Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Journal of Computer and System Sciences 71 (2005) 495–505
JOURNAL OF COMPUTER AND SYSTEM SCIENCES
www.elsevier.com/locate/jcss

Approximation algorithms for covering/packing integer programs

Stavros G. Kolliopoulos[a], Neal E. Young[b,*]

[a] *Department of Informatics and Telecommunications, National and Kapodistrain University of Athens, Panepistimiopolis Ilissia, Athens 157 84, Greece*
[b] *Department of Computer Science and Engineering, University of California, Riverside, CA 92521, USA*

Received 18 January 2005; received in revised form 19 January 2005

Available online 18 August 2005

Abstract

Given matrices A and B and vectors a, b, c and d, all with non-negative entries, we consider the problem of computing $\min\{c^T x : x \in \mathbb{Z}_+^n,\ Ax \geqslant a,\ Bx \leqslant b,\ x \leqslant d\}$. We give a bicriteria-approximation algorithm that, given $\varepsilon \in (0, 1]$, finds a solution of cost $O(\ln(m)/\varepsilon^2)$ times optimal, meeting the covering constraints ($Ax \geqslant a$) and multiplicity constraints ($x \leqslant d$), and satisfying $Bx \leqslant (1+\varepsilon)b + \beta$, where β is the vector of row sums $\beta_i = \sum_j B_{ij}$. Here m denotes the number of rows of A.

This gives an $O(\ln m)$-approximation algorithm for CIP—minimum-cost covering integer programs with multiplicity constraints, i.e., the special case when there are no packing constraints $Bx \leqslant b$. The previous best approximation ratio has been $O(\ln(\max_j \sum_i A_{ij}))$ since 1982. CIP contains the set cover problem as a special case, so $O(\ln m)$-approximation is the best possible unless P = NP.

Keywords: Covering/packing integer programs; Set cover; Approximation algorithms; Multiplicity constraints

1. Introduction

We consider integer covering/packing programs of the following form:

Given $\mathcal{P} = (A, B, a, b, c, d)$ *with* $A \in \mathbb{R}_+^{m\times n}$, $B \in \mathbb{R}_+^{r\times n}$, $a \in \mathbb{R}_+^m$, $b \in \mathbb{R}_+^r$, *and* $c, d \in \mathbb{R}_+^n$, *compute* $\text{OPT} = \min\{c^T x : x \in \mathbb{Z}_+^n,\ Ax \geqslant a,\ Bx \leqslant b,\ x \leqslant d\}$.

* Corresponding author.
E-mail address: neal@cs.ucr.edu (N.E. Young)
URL: http://www.di.uoa.gr/∼sgk.

0022-0000/$ - see front matter © 2005 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcss.2005.05.002

The constraints $Ax \geqslant a$, $Bx \leqslant b$, and $x \leqslant d$ are called, respectively, *covering*, *packing*, and *multiplicity* constraints.

The *width*, W, is $\min\{a_i/A_{ij} : A_{ij} > 0\}$. Note that it is easy to reduce any instance to an equivalent instance with width W at least 1—simply change each A_{ij} to $\min\{A_{ij}, a_i\}$. This does not change the set of integer solutions.

The *dilation*, α, is the maximum number of covering constraints that any variable appears in.

A ρ-*approximate* solution is a solution meeting all constraints and having cost at most ρ times the optimum. A ρ-*approximation algorithm* is a polynomial-time algorithm that produces only ρ-approximate solutions. The quantity ρ is called the *approximation ratio* of the algorithm.

Perhaps the most well-known problem of the form above is set cover: given a collection of sets with costs, choose a minimum-cost collection of sets such that every element is in a chosen set. In the corresponding formulation $A_{ij} \in \{0, 1\}$, and $a_i = 1$, for $i = 1, \dots, m$, $j = 1, \dots, n$. This problem admits a simple $(1 + \ln m)$-approximation algorithm [3,9,12], and no $o(\ln m)$-approximation is possible in polynomial time, unless P = NP [19].

Other special cases include natural generalizations of set cover, including *set multicover* where $a_i \in \mathbb{Z}_+$ and *multiset multicover* where in addition $A_{ij} \in \mathbb{Z}_+$ [24]. In these problems, multiplicity constraints limit the number of times a given set or multiset can be chosen. In facility-location problems (where x_j represents the number of facilities opened at a site j), multiplicity constraints are used to limit the number of facilities opened at a site. The motivation may be capacity limits, security goals, or fault-tolerance (to ensure that when a site is breached or damaged, only a limited number of opened facilities should be affected) [14,23].

We give bicriteria approximation algorithms. For any $\varepsilon \in (0, 1]$, our first algorithm finds a solution $\hat{x}$ such that $A\hat{x} \geqslant a$, $B\hat{x} \leqslant (1+\varepsilon)b+\beta$, $\hat{x} \leqslant \lceil (1+\varepsilon)d \rceil$, where β is the vector of sums of rows of B: $\beta_i = \sum_j B_{ij}$. The cost $c^T\hat{x}$ is $O(1 + \ln(1 + \alpha)/(W\varepsilon^2))$ times the optimum of the standard linear programming (LP) relaxation. Note that the standard LP relaxation has an arbitrarily large integrality gap if multiplicity constraints are to be respected. Our second algorithm finds a solution $\hat{x}$ of cost $O(1+\ln(1+\alpha)/\varepsilon^2)$ times the optimum, satisfying $A\hat{x} \geqslant a$, $B\hat{x} \leqslant (1 + \varepsilon)b + \beta$, $\hat{x} \leqslant d$, thus meeting the multiplicity constraints.

