On k-Column Sparse Packing Programs

Nikhil Bansal[1], Nitish Korula[2], Viswanath Nagarajan[1], and Aravind Srinivasan[3]

[1] IBM T.J. Watson Research Center, Yorktown Heights, NY 10598.
{nikhil,viswanath}@us.ibm.com

[2] Dept. of Computer Science, University of Illinois, Urbana IL 61801.
Partially supported by NSF grant CCF 07-28782 and a University of Illinois Dissertation Completion Fellowship. nkorula2@illinois.edu

[3] Dept. of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742. Supported in part by NSF ITR Award CNS-0426683 and NSF Award CNS-0626636. srin@cs.umd.edu

Abstract. We consider the class of packing integer programs (PIPs) that are *column sparse*, where there is a specified upper bound k on the number of constraints that each variable appears in. We give an improved $(ek+o(k))$-approximation algorithm for k-column sparse PIPs. Our algorithm is based on a linear programming relaxation, and involves randomized rounding combined with alteration. We also show that the integrality gap of our LP relaxation is at least $2k-1$; it is known that even special cases of k-column sparse PIPs are $\Omega(\frac{k}{\log k})$-hard to approximate. We generalize our result to the case of maximizing monotone submodular functions over k-column sparse packing constraints, and obtain an $\left(\frac{e^2 k}{e-1} + o(k)\right)$-approximation algorithm. In obtaining this result, we prove a new property of submodular functions that generalizes the fractionally subadditive property, which might be of independent interest.

1 Introduction

Packing integer programs (PIPs) are those of the form:

$$\max\left\{w^T\, x \mid Sx \le c,\ x \in \{0,1\}^n\right\}, \qquad \text{where } w \in \mathbb{R}_+^n,\ c \in \mathbb{R}_+^m \text{ and } S \in \mathbb{R}_+^{m\times n}.$$

Above, n is the number of variables/columns, m is the number of rows/constraints, S is the matrix of *sizes*, c is the *capacity* vector, and w is the *weight* vector. In general, PIPs are very hard to approximate: a special case is the classic independent set problem, which is NP-Hard to approximate within a factor of $n^{1-\epsilon}$ [30], whereas an n-approximation is trivial. Thus, various special cases of PIPs are often studied. Here, we consider *k-column sparse PIPs* (denoted k-CS-PIP), which are PIPs where the number of non-zero entries in each column of matrix S is at most k. This is a fairly general class and models several basic problems such as k-set packing [19] and independent set in graphs with degree at most k.

Recently, in a somewhat surprising result, Pritchard [25] gave an algorithm for k-CS-PIP where the approximation ratio only depends on k; this is useful when k is small. This result is surprising because in contrast, no such guarantee is possible for k-row sparse PIPs. In particular, the independent set problem on general graphs is a 2-row sparse PIP, but is $n^{1-o(1)}$-hard to approximate. Pritchard's algorithm [25] had an approximation ratio of $2^k \cdot k^2$. Subsequently, an improved $O(k^2)$ approximation algorithm was obtained independently by Chekuri *et al.* [14] and Chakrabarty-Pritchard [11].

Our Results: In this paper, we first consider the k-CS-PIP problem and obtain an $(ek + o(k))$-approximation algorithm for it. Our algorithm is based on solving a strengthened version of the natural LP relaxation of k-CS-PIP, and then performing randomized rounding followed by suitable alterations. In the *randomized rounding* step, we pick each variable independently (according to its LP value) and obtain a set of variables with good expected weight; however some constraints may be violated. Then in the *alteration* step, we drop some variables so as to satisfy all constraints, while still having good expected weight. A similar approach can be used with the natural relaxation for k-CS-PIP obtained by simply dropping the integrality constraints on the variables; this gives a slightly weaker $8k$-approximation bound. However, the analysis of this weaker result is much simpler and we thus present it first. To obtain the $ek + o(k)$ bound, we construct a stronger LP relaxation by adding additional valid constraints to the natural relaxation for k-CS-PIP. The analysis of our rounding procedure is based on exploiting these additional constraints and using the positive correlation between various probabilistic events via the FKG inequality. Due to space constraints, we omit some details; these and other omitted proofs can be found in the full version of this paper [5].

Our result is almost the best possible that one can hope for using the LP based approach. We show that the integrality gap of the strengthened LP is at least $2k-1$, so our analysis is tight up to a small constant factor $e/2 \approx 1.36$ for large values of k. Even without restricting to LP based approaches, an $O(k)$ approximation is nearly best possible since it is NP-Hard to obtain an $O(k/\log k)$-approximation for the special case of k-set packing [18]. We also obtain improved results for k-CS-PIP when capacities are large relative to the sizes. In particular, we obtain a $\Theta(k^{1/\lfloor B \rfloor})$-approximation algorithm for k-CS-PIP, where $B := \min_{i\in[n],j\in[m]} c_j/s_{ij}$ measures the relative slack between the capacities c and sizes S. We also show that this result is tight up to constant factors relative to its LP relaxation.

Our second main result is for the more general problem of maximizing a monotone submodular function over packing constraints that are k-column sparse. This problem is a common generalization of maximizing a submodular function over (a) a k-dimensional knapsack [22], and (b) the intersection of k partition matroids [24]. Here, we obtain an $\left(\frac{e^2 k}{e-1} + o(k)\right)$-approximation algorithm for this problem. Our algorithm uses the continuous greedy algorithm of Vondrák [29] in conjunction with our randomized rounding plus alteration based approach. However, it turns out that the analysis of the approximation guarantee is much more intricate: In particular, we need a generalization of a result of Feige [16] that shows that submodular functions are also *fractionally subadditive*. See Section 3 for a statement of the new result, Theorem 5, and related context. This generalization is based on an interesting connection between submodular functions and the FKG inequality. We believe that this result and technique might be of further use in the study of submodular optimization.

