 + b_2 + b_3)|\mathcal{OPT}| \\
&\leq \left(\frac{b_1}{3} + \frac{b_1 + b_2}{3} + \frac{b_1 + 2b_2 + 3b_3}{3} \right) |\mathcal{OPT}| \\
&\leq \left(\frac{1}{3}b_1 + \frac{2}{3}a_{3,1} + a_{3,2} + \frac{4}{3}a_{3,3} \right) |\mathcal{OPT}|.
\end{aligned}
\tag{8}
$$

Reasoning as before, we obtain that (4) gives the number t_i of i-sets computed during the packing or restricted phase associated with $i = 4, ..., k$. By (4) and (8), we obtain that the performance guarantee of algorithm $\mathsf{PRSLI}_{k,\ell}$ is

$$
\begin{aligned}
\frac{\sum_{i=3}^k t_i}{|\mathcal{OPT}|} &\leq \sum_{i=4}^k \frac{1}{i} \left(\sum_{j=1}^i ja_{i,j} - \sum_{j=1}^{i-1} ja_{i-1,j} \right) + \frac{2}{3}a_{3,1} + a_{3,2} + \frac{4}{3}a_{3,3} + \frac{1}{3}b_1 \\
&= \frac{1}{k}\sum_{j=1}^k ja_{k,j} + \sum_{i=4}^{k-1} \frac{1}{i(i+1)} \sum_{j=1}^i ja_{i,j} + \frac{5}{12}a_{3,1} + \frac{1}{2}a_{3,2} \\
&\quad + \frac{7}{12}a_{3,3} + \frac{1}{3}b_1
\end{aligned}
\tag{9}
$$

Hence, an upper bound on the approximation ratio of algorithm $\mathsf{PRSLI}_{k,\ell}$ follows by maximizing the right part of (9) subject to the constraints (1) for $i = 3, ..., k$, (3) for $i = 4, ..., k$, and (7), with variables $a_{i,j} \geq 0$ for $i = 3, ..., k$ and $j = 1, ..., i$, and $b_1 \geq 0$. Formally, we have proved the following statement.

Lemma 2. *The approximation ratio of algorithm* $\mathsf{PRSLI}_{k,\ell}$ *when a* ρ_i*-approximation* i*-set packing algorithm is used at phase* i *for* $i = \ell + 1, ..., k$ *is upperbounded by the maximum objective value of the following linear program:*

$$\text{maximize } \frac{1}{k}\sum_{j=1}^k ja_{k,j} + \sum_{i=4}^{k-1} \frac{1}{i(i+1)} \sum_{j=1}^i ja_{i,j} + \frac{5}{12}a_{3,1} + \frac{1}{2}a_{3,2} + \frac{7}{12}a_{3,3} + \frac{1}{3}b_1$$

$$\text{subject to } \sum_{j=1}^i a_{i,j} \leq 1, i = 3, ..., k$$

$$\sum_{j=1}^{i-1} ja_{i-1,j} - \sum_{j=1}^{i-1} ja_{i,j} - i(1 - \rho_i)a_{i,i} \leq 0, i = \ell + 1, ..., k$$

$$\sum_{j=1}^{i-1} ja_{i-1,j} - \sum_{j=1}^{i-1} ja_{i,j} - (i-1)a_{i,i} \le 0, i = 4, ..., \ell$$

$$b_1 - a_{i,1} \le 0, i = 3, ..., \ell$$

$$a_{i,j} \ge 0, i = 3, ..., k, j = 1, ..., i$$

$$b_1 \ge 0$$

4   Proofs of Main Theorems

We are now ready to prove our main results. We can show that the maximum objective values of the factor-revealing LPs for algorithms $\mathsf{PSLI}_{k,4}$ and $\mathsf{PRSLI}_{k,5}$ are upper-bounded by the values stated in Theorems 5 and 6. In order to prove this, it suffices to find feasible solutions to the dual LPs that have these values as objective values. In the following, we prove Theorem 5 by considering the case when k is even. The proof for odd k as well as the proof of Theorem 6 (which is slightly more complicated) will appear in the final version of the paper.