These algorithms are appropriate for the case when B has small row sums (for example, a multiset multicover problem with restrictions such as "from the 5 sets $s_1, s_2, \dots, s_5$, only 100 copies can be chosen'') and for the *CIP* (covering integer programming) problem, formed by instances without packing constraints (no "$Bx \leqslant b$''). CIP is well-studied in its own right. For this problem, our second algorithm is an $O(\ln(1 + \alpha))$-approximation algorithm. This is the first approximation algorithm for CIP whose approximation ratio is logarithmic in the input size. Fig. 1 has a table of known approximation algorithms for CIP. [1]

We use here results for another special case—CIP without multiplicity constraints. This problem, which we denote CIP_∞, has a long line of research, but we use only the following results. Randomized rounding easily yields an $O(1 + \ln(m)/W + \sqrt{\ln(m)/W})$-approximation algorithm, where W, called the *width* of the problem instance, is $\max\{a_i/A_{ij} : A_{ij} > 0\}$. Srinivasan gives an $O(1 + \ln(1 + \alpha)/W + \sqrt{\ln(1 + \alpha)/W})$-approximation algorithm, where α, called the *dilation* of the instance, is the maximum

[1] In the table, $H(t)$ is the harmonic series with t terms. It is well-known that $H(t) = \ln t + \Theta(1)$. To give some intuition for the Fisher–Wolsey bound consider for example the case where each $c_j = 1$ and the minimum non-zero entry of A is 1. In this case the bound is asymptotically equal to Dobson's.

Who	Restriction on CIP	Cost approximation ratio	Multiplicity guarantee
Fisher Wolsey [6]	None	$1+\ln(\beta_1/\beta_2)$ $\beta_1 = \max_j \sum_i A_{ij}/c_j$ $\beta_2 = \min\{\frac{A_{ij}}{c_j} \mid A_{ij} > 0\}$	$x \le d$
Dobson [4]	$A_{ij} \in \mathbb{Z}_+$	$H(\max_{j=1}^n \sum_{i=1}^m A_{ij})$	$x \le d$
Rajagopalan Vazirani [18]	$A_{ij} \in \{0,1\}$	$O(\ln(1+\alpha))$	$x \le d$
Srinivasan Teo [23]	$c_j = 1$	$O(1+\ln(m)/(W\varepsilon^2))$	$x \le \lceil(1+\varepsilon)d\rceil$
Kolliopoulos [10]	$A_{ij} \in \{0,\phi_j\}$ for some ϕ_j	$O(\ln(1+\alpha))$	$x \le \lceil 12d \rceil$
Srinivasan [21,22]	None	$O(1+\ln(1+\alpha)/W)$	$x \le O(1+\ln(1+\alpha)/W)d$
This paper	None	$O(1+\ln(1+\alpha)/(W\varepsilon^2))$	$x \le \lceil(1+\varepsilon)d\rceil$
This paper	None	$O(\ln(1+\alpha)/\varepsilon^2)$	$x \le d$

Fig. 1. Approximation algorithms for the CIP problem, $\min\{c^T x : x \in \mathbb{Z}_+^n, Ax \geqslant a, x \leqslant d\}$. The width W is $\min\{a_i/A_{ij} : A_{ij} > 0\}$. Without loss of generality, $W \geqslant 1$. The dilation α is the maximum number of constraints any variable appears in. The algorithms presented in this paper generalize to allow packing constraints ($Bx \leqslant b$); for the general case the approximate solution $\hat{x}$ satisfies $B\hat{x} \leqslant (1+\varepsilon)b + \beta$ where $\beta_i = \sum_j B_{ij}$.

number of constraints that any variable occurs in [21,22]. Neither of these algorithms return solutions that are suitable for CIP, as the solutions can violate the multiplicity constraints by a large factor.

A preliminary version of this paper appeared in [11]. Other work on covering problems includes [4,6,15,18,21,22,26]. See [8] for a survey.

The outline of this paper is as follows. In Section 2 we present our first main algorithm that violates the multiplicity constraints by a $(1+\varepsilon)$ factor. In Section 3 we discuss the integrality gap of the standard LP formulation and present our second main algorithm which meets the multiplicity constraints. We conclude in Section 4 with some open questions.

2. Rounding LP relaxations of CIP$_\infty$ and CIP

The approximation ratios in this paper are proven with respect to various linear programming relaxations of the problems. Our first main result follows from careful consideration of the relation between various forms of the problem and their standard relaxations.

We begin by describing a standard approximation algorithm for CIP$_\infty$. Given an instance $\mathcal{P} = (A, a, c)$ of CIP$_\infty$, the standard linear programming (LP) relaxation is $\text{FOPT}_\infty = \min\{c^T x : x \in \mathbb{R}_+^n, Ax \geqslant a\}$. We call feasible solutions to this LP *fractional* solutions to $\mathcal{P}$. In contrast, we call actual solutions to $\mathcal{P}$ *integer* solutions.