Related Previous Work: Various special cases of k-CS-PIP have been extensively studied. An important special case is the *k-set packing* problem, where given a collection of sets of cardinality at most k, the goal is to find the maximum weight sub-collection of mutually disjoint sets. This is equivalent to k-CS-PIP where the constraint matrix S is 0-1 and the capacity c is all ones. Note that for $k = 2$ this is *maximum weight matching* which can be solved in polynomial time, and for $k = 3$ the problem becomes APX-hard [18]. After a long line of work [19, 2, 12, 9], the best-known approximation ratio for this problem is $\frac{k+1}{2} + \epsilon$ obtained using local search techniques [9]. An improved bound of $\frac{k}{2} + \epsilon$ is also known [19] for the unweighted case, i.e., the weight vector $w = \mathbf{1}$. It is also known that the natural LP relaxation for this problem has integrality gap at least $k - 1 + 1/k$, and in particular this holds for the projective plane instance of order $k - 1$. Hazan *et al.* [18] showed that k-set packing is $\Omega(\frac{k}{\log k})$-hard to approximate.

Another special case of k-CS-PIP is the independent set problem in graphs with maximum degree at most k. This is equivalent to k-CS-PIP where the constraint matrix S is 0-1, capacity c is all ones, and each row is 2-sparse. This problem has an $O(k \log\log k / \log k)$-approximation [17], and is $\Omega(k/\log^2 k)$-hard to approximate [3], assuming the Unique Games Conjecture [20].

Shepherd and Vetta [26] studied the *demand matching* problem on graphs, which is k-CS-PIP with $k = 2$, with the further restriction that in each column the non-zero entries are equal, and that no two columns have non-zero entries in the same two rows. They gave an LP-based 3.264-approximation algorithm [26], and showed that the natural LP relaxation for this problem has integrality gap at least 3. They also showed the demand matching problem to be APX-hard even on bipartite graphs. For larger values of k, problems similar to *demand matching* have been studied under the name of column-restricted PIPs [21], which arise in the context of routing flow unsplittably (see also [6, 7]). In particular, an $11.54k$-approximation algorithm was known [15] where (i) in each column all non-zero entries are equal, and (ii) the maximum entry in S is at most the minimum entry in c (this is also known as the no bottle-neck assumption); later, it was observed in [13] that even without the second of these conditions, one can obtain an $8k$ approximation. The literature on unsplittable flow is quite extensive; we refer the reader to [4, 13] and references therein.

For the general k-CS-PIP, Pritchard [25] gave a $2^k k^2$-approximation algorithm, which was the first result with approximation ratio depending only on k. Pritchard's algorithm was based on solving an iterated LP relaxation, and then applying a randomized selection procedure. Independently, [14] and [11] showed that this final step could be derandomized, yielding an improved bound of $O(k^2)$. All these previous results crucially use the structural properties of basic feasible solutions of the LP relaxation. However, as stated above, our result is based on randomized rounding with alterations and does not use properties of basic solutions. This is crucial for the submodular maximization version of the problem, as a solution to the fractional relaxation there does not have these properties.

We remark that randomized rounding with alteration has also been used earlier by Srinivasan [28] in the context of PIPs. However, the focus of this paper is different from ours; in previous work [27], Srinivasan had bounded the integrality gap for PIPs by showing a randomized algorithm that obtained a "good" solution (one that satisfies all constraints) with positive — but perhaps exponentially small — probability. In [28], he proved that rounding followed by alteration leads to an efficient and parallelizable algorithm; the rounding gives a "solution" of good value in which most constraints are satisfied, and one can alter this solution to ensure that all constraints are satisfied. (We note that [27, 28] also gave derandomized versions of these algorithms.)

Related issues have been considered in discrepancy theory, where the goal is to round a fractional solution to a k-column sparse linear program so that the capacity violation for any constraint is minimized. A celebrated result of Beck-Fiala [8] shows that the capacity violation is at most $O(k)$. A major open question in discrepancy theory is whether the above bound can be improved to $O(\sqrt{k})$, or even $O(k^{1-\epsilon})$ for some $\epsilon > 0$. While the result of [25] uses techniques similar to that of [8], a crucial difference in our problem is that no constraint can be violated at all.

There is a large body of work on constrained maximization of submodular functions; we only cite the relevant papers here. Calinescu *et al.* [10] introduced a continuous relaxation (called the *multi-linear extension* or extension-by-expectation) of submodular functions and subsequently Vondrák [29] gave an elegant $\frac{e}{e-1}$-approximation algorithm for solving this continuous relaxation over any "downward monotone" polytope $\mathcal{P}$, as long as there is a polynomial-time algorithm for optimizing linear functions over $\mathcal{P}$. We use this continuous relaxation in our algorithm for submodular maximization over k-sparse packing constraints. As noted earlier, k-sparse packing constraints generalize both k-partition matroids and k-dimensional knapsacks. Nemhauser *et al.* [24] gave a $(k+1)$-approximation for submodular maximization over the intersection of k partition matroids; when k is constant, Lee *et al.* [23] improved this to $k+\epsilon$. Kulik *et al.* [22] gave an $\left(\frac{e}{e-1}+\epsilon\right)$-approximation for submodular maximization over k-dimensional knapsacks when k is constant; if k is part of the input, the best known approximation bound is $O(k)$.

Problem Definition and Notation: Before we begin, we formally describe the k-CS-PIP problem and fix some notation. Let the items (i.e., columns) be indexed by $i \in [n]$ and the constraints (i.e., rows) be indexed by $j \in [m]$. We consider the following packing integer program.

$$\max \left\{ \sum_{i=1}^{n} w_i x_i \;\middle|\; \sum_{i=1}^{n} s_{ij} \cdot x_i \le c_j,\ \forall\, j \in [m];\ \ x_i \in \{0,1\},\ \forall\, i \in [n] \right\}$$

We say that item i *participates* in constraint j if $s_{ij} > 0$. For each $i \in [n]$, let $N(i) := \{j \in [m] \mid s_{ij} > 0\}$ be the set of constraints that i participates in. In a

k-column sparse PIP, we have $|N(i)| \leq k$ for each $i \in [n]$. The goal is to find the maximum weight subset of items such that all the constraints are satisfied.