Proof of Theorem 5. The dual of the factor-revealing LP of algorithm $\mathsf{PSLI}_{k,4}$ is:

$$\text{minimize} \sum_{i=4}^{k} \beta_i$$

$$\text{subject to } \beta_4 + \gamma_5 \ge \frac{4}{5}$$

$$\beta_4 + 2\gamma_5 \ge \frac{3}{5}$$

$$\beta_4 + 3\gamma_5 \ge \frac{11}{15}$$

$$\beta_4 + 4\gamma_5 \ge \frac{13}{15}$$

$$\beta_i + j\gamma_{i+1} - j\gamma_i \ge \frac{j}{i(i+1)}, i = 5, ..., k-1, j = 1, ..., i-1$$

$$\beta_i + i\gamma_{i+1} - (i-2+\epsilon)\gamma_i \ge \frac{1}{i+1}, i = 5, ..., k-1$$

$$\beta_k - j\gamma_k \ge \frac{j}{k}, j = 1, ..., k-1$$

$$\beta_k - (k-2+\epsilon)\gamma_k \ge 1$$

$$\beta_i \ge 0, i = 4, ..., k$$

$$\gamma_i \ge 0, i = 5, ..., k$$

We consider only the case of even k. We set $\gamma_k = \frac{1}{k(k-1)}$ and $\gamma_{k-1} = 0$. If $k \ge 8$, we set $\gamma_i = \gamma_{i+2} + \frac{2}{i(i+1)(i+2)}$ for $i = 5, ..., k-2$. We also set $\beta_k = 1 + (k-2+\epsilon)\gamma_k$, $\beta_4 = \frac{13}{15} - 4\gamma_5$ and

$$\beta_i = \frac{1}{i+1} - i\gamma_{i+1} + (i-2+\epsilon)\gamma_i$$

for $i = 5, ..., k-1$.

We will show that all the constraints of the dual LP are satisfied. Clearly, $\gamma_i \geq 0$ for $i = 5, ..., k$. Observe that $\gamma_{k-1} + \gamma_k = \frac{1}{k(k-1)}$. If $k \geq 8$, by the definition of γ_i for $i = 5, ..., k-2$, we have that

$$\gamma_i + \gamma_{i+1} - \frac{1}{i(i+1)} = \gamma_{i+1} + \gamma_{i+2} + \frac{2}{i(i+1)(i+2)} - \frac{1}{i(i+1)}$$

$$= \gamma_{i+1} + \gamma_{i+2} - \frac{1}{(i+1)(i+2)}$$

and, hence,

$$\gamma_i + \gamma_{i+1} - \frac{1}{i(i+1)} = \gamma_{k-1} + \gamma_k - \frac{1}{k(k-1)} = 0,$$

i.e., $\gamma_i + \gamma_{i+1} = \frac{1}{i(i+1)}$ for $i = 5, ..., k-1$. Now, the definition of β_i's yields

$$\begin{aligned}
\beta_i &= \frac{1}{i+1} - i\gamma_{i+1} + (i - 2 + \epsilon)\gamma_i \\
&= \frac{i-1}{i(i+1)} - (i-1)\gamma_{i+1} + (i-1)\gamma_i + \epsilon\gamma_i \\
&\geq \frac{j}{i(i+1)} - j\gamma_{i+1} + j\gamma_i \\
&\geq 0
\end{aligned}$$
(10)

for $i = 5, ..., k-1$ and $j = 1, ..., i-1$. Hence, all the constraints on β_i for $i = 5, ..., k-1$ are satisfied.