The value FOPT_∞ can be computed in polynomial time (using linear programming) and is a lower bound on the optimum value OPT. The algorithm computes an optimal solution $\bar{x}$ (of cost FOPT_∞) to the fractional relaxation, then rounds $\bar{x}$ to an integer solution $\hat{x}$ using the following randomized rounding scheme:

Lemma 1 (*folklore*). *Given a* CIP_∞ *instance* $\mathcal{P} = (A, a, c)$ *and fractional solution* $\bar{x}$, *let* $L = 1 + \max\{4\ln(2m)/W, \sqrt{4\ln(2m)/W}\}$. *With positive probability, the following rounding scheme produces an integer solution* $\hat{x}$ *of cost at most* $2L$ *times the cost of* $\bar{x}$:

1. *Let* $x' = L\bar{x}$.
2. *Randomly round* x' *to* $\hat{x}$:
   let $\hat{x}_j = \lceil x'_j \rceil$ *with probability* $x'_j - \lfloor x'_j \rfloor$, *and* $\hat{x}_j = \lfloor x'_j \rfloor$ *otherwise.*

The proof is standard and we postpone it until the appendix. In what follows, the floor (ceiling) of a vector t denotes the vector where the ith coordinate is the floor (ceiling) of t_i.

Corollary 2. *Given a* CIP_∞ *instance* $\mathcal{P} = (A, a, c)$ *and fractional solution* $\bar{x}$, *let* $L = 1 + \max\{4\ln(2m)/W, \sqrt{4\ln(2m)/W}\}$. *One can compute in polynomial time an integer solution* $\hat{x} \leqslant \lceil L\bar{x} \rceil$ *of cost at most* $2L$ *times the cost of* $\bar{x}$.

The corollary follows because the rounding scheme can be derandomized using the method of conditional probabilities [5,16,20]. The rounding scheme above has been improved by Srinivasan, who shows the following:

Theorem 3 (*Srinivasan [21]*). *Given a* CIP_∞ *instance* $\mathcal{P} = (A, a, c)$ *and fractional solution* $\bar{x}$, *let* α *be the maximum number of constraints in which any variable appears. For some* $L = 1 + O(\ln(1+\alpha)/W + \sqrt{\ln(1+\alpha)/W})$, *one can compute in polynomial time an integer solution* $\hat{x} \leqslant \lceil L\bar{x} \rceil$ *of cost* $O(L)$ *times the cost of* $\bar{x}$.

Since the optimal fractional solution $\bar{x}$ can be computed in polynomial time, Srinivasan immediately obtains an $O(L)$-approximation algorithm for CIP_∞.

2.1. Extending to CIP using $1/K$*-granularity*

A natural idea would be to extend the rounding schemes above for CIP_∞ to handle CIP problems too. Of course, to do this, we need to figure out how to handle the multiplicity constraints. The natural LP relaxation of CIP is

$$\text{FOPT} = \min\{c^T x : x \in \mathbb{R}^n_+, Ax \geqslant a, x \leqslant d\}.$$

The first idea would be to compute the optimal fractional solution $\bar{x}$, then use the rounding scheme from Lemma 1 or Theorem 3 to find an integer solution $\hat{x}$ approximating $\bar{x}$. But those rounding schemes return $\hat{x}$ such that $\hat{x} \approx L\bar{x}$. So, $\hat{x}$ would violate the multiplicity constraints by a factor of L. But L can be as large as $\Omega(\ln m)$, and we would prefer to not violate the multiplicity constraints so much.

To work around this, given a CIP $\mathcal{P} = (A, a, c, d)$, we do compute an optimal fractional solution $\bar{x}$, but then, instead of computing an integer solution $\hat{x}$ that approximates $\bar{x}$, we first compute a fractional solution $\bar{x}'$ that is what we call $(1/K)$-*granular*—meaning that each coordinate of $\bar{x}'$ is an integer multiple of $1/K$. We do this for a sufficiently large integer K, so that the $(1/K)$-granular solution $\bar{x}'$ has $\bar{x}' \approx (1+\varepsilon)\bar{x}$ (and satisfies all covering constraints). To get the final integer solution $\hat{x}$, we round $\bar{x}'$ up deterministically by rounding each coordinate up to its nearest integer. Then $\hat{x} = \lceil \bar{x}' \rceil \leqslant \lceil (1+\varepsilon)\bar{x} \rceil$. A little thought shows that this last rounding step increases the cost by at most a factor of K, so that the cost of $\hat{x}$ is $O(K)$ times the cost of $\bar{x}$.

The next lemma captures the exact tradeoff between granularity and approximation of the cost (and, implicitly, multiplicity constraints). The lemma is a straightforward consequence of the previous results.

Lemma 4. *Fix any integer* $K > 0$. *Given a* CIP_∞ *instance* (A, a, c) *and fractional solution* $\bar{x}$, *let* α *be the maximum number of constraints in which any variable appears. For some* $L = 1 + O(\ln(1+\alpha)/KW + \sqrt{\ln(1+\alpha)/KW})$, *one can compute in polynomial time a* $(1/K)$*-granular solution* $x'' \leqslant \lceil L\bar{x} \rceil$ *of cost* $O(L)$ *times the cost of* $\bar{x}$.

Proof. Here is the algorithm. The input is $\mathcal{P} = (A, a, c)$, $\bar{x}$, and K.

1. Construct CIP_∞ instance $\mathcal{P}' = (A, Ka, c)$. Let $\bar{x}' = K\bar{x}$.
2. Let $\hat{x}'$ be the integer solution obtained by applying Theorem 3 to $\mathcal{P}'$ and $\bar{x}'$.
3. Return $x'' = \hat{x}'/K$.