We define the *slack* as $B := \min_{i\in[n],j\in[m]} c_j/s_{ij}$. By scaling the constraint matrix, we may assume that $c_j = 1$ for all $j \in [m]$. We also assume that $s_{ij} \leq 1$ for each i, j; otherwise, we can just fix $x_i = 0$. Finally, for each constraint j, we let $P(j)$ denote the set of items participating in this constraint. Note that $|P(j)|$ can be arbitrarily large.

Organization: In Section 2 we begin with the natural LP relaxation, and describe a simple $8k$-approximation algorithm. We then present a stronger relaxation, and sketch a proof of an $(e + o(1))k$-approximation. We also present the integrality gap of $2k - 1$ for this strengthened LP, implying that our result is almost tight. In Section 3, we describe the $O(k)$-approximation for k-column sparse packing problems over a submodular objective. Finally, in Section 4, we state the significantly better ratios that can be obtained for both linear and submodular objectives if the capacities of all constraints are large relative to the sizes; there are matching integrality gaps up to a constant factor.

2 The Algorithm for k-CS-PIP

Before presenting our algorithm, we describe a (seemingly correct) algorithm that does not quite work. Understanding why this easier algorithm fails gives useful insight into the design for the correct algorithm.

A Strawman Algorithm: Consider the following algorithm. Let x be some optimum solution to the natural LP relaxation of k-CS-PIP (i.e., dropping integrality). For each element $i \in [n]$, select it independently at random with probability $x_i/(2k)$. Let $\mathcal{S}$ be the chosen set of items. For any constraint $j \in [m]$, if it is violated, then discard all items in $\mathcal{S} \cap P(j)$, i.e., items $i \in \mathcal{S}$ for which $s_{ij} > 0$.

As the probabilities are scaled down by $2k$, by Markov's inequality any constraint j is violated with probability at most $1/(2k)$, and hence discards its items with at most this probability. By the k-sparse property, each element can be discarded by at most k constraints, and so by the union bound it is discarded with probability at most $k \cdot 1/(2k) = 1/2$. Since an element is chosen in $\mathcal{S}$ with probability $x_i/(2k)$, this implies that it lies in the overall solution with probability at least $x_i/(4k)$, implying that the proposed algorithm is a $4k$-approximation.

However, the above argument is not correct. Consider the following example. Suppose there is a single constraint (and so $k = 1$),

$$Mx_1 + x_2 + x_3 + x_4 + \ldots + x_M \leq M$$

where $M \gg 1$ is a large integer. Clearly, setting $x_i = 1/2$ for $i = 1, \ldots, M$ is a feasible solution. Now consider the execution of the strawman algorithm. Note that whenever item 1 is chosen in $\mathcal{S}$, it is very likely that some item other than 1 will also be chosen (since $M \gg 1$ and we pick each item independently with probability $x_i/(2k) = 1/4$); in this case, item 1 will be discarded. Thus the final solution will almost always *not contain* item 1, violating the claim that it lies in the final solution with probability at least $x_1/(4k) = 1/8$.

The key point is that we must consider the probability of an item being discarded by some constraint, *conditional* on it being chosen in the set $\mathcal{S}$ (for item 1 in the above example, this probability is close to one, not at most half). This is not a problem if either all item sizes are small (say $s_{ij} \leq c_j/2$), or all item sizes are large (say $s_{ij} \approx c_j$). The algorithm we analyze shows that the difficult case is indeed when some constraints contain both large and small items, as in the example above.

2.1 A Simple Algorithm for k-CS-PIP

We use the obvious LP relaxation for k-CS-PIP (i.e., dropping the integrality condition) to obtain an $8k$-approximation algorithm. An item $i \in [n]$ is called *big* for constraint $j \in [m]$ iff $s_{ij} > \frac{1}{2}$, and *small* for constraint j iff $0 < s_{ij} \leq \frac{1}{2}$. We first solve the LP relaxation to obtain an optimal fractional solution x, and then round to an integral solution as follows. With foresight, set $\alpha = 4$.

1. Sample each item $i \in [n]$ independently with probability $x_i/(\alpha k)$.
   Let $\mathcal{S}$ denote the set of chosen items. We call an item in $\mathcal{S}$ an $\mathcal{S}$-item.
2. For each item i, mark i (for deletion) if, for any constraint $j \in N(i)$, either:
   - $\mathcal{S}$ contains some *other* item $i' \in [n] \setminus \{i\}$ which is big for constraint j or
   - The sum of sizes of $\mathcal{S}$-items that are small for j exceeds 1. (i.e., the capacity).
3. Delete all marked items, and return $\mathcal{S}'$, the set of remaining items.

Analysis: We will show that this algorithm gives an $8k$-approximation.

Lemma 1. *Solution $\mathcal{S}'$ is feasible with probability one.*

Proof Sketch. Consider any fixed constraint $j \in [m]$. If there is some $i' \in \mathcal{S}'$ that is big for j, it will be the only item in $\mathcal{S}'$ that participates in constraint j. If all $\mathcal{S}'$-items participating in j are small, their total size is at most 1. □

We now prove the main technical result of this section.