The constraints on β_k are also maintainted. Since $\gamma_k = \frac{1}{k(k-1)} \leq \frac{1}{k}$ we have that $\beta_k = 1 + (k - 2 + \epsilon)\gamma_k \geq \frac{k-1}{k} + (k-1)\gamma_k + \epsilon\gamma_k \geq \frac{j}{k} + j\gamma_k \geq 0$ for $j = 1, ..., k-1$. It remains to show that the constraints on β_4 are also satisfied. It suffices to show that $\gamma_5 \leq 1/45$. This is clear when $k = 6$. If $k \geq 8$, consider the equalities $\gamma_i + \gamma_{i+1} = \frac{1}{i(i+1)}$ for odd $i = 5, ..., k-3$ and $-\gamma_i - \gamma_{i+1} = -\frac{1}{i(i+1)}$ for even $i = 6, ..., k-2$. Summing them, and since $\gamma_{k-1} = 0$, we obtain that

$$\begin{aligned}
\gamma_5 &= \sum_{i=3}^{k/2-1} \left(\frac{1}{2i(2i-1)} - \frac{1}{2i(2i+1)} \right) \\
&= \sum_{i=3}^{k/2-1} \left(\frac{1}{2i-1} - \frac{1}{2i} - \frac{1}{2i} + \frac{1}{2i+1} \right) \\
&= \sum_{i=3}^{k/2-1} \left(\frac{1}{2i-1} + \frac{1}{2i+1} \right) - \sum_{i=3}^{k/2-1} \frac{1}{i} \\
&= -\frac{1}{5} + \sum_{i=3}^{k/2-1} \frac{2}{2i-1} + \frac{1}{k-1} - H_{k/2-1} + \frac{3}{2} \\
&= -\frac{1}{5} + 2H_{k-2} - H_{k/2-1} - \frac{8}{3} + \frac{1}{k-1} - H_{k/2-1} + \frac{3}{2}
\end{aligned}$$

$$= 2H_{k-2} - 2H_{k/2-1} + \frac{1}{k-1} - \frac{41}{30}$$

$$\leq 2\ln 2 - \frac{41}{30}$$

$$\leq 1/45$$

The first inequality follows since $2H_{k-2} - 2H_{k/2-1} + \frac{1}{k-1}$ is increasing on k (this can be easily seen by examining the original definition of γ_5) and since $\lim_{t\to\infty} H_t/\ln t = 1$.

We have shown that all the constraints of the dual LP are satisfied, i.e., the solution is feasible. In order to compute the objective value, we use the definition of β_i's and equality (10). We obtain

$$\sum_{i=4}^{k} \beta_i = \beta_4 + \beta_5 + \sum_{i=3}^{k/2-1} (\beta_{2i} + \beta_{2i+1}) + \beta_k$$

$$= \frac{13}{15} - 4\gamma_5 + \frac{2}{15} + 4\gamma_5 - 4\gamma_6 + \sum_{i=3}^{k/2-1} \left(\frac{1}{2i+1} - 2i\gamma_{2i+1} + (2i-2)\gamma_{2i} \right.$$

$$\left. + \frac{2i}{(2i+1)(2i+2)} - 2i\gamma_{2i+2} + 2i\gamma_{2i+1} \right) + 1 + (k-2)\gamma_k + \epsilon\sum_{i=5}^{k}\gamma_i$$

$$= 2 + \sum_{i=3}^{k/2-1} \frac{1}{i+1} - 4\gamma_6 + \sum_{i=3}^{k/2-1} ((2i-2)\gamma_{2i} - 2i\gamma_{2i+2}) + (k-2)\gamma_k$$

$$+ \epsilon\sum_{i=5}^{k}\gamma_i$$

$$\leq H_{k/2} + 1/6 + \epsilon$$

where the last inequality follows since $\sum_{i=5}^{k} \gamma_i \leq \sum_{i=5}^{k} \frac{1}{i(i-1)} = \sum_{i=5}^{k} \left(\frac{1}{i-1} - \frac{1}{i} \right) = 1/4 - 1/k$.