Step 2 is well defined as $\bar{x}'$ is a fractional solution to $\mathcal{P}'$.

By Theorem 3, $\hat{x}' \leqslant \lceil LK\bar{x} \rceil$ is an integer solution to $\mathcal{P}'$ of cost $O(KL)$ times the cost of $\bar{x}$, with $L = 1 + O(\ln(1+\alpha)/KW + \sqrt{\ln(1+\alpha)/KW})$.

Thus (using $x'' = \hat{x}'/K$), $x'' \leqslant \lceil L\bar{x} \rceil$ is a $(1/K)$-granular solution to $\mathcal{P}$ of cost $O(L)$ times the cost of $\bar{x}$. (We also use here $\lceil LK\bar{x} \rceil / K \leqslant \lceil L\bar{x} \rceil$ for integer K.) □

Note: In Step 2 of the algorithm in the proof, Lemma 1 can be used instead of Theorem 3, in which case the $1+\alpha$'s in the definition of L (in the lemma) are replaced by m's.

In the remainder of the section, by *a* (ρ, ℓ)*-bicriteria approximate solution* for a CIP, we mean an integer solution $\hat{x}$ that satisfies $Ax \geqslant a$ and $x \leqslant \lceil \ell d \rceil$, with cost at most ρ times the optimum FOPT. By a (ρ, ℓ)*-bicriteria approximation algorithm*, we mean a polynomial-time algorithm that returns (ρ, ℓ)-approximate solutions.

Our first algorithm works as follows. It first computes a $(1/K)$-granular solution $\bar{x}'$ (where $K \approx \ln(1+\alpha)/(W\varepsilon^2)$) approximating the optimal fractional solution $\bar{x}$. Then it gets an integer solution $\hat{x}$ by deterministically rounding each coordinate of $\bar{x}'$ up to the nearest integer. It returns $\hat{x}$.

Here is a sketch of the analysis. For this choice of K, $\bar{x}' = (1 + O(\varepsilon))\bar{x}$, so that $\hat{x}$ nearly satisfies the multiplicity constraints: $\hat{x} \leqslant \lceil (1 + O(\varepsilon))\bar{x} \rceil$. Since $\bar{x}'$ meets the covering constraints, so does $\hat{x}$. Finally, $\bar{x}'$ has cost $1 + O(\varepsilon)$ times the cost of $\bar{x}$, and, crucially, since $\bar{x}'$ is $(1/K)$-granular, *deterministically rounding* $\bar{x}'$ *up increases the cost by at most a factor of* K. So the final integer solution $\hat{x}$ has cost at most K times the cost of $\bar{x}'$, i.e., $O(K)$ times the cost of the original fractional solution $\bar{x}$.

The next lemma gives a detailed statement of the result and its proof.

Lemma 5. *Fix any* $\varepsilon \in (0, 1]$. *Given a CIP instance* (A, a, c, d) *and fractional solution* $\bar{x}$, *one can compute in polynomial time an* $(O(1+\ln(1+\alpha)/(W\varepsilon^2)), 1+\varepsilon)$*-bicriteria approximate solution* $\hat{x} \leqslant \lceil (1+\varepsilon)\bar{x} \rceil$.

Proof. Here is the algorithm. The input is $\mathcal{P} = (A, a, c, d)$, $\bar{x}$, and ε.

1. Take $K = \lceil \ln(1+\alpha)/W\varepsilon^2 \rceil$.

2. Obtain a $(1/K)$-granular solution $\bar{x}'$ by applying Lemma 4 to the CIP_∞ instance $\mathcal{P}' = (A, a, c)$ with fractional solution $\bar{x}$.
3. Return $\hat{x} = \lceil \bar{x}' \rceil$.

By Lemma 4, for some $L = 1 + O(\ln(1+\alpha)/KW + \sqrt{\ln(1+\alpha)/KW})$, we have that $\bar{x}' \leqslant \lceil L\bar{x} \rceil$ and that $\bar{x}'$ has cost $O(L)$ times the cost of $\bar{x}$.

It follows that $\hat{x} \leqslant \lceil L\bar{x} \rceil$ and that $\hat{x}'$ has cost $O(KL)$ times the cost of $\bar{x}$. (The latter because $\bar{x}'$ is $(1/K)$-granular, which implies that the cost of $\hat{x}$ is at most K times the cost of $\bar{x}'$.) Since (by the choice of K) $L = 1 + O(\varepsilon)$, this implies the result. □

Remark 1. The result of the lemma is best possible in the following sense. For any finite ρ, a $(\rho, 1)$-approximate solution w.r.t. FOPT is impossible because of the arbitrarily large integrality gap (see Section 3 for an example). It is also well-known that the integrality gap for FOPT_∞ is $\Omega(\ln m)$ for the special case of set cover where arbitrarily large values for the variables are allowed. Hence for any l, a (ρ, l)-approximate solution for a CIP with $\rho = o(\ln m)$ is also impossible.

Now we can state our first main result—an approximation algorithm for any general integer covering/packing problem with multiplicity constraints:

$$\text{OPT} = \min\{c^T x : x \in \mathbb{Z}_+^n,\ Ax \geqslant a,\ Bx \leqslant b,\ x \leqslant d\}.$$

The algorithm returns a solution that meets the covering constraints, approximately meets the multiplicity constraints (and hence approximately meets the packing constraints), and has cost $O(K)$ times the cost FOPT of the fractional solution.