Theorem 1. *For any item $i \in [n]$, the probability $\Pr[i \in \mathcal{S}' \mid i \in \mathcal{S}] \geq 1 - \frac{2}{\alpha}$. Equivalently, the probability that item i is deleted from $\mathcal{S}$ conditional on it being chosen in $\mathcal{S}$ is at most $2/\alpha$.*

Proof. For any item i and constraint $j \in N(i)$, let B_{ij} denote the event that i is marked for deletion from $\mathcal{S}$ because there is some other $\mathcal{S}$-item that is big for constraint j. Let G_j denote the event that the total size of $\mathcal{S}$-items that are small for constraint j exceeds 1. For any item $i \in [n]$ and constraint $j \in N(i)$, we will show that:

$$\Pr[B_{ij} \mid i \in \mathcal{S}] + \Pr[G_j \mid i \in \mathcal{S}] \leq \frac{2}{\alpha k} \tag{1}$$

To see that (1) implies the theorem, for any item i, simply take the union bound over all $j \in N(i)$. Thus, the probability that i is deleted from $\mathcal{S}$ conditional on it being chosen in $\mathcal{S}$ is at most $2/\alpha$. Equivalently, $\Pr[i \in \mathcal{S}' \mid i \in \mathcal{S}] \geq 1 - 2/\alpha$.

We now prove (1) using the following intuition: The total extent to which the LP selects items that are *big* for any constraint cannot be more than 2 (each big item has size at least 1/2); therefore, B_{ij} is unlikely to occur since we scaled down probabilities by factor αk. Ignoring for a moment the conditioning on $i \in \mathcal{S}$, event G_j is also unlikely, by Markov's Inequality. But items are selected for $\mathcal{S}$ independently, so if i is big for constraint j, then its presence in $\mathcal{S}$ does not affect the event G_j at all. If i is small for constraint j, then *even if* $i \in \mathcal{S}$, the total size of $\mathcal{S}$-items is unlikely to exceed 1.

To prove (1) formally, let $B(j)$ denote the set of items that are big for constraint j, and $Y_j := \sum_{\ell \in B(j)} x_\ell$. By the LP constraint for j, it follows that $Y_j \leq 2$ (since each $\ell \in B(j)$ has size $s_{\ell j} > \frac{1}{2}$). Now by a union bound,

$$\Pr[B_{ij} \mid i \in \mathcal{S}] \leq \frac{1}{\alpha k} \sum_{\ell \in B(j) \setminus \{i\}} x_\ell \leq \frac{Y_j}{\alpha k} \leq \frac{2}{\alpha k}. \tag{2}$$

Now, let $G_{-i}(j)$ denote the set of items that are small for constraint j, *not counting* item i, even if it is small. Using the LP constraint j, we have:

$$\sum_{\ell \in G_{-i}(j)} s_{\ell j} \cdot x_l \leq 1 - \sum_{\ell \in B(j)} s_{\ell j} \cdot x_\ell \leq 1 - \frac{Y_j}{2}. \tag{3}$$

Since each item i' is chosen into $\mathcal{S}$ with probability $x_{i'}/(\alpha k)$, inequality (3) implies that the expected total size of $\mathcal{S}$-items in $G_{-i}(j)$ is at most $\frac{1}{\alpha k}(1 - Y_j/2)$. By Markov's inequality, the probability that the total size of these $\mathcal{S}$-items exceeds 1/2 is at most $\frac{2}{\alpha k}(1 - Y_j/2)$. Since items are chosen independently and $i \notin G_{-i}(j)$, we obtain this probability even conditioned on $i \in \mathcal{S}$.

Whether i is big or small for j, event G_j can occur only if the total size of $\mathcal{S}$-items in $G_{-i}(j)$ exceeds 1/2. Thus,

$$\Pr[G_j \mid i \in \mathcal{S}] \leq \frac{2}{\alpha k}\left(1 - \frac{Y_j}{2}\right) = \frac{2}{\alpha k} - \frac{Y_j}{\alpha k}$$

which, combined with inequality (2), yields (1).

Using the theorem above, we obtain the desired approximation:

Theorem 2. *There is a randomized* $8k$*-approximation algorithm for* k-CS-PIP.

Proof. From Lemma 1, our algorithm always outputs a feasible solution. To bound the objective value, recall that $\Pr[i \in \mathcal{S}] = \frac{x_i}{\alpha k}$ for all $i \in [n]$. Hence Theorem 1 implies that for all $i \in [n]$

$$\Pr[i \in \mathcal{S}'] \geq \Pr[i \in \mathcal{S}] \cdot \Pr[i \in \mathcal{S}' | i \in \mathcal{S}] \geq \frac{x_i}{\alpha k} \cdot \left(1 - \frac{2}{\alpha}\right).$$

Finally, using linearity of expectation and $\alpha = 4$, we obtain the theorem.

Remarks: We note that the analysis above only uses Markov's inequality conditioned on a single item being chosen in set $\mathcal{S}$. Thus a pairwise independent distribution suffices to choose the set $\mathcal{S}$, and hence the algorithm can be easily derandomized. More generally, one could consider k-CS-PIP with *arbitrary upper-bounds* on the variables: the above $8k$-approximation algorithm extends easily to this setting (details in the full version).

2.2 A Stronger LP, and Improved Approximation

We now present our strengthened LP and the $(ek + o(k))$-approximation algorithm for k-CS-PIP.

Stronger LP relaxation. Recall that entries are scaled so that all capacities are one. An item i is called *big* for constraint j iff $s_{ij} > 1/2$. For each constraint $j \in [m]$, let $B(j) = \{i \in [n] \mid s_{ij} > \frac{1}{2}\}$ denote the set of big items. Since no two items that are big for some constraint can be chosen in an integral solution, the inequality $\sum_{i \in B(j)} x_i \leq 1$ is valid for each $j \in [m]$. The strengthened LP relaxation that we consider is as follows.

$$\max \quad \sum_{i=1}^{n} w_i x_i \tag{4}$$

$$\text{s.t.} \quad \sum_{i=1}^{n} s_{ij} \cdot x_i \leq c_j, \qquad \forall j \in [m] \tag{5}$$

$$\sum_{i \in B(j)} x_i \leq 1, \qquad \forall j \in [m]. \tag{6}$$

$$0 \leq x_i \leq 1, \qquad \forall i \in [n]. \tag{7}$$

The Algorithm: The algorithm obtains an optimal solution x to the LP relaxation (4-7), and rounds it to an integral solution $\mathcal{S}'$ as follows (parameter α will be set to 1 later).