By duality, $\sum_{i=4}^{k} \beta_i$ is an upper bound on the maximum objective value of the factor-revealing LP. The theorem follows by Lemma 1. □

5   Extensions

We have experimentally verified using Matlab that our upper bounds are tight in the sense that they are the maximum objective values of the factor-revealing LPs (ignoring the ϵ term in the approximation bound). Our analysis technique can also be used to provide simpler proofs of the results in [2] (i.e., Theorems 2 and 3); this is left as an exercise to the reader. The several cases that are considered in the proofs of [2] are actually included as constraints in the factor-revealing LPs which are much simpler than the ones for algorithms $\mathsf{PSLI}_{k,\ell}$ and $\mathsf{PRSLI}_{k,\ell}$. Furthermore, note that we have not combined our techniques with the recent algorithm of Levin [12] that handles sets of size 4 using a restricted local search phase. It is tempting to conjecture that further improvements are possible.

References

1. Chvátal, V.: A greedy hueristic for the set-covering problem. Mathematics of Operations Research 4, 233–235 (1979)
2. Duh, R., Fürer, M.: Approximation of k-set cover by semi local optimization. In: Proceedings of the 29th Annual ACM Symposium on Theory of Computing (STOC '97), pp. 256–264. ACM Press, New York (1997)
3. Feige, U.: A threshold of $\ln n$ for approximating set cover. Journal of the ACM 45(4), 634–652 (1998)
4. Goldschmidt, O., Hochbaum, D., Yu, G.: A modified greedy heuristic for the set covering problem with improved worst case bound. Information Processing Letters 48, 305–310 (1993)
5. Halldórsson, M.M.: Approximating discrete collections via local improvements. In: Proceedings of the 6th Annual ACM/SIAM Symposium on Discrete Algorithms (SODA '95), pp. 160–169 (1995)
6. Halldórsson, M.M.: Approximating k-set cover and complementary graph coloring. In: Cunningham, W.H., Queyranne, M., McCormick, S.T. (eds.) Integer Programming and Combinatorial Optimization. LNCS, vol. 1084, pp. 118–131. Springer, Heidelberg (1996)
7. Hazan, E., Safra, S., Schwartz, O.: On the complexity of approximating k-set packing. Computational Complexity 15(1), 20–39 (2006)
8. Hurkens, C.A.J., Schrijver, A.: On the size of systems of sets every t of which have an SDR, with an application to the worst-case ratio of heuristics for packing problems. SIAM Journal on Discrete Mathematics 2(1), 68–72 (1989)
9. Jain, K., Mahdian, M., Markakis, E., Saberi, A., Vazirani, V.V.: Greedy facility location algorithms analyzed using dual fitting with factor-revealing LP. Journal of the ACM 50(6), 795–824 (2003)
10. Johnson, D.S.: Approximation algorithms for combinatorial problems. Journal of Computer and System Sciences 9, 256–278 (1974)
11. Kann, V.: Maximum bounded 3-dimensional matching is MAX SNP-complete. Information Processing Letters 37, 27–35 (1991)
12. Levin, A.: Approximating the unweighted k-set cover problem: greedy meets local search. In: Erlebach, T., Kaklamanis, C. (eds.) WAOA 2006. LNCS, vol. 4368, pp. 290–310. Springer, Heidelberg (2007)
13. Lovász, L.: On the ratio of optimal integral and fractional covers. Discrete Mathematics 13, 383–390 (1975)
14. Slavík, P.: A tight analysis of the greedy algorithm for set cover. Journal of Algorithms 25, 237–254 (1997)
15. Trevisan, L.: Non-approximability results for optimization problems on bounded degree instances. In: Proceedings of the 33rd Annual ACM Symposium on Theory of Computing (STOC '01), pp. 453–461. ACM Press, New York (2001)