Theorem 6 (*First main result*). *Let $\varepsilon \in (0, 1]$, and an integer covering/packing program* $\text{OPT} = \min\{c^T x : x \in \mathbb{Z}_+^n,\ Ax \geqslant a,\ Bx \leqslant b,\ x \leqslant d\}$, *with fractional solution $\bar{x}$, be given. Let $\beta_i = \sum_j B_{ij}$. Then one can compute in polynomial time an $\hat{x} \in \mathbb{Z}_+^n$ such that*

1. $c^T \hat{x} \leqslant O(1 + \ln(1+\alpha)/(W\varepsilon^2))\, c^T \bar{x}$,
2. $A\hat{x} \geqslant A\bar{x} \geqslant a$,
3. $\hat{x} \leqslant \lceil (1+\varepsilon)\bar{x} \rceil \leqslant \lceil (1+\varepsilon)d \rceil$, *and*
4. $B\hat{x} \leqslant (1+\varepsilon)\bar{x} + \beta \leqslant (1+\varepsilon)b + \beta$.

Proof. Here is the algorithm. The input is $\mathcal{P} = (A, B, a, b, c, d)$, $\bar{x}$, and ε.

1. Let $\hat{x}$ be the approximate solution obtained by applying Lemma 5 to the CIP instance $\mathcal{P}' = (A, a, c, d)$, and fractional solution $\bar{x}$.
2. Return $\hat{x}$.

Properties 1–3 of $\hat{x}$ follow immediately from Lemma 5. To see that property 4 holds, note that, from $\hat{x} \leqslant \lceil (1+\varepsilon)\bar{x} \rceil$ it follows that $\hat{x}_j < (1+\varepsilon)\bar{x}_j + 1$, which implies $(B\hat{x})_i \leqslant (B(1+\varepsilon)\bar{x})_i + \beta_i$. □

The optimal fractional solution $\bar{x}$ to the LP relaxation can be computed in polynomial time, so Theorem 6 immediately implies that the desired approximate solution $\hat{x}$ (having properties 1–4 from the theorem and cost $O(1 + \ln(1+\alpha))\text{FOPT}$) can be computed in polynomial time.

Remark 2. Note that for a CIP problem with $\max_j d_j = O(1)$, by taking $\varepsilon = 1/(2\max_j d_j)$, the above theorem implies that one can find in polynomial time an integer solution having cost $O(1 + \ln(1 + \alpha)/W)$FOPT and $\hat{x}_j \leqslant d_j + 1$. That is, the multiplicity constraints can be met within an *additive* 1.

3. Meeting the multiplicity constraints

Given a fractional solution $\bar{x}$, it is not in general possible to find an integer solution $\hat{x}$ meeting the covering and multiplicity constraints exactly and having cost $O(\ln(1 + \alpha))$ times the cost of $\bar{x}$. To see this, fix $\delta > 0$ arbitrarily small, and consider the following CIP, which is a simple instance of Minimum Knapsack:

$$\min\{x_2 : x \in \mathbb{Z}_+^2, (1 - \delta)x_1 + x_2 \geqslant 1, x_1 \leqslant 1\}.$$

The optimal fractional solution has cost δ, whereas the optimal integer solution has cost 1. This example demonstrates that the integrality gap can be arbitrarily large if multiplicity constraints are to be respected. [2] However, notice that the two constraints ($(1 - \delta)x_1 + x_2 \geqslant 1$ and $x_1 \leqslant 1$) imply a third: $x_2 \geqslant \delta$. This third constraint, and the observation that $x_2 \in \mathbb{Z}$ imply $\delta x_2 \geqslant \delta$.

The constraint "$\delta x_2 \geqslant \delta$" above is a *valid inequality* for the CIP, meaning that it holds for all feasible integer solutions. Adding a valid inequality to the integer program (IP) does not change the space of solutions or the value of the optimal solution. But adding the constraint can strengthen the linear programming relaxation by ruling out some fractional solutions, and this can give a better bound on OPT. For example, adding the constraint to the example above, and then solving the LP relaxation with the added constraint, gives a lower bound of 1 on OPT.

For the general problem, reasoning as above leads to a class of valid inequalities called Knapsack Cover (KC) inequalities. These inequalities generalize valid inequalities used for CIP problems with $A_{ij} \in \{0, 1\}$ in [1,7,25]. They were also used by Carr et al. [2].

Our next algorithm begins by finding a fractional solution $\bar{x}$ to the LP relaxation with a number of KC inequalities added. It then rounds $\bar{x}$ to an integer solution $\hat{x}$ as follows: for j such that $\bar{x}_j \geqslant d_j/(1 + \varepsilon)$, it "pins" $\hat{x}_j = d_j$. (This increases the cost by at most $1 + \varepsilon$.) To set the remaining $\hat{x}_j$'s, it rounds the corresponding $\bar{x}_j$'s using the randomized rounding algorithm from (Lemma 1) or Srinivasan's algorithm (Theorem 3). Since each non-pinned $\bar{x}_j$ is at most $d_j/(1 + \varepsilon)$, this rounding can be done so that $\hat{x}_j$ is at most d_j.

An astute reader may ask whether this process will work if started with a fractional solution $\bar{x}$ to the LP relaxation *without* KC inequalities. If so, this would yield a faster algorithm. After we describe and analyze the algorithm sketched above, we discuss this question.