1. Pick each item $i \in [n]$ independently with probability $x_i/(\alpha k)$, with $\alpha \geq 1$. Let $\mathcal{S}$ denote the set of chosen items.
2. For any item i and constraint $j \in N(i)$, let E_{ij} denote the event that the items $\{i' \in \mathcal{S} \mid s_{i'j} \geq s_{ij}\}$ have total size (in constraint j) exceeding one. Mark i for deletion if E_{ij} occurs for any $j \in N(i)$.
3. Return set $\mathcal{S}' \subseteq \mathcal{S}$ consisting of all items $i \in \mathcal{S}$ not marked for deletion.

Note the rule for deleting an item from $\mathcal{S}$. In particular, whether item i is deleted due to constraint j only depends on items that are at least as large as i in j.

Analysis: It is clear that $\mathcal{S}'$ is feasible with probability one. The improved approximation ratio comes from *four* different steps: First, we use the stronger LP relaxation. Second, the more careful alteration step does not discard items unnecessarily; the previous algorithm sometimes deleted items from $\mathcal{S}$ even when constraints were not violated. Third, in analyzing the probability that constraint

j causes item i to be deleted from $\mathcal{S}$, we further exploit discreteness of item sizes. And fourth, for each item i, we use the FKG inequality to bound the probability it is deleted instead of the weaker union bound over all constraints in $N(i)$.

The main lemma is the following, where we show that each item appears in $\mathcal{S}'$ with good probability.

Lemma 2. *For every item* $i \in [n]$ *and constraint* $j \in N(i)$*, we have* $\Pr[E_{ij} \mid i \in \mathcal{S}] \leq \frac{1}{\alpha k}\left(1 + (\frac{2}{\alpha k})^{1/3}\right)$.

Proof Sketch. Let $\ell := (4\alpha k)^{1/3}$. We classify items in relation to constraints as:

- Item $i \in [n]$ is *big* for constraint $j \in [m]$ if $s_{ij} > \frac{1}{2}$.
- Item $i \in [n]$ is *medium* for constraint $j \in [m]$ if $\frac{1}{\ell} \leq s_{ij} \leq \frac{1}{2}$.
- Item $i \in [n]$ is *tiny* for constraint $j \in [m]$ if $s_{ij} < \frac{1}{\ell}$.

We separately bound $\Pr[E_{ij} | i \in \mathcal{S}]$ when item i is big, medium, and tiny.

Claim. For any $i \in [n]$ and $j \in [m]$:

1. If item i is big for constraint j, $\Pr[E_{ij} \mid i \in \mathcal{S}] \leq \frac{1}{\alpha k}$.
2. If item i is medium for constraint j, $\Pr[E_{ij} \mid i \in \mathcal{S}] \leq \frac{1}{\alpha k}\left(1 + \frac{\ell^2}{2\alpha k}\right)$.
3. If item i is tiny for constraint j, $\Pr[E_{ij} \mid i \in \mathcal{S}] \leq \frac{1}{\alpha k}\left(1 + \frac{2}{\ell}\right)$.

In case 1, E_{ij} occurs only if some other big item for constraint j is chosen in $\mathcal{S}$; the new constraints (6) of the strengthened LP bound this probability. In case 2, E_{ij} can occur only if some big item or at least two medium items other than i are selected for $\mathcal{S}$; we argue that the latter probability is much smaller than $1/\alpha k$. In case 3, E_{ij} can occur only if the total size (in constraint j) of items in $\mathcal{S} \setminus \{i\}$ is greater than $1 - \frac{1}{\ell}$; Markov's inequality gives the desired result.

Thus, for any item i and constraint $j \in N(i)$, $\Pr[E_{ij} \mid i \in \mathcal{S}] \leq \frac{1}{\alpha k} \max\{(1 + \frac{2}{\ell}), (1 + \frac{\ell^2}{2\alpha k})\}$. From the choice of $\ell = (4\alpha k)^{1/3}$, which makes the probability in parts 2 and 3 of the claim equal, we obtain the lemma. □

We now prove the main result of this section.

Theorem 3. *For each* $i \in [n]$, $\Pr[i \in \mathcal{S}' \mid i \in \mathcal{S}] \geq \left(1 - \frac{1}{\alpha k}\left(1 + (\frac{2}{\alpha k})^{1/3}\right)\right)^k$.

Proof. For any item i and constraint $j \in N(i)$, the conditional event $(\neg E_{ij} \mid i \in \mathcal{S})$ is a *decreasing* function over the choice of items in set $[n] \setminus \{i\}$. Thus, by the FKG inequality [1], for any fixed item $i \in [n]$, the probability that no event $(E_{ij} \mid i \in \mathcal{S})$ occurs is:

$$\Pr\left[\bigwedge_{j \in N(i)} \neg E_{ij} \;\middle|\; i \in \mathcal{S}\right] \geq \prod_{j \in N(i)} \Pr[\neg E_{ij} \mid i \in \mathcal{S}]$$

From Lemma 2, $\Pr[\neg E_{ij} \mid i \in \mathcal{S}] \geq 1 - \frac{1}{\alpha k}\left(1 + (\frac{2}{\alpha k})^{1/3}\right)$. As each item is in at most k constraints, we obtain the theorem.

Now, by setting $\alpha = 1$,[4] we have $\Pr[i \in \mathcal{S}] = 1/k$, and $\Pr[i \in \mathcal{S}' \mid i \in \mathcal{S}] \geq \frac{1}{e+o(1)}$, which immediately implies:

Theorem 4. *There is a randomized* $(ek + o(k))$*-approximation algorithm for* k-CS-PIP.

Remark: We note that this algorithm can be derandomized using conditional expectation and pessimistic estimators, since we can exactly compute estimates of the relevant probabilities. Also, using ideas from [28], the algorithm can be implemented in RNC. We defer details to the full version.

Integrality Gap of LP (4-7). Consider the instance on $n = m = 2k-1$ items and constraints defined as follows. We view the indices $[n] = \{0, 1, \cdots, n-1\}$ as integers modulo n. The weights $w_i = 1$ for all $i \in [n]$. The sizes are:

$$s_{ij} := \begin{cases} 1 & \text{if } i = j \\ \epsilon & \text{if } j \in \{i+1, \cdots, i+k-1 \pmod{n}\} \\ 0 & \text{otherwise} \end{cases}, \qquad \forall i, j \in [n].$$

where $\epsilon > 0$ is arbitrarily small, in particular $\epsilon \ll \frac{1}{nk}$.

Observe that setting $x_i = 1 - k\epsilon$ for all $i \in [n]$ is a feasible fractional solution to the strengthened LP (4-7); each constraint has only one big item and so the new constraint (6) is satisfied. Thus the optimal LP value is at least $(1-k\epsilon)\cdot n \approx n = 2k-1$. On the other hand, it is easy to see that the optimal integral solution can only choose one item and hence has value 1. Thus the integrality gap of the LP we consider is at least $2k-1$, for every $k \geq 1$.

3 Submodular Objective Functions

We now consider the more general case when the objective we seek to maximize is an arbitrary non-negative *monotone submodular function* $f : 2^{[n]} \to \mathbb{R}_+$. The problem we consider is:

$$\max \left\{ f(T) \mid \sum_{i \in T} s_{ij} \leq c_j, \ \forall j \in [m]; \ T \subseteq [n] \right\} \tag{8}$$

As is standard when dealing with submodular functions, we only assume *value-oracle* access to the function: i.e., the algorithm can query any subset $T \subseteq [n]$, and it obtains the function value $f(T)$ in constant time. Again, we let k denote the column-sparseness of the underlying constraint matrix. In this section we obtain an $O(k)$-approximation algorithm for Problem (8). The algorithm is similar to that for k-CS-PIP (where the objective was linear):

1. We first solve (approximately) a suitable continuous relaxation of (8). This step follows directly from the algorithm of Vondrák [29].

[4] Note that this is optimal only asymptotically; in the case of $k = 2$, for instance, it is better to choose $\alpha \approx 2.8$.

2. Then, using the fractional solution, we perform the randomized rounding with alteration described in Section 2. Although the algorithm is the same as for linear functions, the analysis requires considerably more work. In the process, we also establish a new property of submodular functions that generalizes *fractional subadditivity* [16].

Solving the Continuous Relaxation. The *extension-by-expectation* (also called the *multi-linear* extension) of a submodular function f is a continuous function $F : [0,1]^n \to \mathbb{R}_+$ defined as follows:

$$F(x) := \sum_{T \subseteq [n]} \Pi_{i \in T}\ x_i \cdot \Pi_{j \notin T}\ (1 - x_j) \cdot f(T)$$

Note that $F(x) = f(x)$ for $x \in \{0,1\}^n$ and hence F is an extension of f. Even though F is a non-linear function, using the continuous greedy algorithm of Vondrák [29], we can obtain an $\left(1 - \frac{1}{e}\right)$-approximation algorithm to the following *fractional relaxation* of (8):

$$\max \left\{ F(x) \mid \sum_{i=1}^{n} s_{ij} \cdot x_i \le c_j,\ \forall j \in [m];\ 0 \le x_i \le 1,\ \forall i \in [n] \right\} \tag{9}$$

In order to apply the algorithm from [29], one needs to solve in polynomial time the problem of maximizing a *linear* objective over the constraints $\{\sum_{i=1}^{n} s_{ij} \cdot x_i \le c_j,\ \forall j \in [m];\ 0 \le x_i \le 1,\ \forall i \in [n]\}$. This is indeed possible since it is a linear program on n variables and m constraints.

The Rounding Algorithm and Analysis. The rounding algorithm is identical to that for k-CS-PIP. Let x denote any feasible solution to Problem (9). We apply the rounding algorithm from the previous section, to first obtain (possibly infeasible) solution $\mathcal{S} \subseteq [n]$ and then *feasible integral solution* $\mathcal{S}' \subseteq [n]$.

However, the analysis approach in Theorem 3 does not work. The problem is that even though S (which is chosen by random sampling) has good expected profit, i.e., $E[f(\mathcal{S})] = \Omega(\frac{1}{k})F(x)$, it may happen that the alteration step used to obtain $\mathcal{S}'$ from $\mathcal{S}$ may end up throwing away essentially all the profit. This was not an issue for linear objective functions since our alteration procedure guarantees that $\Pr[i \in \mathcal{S}' | i \in \mathcal{S}] = \Omega(1)$ for each $i \in [n]$; if f is linear, this implies $E[f(\mathcal{S})] = \Omega(1)\, E[f(\mathcal{S}')]$. However, this property is not enough for general monotone submodular functions. Consider the following:

Example: Let set $\mathcal{S} \subseteq [n]$ be drawn from the following distribution:

- With probability $1/2n$, $\mathcal{S} = [n]$.
- For each $i \in [n]$, $\mathcal{S} = \{i\}$ with probability $1/2n$.
- With probability $1/2 - 1/2n$, $\mathcal{S} = \emptyset$.