3.1. The KC inequalities

Fix a problem instance $\mathcal{P} = (A, B, a, b, c, d)$. For each constraint $(Ax)_i \geqslant a_i$ and any subset F of the j's (corresponding to x_j's that we imagine pinning), define $a_i^F \doteq \max\{0, a_i - \sum_{j \in F} A_{ij} d_j\}$. Define also $A_{ij}^F \doteq \min\{A_{ij}, a_i^F\}$ for $j \in F$ and $A_{ij}^F \doteq 0$ for $j \notin F$. In words, a_i^F is the residual covering requirement

[2] A similar example appears in [2]. In [18] the integrality gap was erroneously claimed to be $H(\max_{j=1}^n \sum_{i=1}^m A_{ij})$.

of the *i*th constraint if all variables in F were to be set to their upper bounds, and A^F_{ij} is A_{ij}, possibly lowered to ensure the width is at least 1. (In the small example above, we knew that, for $x_2 \in \mathbb{Z}_+$, the inequality $x_2 \geqslant \delta$ held if and only if the inequality $\delta x_2 \geqslant \delta$ did, so we replaced the former constraint with the latter.) The KC inequalities for a set $F \subset N$ are $A^F x \geqslant a^F$. The *LP-KC relaxation* of $\mathcal{P}$ is to find $x \in \mathbb{R}^n_+$ minimizing $c^T x$ subject to $Ax \geqslant a$, $Bx \leqslant b$, $x \leqslant d$, and subject to the KC inequalities for all $F \subset N$.

We are not aware of an algorithm that solves this relaxation exactly in polynomial time. Carr et al. [2] define the following type of solutions, which are adequate for our purpose. For $\lambda > 1$, call a vector x a *λ-relaxed solution* to LP-KC if it has cost at most the fractional optimum of LP-KC and satisfies (i) $Ax \geqslant a$, (ii) $Bx \leqslant b$, (iii) $x \leqslant d$ and (iv) the KC inequalities for the set $F_\lambda = \{j : x_j \geqslant d_j/\lambda\}$. The following theorem follows from the results in [2] together with the properties of the ellipsoid method (see, e.g., [13]).

Theorem 7 (*Carr et al. [2], Lovász [13]*). *Suppose $\mathcal{P} = (A, B, a, b, c, d)$ has rational coefficients. For any constant $\lambda > 1$, a λ-relaxed solution to the LP-KC relaxation of $\mathcal{P}$ can be found in polynomial time.*

For the sake of completeness we sketch the idea behind the theorem. When the ellipsoid method queries the separation oracle with a point x, the oracle returns a separating hyperplane corresponding either to a constraint of the standard LP, or to one that is a valid KC inequality for the set of variables in x that are high (in this particular x). In the end, look at the set of hyperplanes the separation oracle has passed to the ellipsoid method. That set defines a polytope which is a relaxation of the LP-KC polytope.

The input to our next algorithm is an instance $\mathcal{P} = (A, B, a, b, c, d)$ of the general problem and an $\varepsilon \in (0, 1]$. The algorithm can also be viewed as a reduction of the problem of finding a ρ-approximate solution to a CIP to finding a (ρ, ℓ)-bicriteria approximate solution for appropriate ℓ.

1. Set $d' := \lfloor d \rfloor$.
2. Let $\bar{x}$ be a $(1+\varepsilon)$-relaxed solution to the LP-KC relaxation of $\mathcal{P} = (A, B, a, b, c, d')$.
3. Let $F = \{j : \bar{x}_j \geqslant d'_j/(1+\varepsilon)\}$.
4. Define CIP $\mathcal{P}' = (A', a', c, d'')$ by setting $A' := A^F$, $a' := a^F$, and defining fractional solution $\bar{x}'$ and d'' as follows:
5. For $j \in F$ let $\bar{x}'_j = d''_j = 0$. For $j \notin F$ let $\bar{x}'_j = d''_j = \bar{x}_j$.
6. Find integer solution $\hat{x}'$ to $\mathcal{P}'$ by applying Theorem 6 with fractional solution $\bar{x}'$ and the given ε.
7. Let $\hat{x}_j = d_j$ for $j \in F$ and $\hat{x}_j = \hat{x}'_j$ for $j \notin F$. Return $\hat{x}$.

Theorem 8 (*Second main result*). *Given $\varepsilon \in (0, 1]$, and an integer covering/packing program* $\textsc{opt} = \min\{c^T x : x \in \mathbb{Z}^n_+, Ax \geqslant a, Bx \leqslant b, x \leqslant d\}$, *let $\beta_i = \sum_j B_{ij}$. The algorithm above computes in polynomial time an $\hat{x} \in \mathbb{Z}^n_+$ such that*

1. $c^T \hat{x} \leqslant O(1 + \ln(1+\alpha)/(W\varepsilon^2))\textsc{opt}$,
2. $A\hat{x} \geqslant a$,
3. $\hat{x} \leqslant d$, *and*
4. $B\hat{x} \leqslant (1+\varepsilon)b + \beta$.

Proof. Note that the cost of $\bar{x}$ is a lower bound on $\textsc{opt}$. Observe also that Step 1 does not change the space of integer solutions.