Define $\mathcal{S}' = \mathcal{S}$ if $\mathcal{S} = [n]$, and $\mathcal{S}' = \emptyset$ otherwise. For each $i \in [n]$, we have $\Pr[i \in \mathcal{S}' \mid i \in \mathcal{S}] = 1/2 = \Omega(1)$. However, consider the profit with respect to the "coverage" submodular function f, where $f(T) = 1$ if $T \neq \emptyset$ and $= 0$ otherwise. We have $E[f(\mathcal{S})] = 1/2 + 1/2n$, but $E[f(\mathcal{S}')]$ is only $1/2n \ll E[f(\mathcal{S})]$.

Remark: Note that if $\mathcal{S}'$ itself was chosen randomly from $\mathcal{S}$ such that $\Pr[i \in \mathcal{S}' | \mathcal{S} = T] = \Omega(1)$ for *every* $T \subseteq [n]$ *and* $i \in T$, then we would be done by Feige's

Subadditivity Lemma [16]. Unfortunately, this is too much to hope for. In our rounding procedure, for any particular choice of $\mathcal{S}$, set $\mathcal{S}'$ is a fixed subset of $\mathcal{S}$; and there could be (bad) sets $\mathcal{S}$, where after the alteration step we end up with sets $\mathcal{S}'$ such that $|\mathcal{S}'| \ll |\mathcal{S}|$.

However, it turns out that we can use the following two *additional* properties of our algorithm to argue that $\mathcal{S}'$ has reasonable profit. First, the sets $\mathcal{S}$ we construct are drawn from a product distribution on the items. Second, our alteration procedure has the following 'monotonicity' property: Suppose $i \in T_1 \subseteq T_2 \subseteq [n]$, and $i \in \mathcal{S}'$ when $\mathcal{S} = T_2$. Then we are guaranteed that $i \in \mathcal{S}'$ when $\mathcal{S} = T_1$. (That is, if $\mathcal{S}$ contains additional items, it is more likely that i will be discarded by some constraint it participates in.) The example above does not satisfy either of these properties. Corollary 1 shows that these properties suffice. Roughly speaking, the intuition is that since f is submodular, the marginal contribution of item i to $\mathcal{S}$ is largest when $\mathcal{S}$ is "small"; this is also the case when i is most likely to be retained for $\mathcal{S}'$. That is, for every $i \in [n]$, both $\Pr[i \in \mathcal{S}' \mid i \in \mathcal{S}]$ and the marginal contribution of i to $f(\mathcal{S})$ are *decreasing* functions of $\mathcal{S}$. We prove (see [5]) the following generalization of Feige's Subadditivity Lemma.

Theorem 5. *Let $[n]$ denote a groundset, $x \in [0,1]^n$, and for each $B \subseteq [n]$ define $p(B) = \Pi_{i\in B} x_i \cdot \Pi_{j \notin B}(1 - x_j)$. Associated with each $B \subseteq [n]$, there is an arbitrary distribution over subsets of B, where each set $A \subseteq B$ has probability $q_B(A)$; so $\sum_{A\subseteq B} q_B(A) = 1$ for all $B \subseteq [n]$. That is, we choose B from a product distribution, and then retain a subset A of B by applying a randomized alteration. Suppose that the system satisfies the following conditions.*

Marginal Property:

$$\forall i \in [n], \quad \sum_{B\subseteq[n]} p(B) \sum_{A\subseteq B: i\in A} q_B(A) \;\geq\; \beta \cdot \sum_{B\subseteq[n]: i\in B} p(B). \tag{10}$$

Monotonicity: *For any two subsets $B \subseteq B' \subseteq [n]$ we have,*

$$\forall i \in B, \quad \sum_{A\subseteq B: i\in A} q_B(A) \;\geq\; \sum_{A'\subseteq B': i\in A'} q_{B'}(A') \tag{11}$$

Then, for any monotone submodular function f,

$$\sum_{B\subseteq[n]} p(B) \sum_{A\subseteq B} q_B(A) \cdot f(A) \;\geq\; \beta \cdot \sum_{B\subseteq[n]} p(B) \cdot f(B). \tag{12}$$

Corollary 1. *Let $\mathcal{S}$ be a random set drawn from a product distribution on $[n]$. Let $\mathcal{S}'$ be another random set where for each choice of $\mathcal{S}$, set $\mathcal{S}'$ is an arbitrary subset of $\mathcal{S}$. Suppose that for each $i \in [n]$ the following hold.*

- $\Pr_{\mathcal{S}}[i \in \mathcal{S}' \mid i \in \mathcal{S}] \geq \beta$, *and*
- *For all $T_1 \subseteq T_2$ with $T_1 \ni i$, if $i \in \mathcal{S}'$ when $\mathcal{S} = T_2$ then $i \in \mathcal{S}'$ when $\mathcal{S} = T_1$.*

Then $E[f(\mathcal{S}')] \geq \beta E[f(\mathcal{S})]$.

We are now ready to prove the performance guarantee of our algorithm. Observe that our rounding algorithm satisfies the hypothesis of Corollary 1 with $\beta = \frac{1}{e+o(1)}$, when parameter $\alpha = 1$. Moreover, one can show that $E[f(\mathcal{S})] \geq F(x)/(\alpha k)$. Thus, $E[f(\mathcal{S}')] \geq \frac{1}{e+o(1)} E[f(\mathcal{S})] \geq \frac{1}{ek+o(k)} \cdot F(x)$. Combined with the fact that x is an $\frac{e}{e-1}$-approximate solution to the continuous relaxation (9), we have proved our main result:

Theorem 6. *There is a randomized algorithm for maximizing any monotone submodular function over k-column sparse packing constraints achieving approximation ratio $\frac{e^2}{e-1}k + o(k)$.*

4 k-CS-PIP Algorithm for large B

We can obtain substantially better approximation guarantees for k-CS-PIP when the capacities are large relative to the sizes. Recall the definition of the slack parameter B. We consider the k-CS-PIP problem as a function of both k and B, and obtain improved approximation ratios given in the following.

Theorem 7. *There is a $\left(4e \cdot \left((e + o(1)) \lfloor B \rfloor k\right)^{1/\lfloor B \rfloor}\right)$-approximation algorithm for k-CS-PIP, and a $\left(\frac{4e^2}{e-1} \cdot \left((e + o(1)) \lfloor B \rfloor k\right)^{1/\lfloor B \rfloor}\right)$-approximation for maximizing monotone submodular functions over k-column sparse packing constraints.*

The algorithms that obtain these approximation ratios are similar to those of the preceding sections, but additional care is required in the analysis; as B is large, one can now use a smaller scaling factor in the randomized rounding step while bounding the probability that an element is deleted in the alteration step. We also show that the natural LP relaxation for k-CS-PIP has an $\Omega(k^{1/\lfloor B \rfloor})$ integrality gap for every $B \geq 1$.

Acknowledgements: NK thanks Chandra Chekuri and Alina Ene for detailed discussions on k-CS-PIP. We also thank Deeparnab Chakarabarty and David Pritchard for discussions and sharing a copy of [11]. We thank Jan Vondrák and Chandra Chekuri for pointing out an error in the original proof of Theorem 6, which prompted us to prove Theorem 5. Our thanks also to the IPCO referees for their helpful suggestions.

References

1. N. Alon and J. Spencer, The Probabilistic Method. 3rd ed. Wiley-Interscience, New York, 2008.
2. E. M. Arkin and R. Hassin, On Local Search for Weighted k-Set Packing, in *European Symposium on Algorithms,* 1997, 13-22.
3. P. Austrin, S. Khot and S. Safra, Inapproximability of Vertex Cover and Independent Set in Bounded Degree Graphs. In *Comp. Complexity Conference*, 2009.
4. N. Bansal, Z. Friggstad, R. Khandekar and M. R. Salavatipour, A logarithmic approximation for unsplittable flow on line graphs. In *SODA*, 2009.

5. N. Bansal, N. Korula, V. Nagarajan and A. Srinivasan, On k-Column Sparse Packing Programs (full version), *arXiv*, 2010.
6. A. Baveja and A. Srinivasan, Approximating Low-Congestion Routing and Column-Restricted Packing Problems. *Information Proc. Letters* (74), 19–25, 2000.
7. A. Baveja and A. Srinivasan, Approximation Algorithms for Disjoint Paths and Related Routing and Packing Problems. *Math. of Oper. Res.* (25), 255–280, 2000.
8. J. Beck and T. Fiala, "Integer making" theorems. In *Discrete Appl. Math.*, 3, 1–8, 1981.
9. Piotr Berman, A $d/2$ approximation for maximum weight independent set in d-claw free graphs. In *Nordic Journal of Computing*, 7(3), 178-184, 2000.
10. G. Calinescu, C. Chekuri, M. Pál and J. Vondrák. Maximizing a monotone submodular function under a matroid constraint, In *IPCO*, 2007.
11. D. Chakrabarty and D. Pritchard, Personal Communication, 2009.
12. B. Chandra and M. Halldórsson, Greedy Local Improvement and Weighted Packing Approximation. In *SODA*, 1999.
13. C. Chekuri, A. Ene and N. Korula. Unsplittable Flow in Paths and Trees and Column-Restricted Packing Integer Programs. In *APPROX*, 2009.
14. C. Chekuri, A. Ene and N. Korula, Personal Communication, 2009.
15. C. Chekuri, M. Mydlarz and B. Shepherd, Multicommodity Demand Flow in a Tree and Packing Integer Programs. In *ACM Trans. on Algorithms*, 3(3), 2007.
16. U. Feige, On maximizing welfare when utility functions are subadditive. In *STOC*, 2006, 41-50.
17. E. Halperin, Improved Approximation Algorithms for the Vertex Cover Problem in Graphs and Hypergraphs. In *SIAM J. Comput.*, 31(5), 1608-1623, 2002.
18. E. Hazan, S. Safra and O. Schwartz, On the complexity of approximating k-set packing. In *Computational Complexity*, 15(1), 20-39, 2003.
19. A. J. Hurkens, A. Schrijver, 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. In *SIAM J. Discrete Math.*, 2(1), 68-72 1989.
20. S. Khot, On the power of unique 2-prover 1-round games, In *STOC*, 2002, 767–775.
21. S. Kolliopoulos and C. Stein, Approximating Disjoint-Path Problems using Packing Integer Programs. *Mathematical Programming A* (99), 63–87, 2004.
22. A. Kulik, H. Shachnai and T. Tamir. Maximizing submodular functions subject to multiple linear constraints. In *SODA*, 2009.
23. J. Lee, V. Mirrokni, V. Nagarajan and M. Sviridenko. Non-monotone submodular maximization under matroid and knapsack constraints, In *STOC*, 2009, 323–332.
24. G. L. Nemhauser, L. A. Wolsey and M. L. Fisher. An analysis of approximations for maximizing submodular set functions II. *Math. Prog. Study* 8 (1978), 73–87.
25. D. Pritchard, Approximability of Sparse Integer Programs, In *ESA*, 2009.
26. B. Shepherd and A. Vetta, The demand matching problem, *Mathematics of Operations Research*, 32, 563-578, 2007.
27. A. Srinivasan, Improved Approximation Guarantees for Packing and Covering Integer Programs, *SIAM J. Comput.*, 29(2), 1999, 648-670.
28. A. Srinivasan, New approaches to covering and packing problems, In *SODA*, 2001, 567–576.
29. J. Vondrák, Optimal approximation for the submodular welfare problem in the value oracle model. In *STOC*, 2008, 67-74.
30. David Zuckerman, Linear Degree Extractors and the Inapproximability of Max Clique and Chromatic Number, *Theory of Computing*, 2007, 103-128, 3(1).