First we bound the cost of the solution $\hat{x}'$ (to the restricted problem $\mathcal{P}'$). Since $\bar{x}$ satisfies the KC inequalities for the specific set F, the definitions of F, A', b', and d'' ensure that $\bar{x}'$ is a fractional solution of $\mathcal{P}'$. By definition of A^F, the width of $\mathcal{P}'$ is at least 1. Thus, the cost of $\hat{x}'$ is $O(\ln(1+\alpha))$ times the cost of $\bar{x}'$, which is also $O(\ln(1+\alpha))$ times the cost of $\bar{x}$, and thus $O(\ln(1+\alpha)\text{OPT})$.

Next we bound the cost of the final solution $\hat{x}$. The cost of $\hat{x}$ is at most $1+\varepsilon$ times the cost of $\bar{x}$, plus the cost of $\hat{x}'$. Thus, the cost of $\hat{x}$ is $O(\ln(1+\alpha)\text{OPT})$.

Next we verify that $\hat{x}$ does not exceed the multiplicity constraints. This is clear for the pinned variables: $\hat{x}_j = d_j$ for $j \in F$. For the other variables ($j \notin F$), we have $\hat{x}_j = \hat{x}'_j \leqslant \lceil(1+\varepsilon)d''_j\rceil = \lceil(1+\varepsilon)\bar{x}_j\rceil < \lceil(1+\varepsilon)d'_j/(1+\varepsilon)\rceil \leqslant d_j$.

Finally, $B\hat{x} \leqslant (1+\varepsilon)b + \beta$ follows from $B\bar{x} \leqslant b$ and $\hat{x} \leqslant \lceil(1+\varepsilon)\bar{x}\rceil$. □

Corollary 9. *The integrality gap of the* LP-KC *relaxation for CIP is* $O(\ln(1+\alpha))$.

3.2. Remarks on the necessity of the LP-KC relaxation

Consider for simplicity that $d' = d$. The algorithm starts with a $(1+\varepsilon)$-relaxed solution $\bar{x}$ to LP-KC, "pins'' $\hat{x}_j = d_j$ for j with $\bar{x}_j \geqslant d_j/(1+\varepsilon)$, then uses an existing bicriteria approximation algorithm to set the remaining variables. A natural question is whether the KC inequalities are necessary. Would it be enough to start with a fractional solution $\bar{x}$ to the standard LP relaxation of the CIP?

If we do this, the analysis of the algorithm (as it stands) fails because $\bar{x}'$ may no longer be a feasible solution to $\mathcal{P}'$. (Indeed, the problem $\mathcal{P}'$ may be infeasible with d'' defined as it is, or even with $d''_j = d_j/(1+\varepsilon)$. To see this, consider the simple example at the start of the section.) This breaks the argument that bounds the cost of $\hat{x}$.

Perhaps the first fix that comes to mind is to modify the algorithm to take $A'_{ij} = A_{ij}$ instead of $A'_{ij} = A^F_{ij}$ for $j \notin F$. But this does not work because the resulting $\mathcal{P}'$ can have width less than 1, worsening the approximation ratio.

Perhaps the second fix that comes to mind is to modify the algorithm to, say, set $d''_j = d_j$ for $j \notin F$, then solve $\mathcal{P}'$ from scratch to obtain a (new) optimal fractional solution $\bar{x}''$. In Step 7, the algorithm would pass that new fractional solution $\bar{x}''$ to Theorem 6 (instead of $\bar{x}'$) to compute $\hat{x}'$. Since the cost of $\bar{x}''$ is still a lower bound on OPT, it would seem that we can again bound the cost of $\hat{x}$ as desired.

The problem with this fix is that the new fractional solution $\bar{x}''$ can have $\bar{x}''_j > d_j/(1+\varepsilon)$ for $j \notin F$. Indeed, it can have $\bar{x}''_j = d_j$ for $j \notin F$. Thus, the rounded solution $\hat{x}'$ from Theorem 6 could violate the multiplicity constraints.

The natural work-around is to augment F by adding any such j to F, then start over by returning to step 4 with the new F. But, as this process may repeat many times, it is not clear how one might relate the cost of all the pinned variables to OPT.

4. Open questions

Can one find in polynomial time an integer solution for CIP with an additive 1 violation of the multiplicity constraints and logarithmic cost guarantee with respect to the standard LP optimum (without KC inequalities)? We have shown this is possible for the case $\max_j d_j = O(1)$. Is there a faster (possibly greedy?) $O(\ln m)$-approximation algorithm for CIP?

Acknowledgments

Thanks to Vijay Vazirani for a clarification on [18]. Thanks to Lisa Fleischer and Aravind Srinivasan for comments on a draft of this paper.

Appendix

Proof of Lemma 1. We prove that $\hat{x}$ is a $2L$-approximate solution with positive probability. It suffices to prove that the probability that any of the following events happens is less than 1:

$$(1)\ c^T\hat{x} > 2Lc^T\bar{x}, \quad \text{or} \quad (2)\ (\exists i)\ (A\hat{x})_i W/a_i < W.$$

Note that $E[\hat{x}] = x' = L\bar{x}$, so that by linearity of expectation

$$E[c^T\hat{x}] = LE[c^T\bar{x}] = L \times (\text{FOPT}_\infty)$$

and

$$(\forall i)\ E[(A\hat{x})_i W/a_i] = L(A\bar{x})_i W/a_i \geqslant LW.$$

By the Markov bound, the probability of (1) is at most $1/2$.

Note that each $\hat{x}_j$ can be thought of as a sum of independent random variables in $[0, 1]$ (where we consider the fixed part, $\lfloor x'_j \rfloor$, to be the sum of $\lfloor x'_j \rfloor$ variables each taking the value 1 with probability 1). Thus (by the choice of W) $(A\hat{x})_i W/a_i = \sum_j A_{ij}\hat{x}_j W/a_i$ is also a sum of independent random variables in $[0, 1]$. By a standard Chernoff bound [17],

$$\Pr[\ (A\hat{x})_i W/a_i \leqslant (1-\varepsilon)LW\ ] < \exp(-\varepsilon^2 LW/2).$$

Taking ε such that $(1-\varepsilon)L = 1$, for the choice of L in the rounding scheme, $\exp(-\varepsilon^2 LW/2) \leqslant 1/2m$. Thus, the above bound implies

$$\Pr[\ (A\hat{x})_i W/a_i \leqslant W\ ] < 1/2m.$$

Thus, by the naive union bound, the probability that (1) or (2) occurs is less than $1/2 + m/2m = 1$.

We have proven that the randomized rounding procedure returns a $2L$-approximate solution with positive probability. □

References

[1] E. Balas, Facets of the knapsack polytope, Math. Programming 8 (1975) 146–164.
[2] R.D. Carr, L.K. Fleischer, V.J. Leung, C.A. Phillips, Strengthening integrality gaps for capacitated network design and covering problems, in: Proceedings of the 11th ACM-SIAM Symposium on Discrete Algorithms, 2000.
[3] V. Chvátal, A greedy heuristic for the set-covering problem, Math. Oper. Res. 4 (3) (1979) 233–235.

[4] G. Dobson, Worst-case analysis of greedy heuristics for integer programming with non-negative data, Math. Oper. Res. 7 (4) (1982) 515–531.
[5] P. Erdős, J.L. Selfridge, On a combinatorial game, J. Combin. Theory A 14 (1973) 298–301.
[6] M.L. Fisher, L.A. Wolsey, On the greedy heuristic for continuous covering and packing problems, SIAM J. Algebraic Discrete Methods 3 (1982) 584–591.
[7] P. Hammer, E. Johnson, U. Peled, Facets of regular 0–1 polytopes, Math. Programming 8 (1975) 179–206.
[8] D.S. Hochbaum, Approximating covering and packing problems: set cover, vertex cover, independent set, and related problems, in: D.S. Hochbaum (Ed.), Approximation Algorithms for NP-hard Problems, PWS, 1997, pp. 94–143.
[9] D.S. Johnson, Approximation algorithms for combinatorial problems, J. Comput. System Sci. 9 (1974) 256–278.
[10] S.G. Kolliopoulos, Approximating covering integer programs with multiplicity constraints, Discrete Appl. Math. 129 (2003) 461–473.
[11] S.G. Kolliopoulos, N.E. Young, Tight approximation results for general covering integer programs, in: Proceedings of the 42nd Annual IEEE Symposium on Foundations of Computer Science, 2001, pp. 522–528.
[12] L. Lovász, On the ratio of optimal and fractional covers, Discrete Math. 13 (1975) 383–390.
[13] L. Lovász, An Algorithmic Theory of Numbers, Graphs and Complexity, SIAM, Philadelphia, PA, 1986.
[14] M. Naor, R.M. Roth, Optimal file sharing in distributed networks, SIAM J. Comput. 24 (1995) 158–183.
[15] S. Plotkin, D.B. Shmoys, É. Tardos, Fast approximation algorithms for fractional packing and covering problems, Math. Oper. Res. 20 (1995) 257–301.
[16] P. Raghavan, Probabilistic construction of deterministic algorithms: approximating packing integer programs, J. Comput. System Sci. 37 (1988) 130–143.
[17] P. Raghavan, C.D. Thompson, Randomized rounding: a technique for provably good algorithms and algorithmic proofs, Combinatorica 7 (1987) 365–374.
[18] S. Rajagopalan, V.V. Vazirani, Primal–dual RNC approximation algorithms for (multi)-set (multi)-cover and covering integer programs, in: Proceedings of the 34th Annual IEEE Symposium on Foundations of Computer Science, 1993, pp. 322–331.
[19] R. Raz, S. Safra, A sub-constant error-probability low-degree test, and a sub-constant error-probability PCP characterization of NP, in: Proceedings of the 29th Annual ACM Symposium on Theory of Computing, 1997, pp. 475–484.
[20] J. Spencer, Ten Lectures on the Probabilistic Method, SIAM, Philadelphia, 1987.
[21] A. Srinivasan, An extension of the Lovász local lemma and its applications to integer programming, in: Proceedings of the Seventh ACM-SIAM Symposium on Discrete Algorithms, 1996, pp. 6–15.
[22] A. Srinivasan, Improved approximations guarantees for packing and covering integer programs, SIAM J. Comput. 29 (1999) 648–670 (preliminary version in Proc. STOC 95).
[23] A. Srinivasan, C.-P. Teo, A constant-factor approximation algorithm for packet routing and balancing local vs. global criteria, SIAM J. Comput. 30 (2001) 2051–2068 (preliminary version in Proc. STOC 97).
[24] V.V. Vazirani, Approximation Algorithms, Springer, Berlin, 2001.
[25] L. Wolsey, Facets for a linear inequality in 0–1 variables, Math. Programming 8 (1975) 168–175.
[26] N.E. Young, Randomized rounding without solving the linear program, in: Proceedings of the Sixth ACM-SIAM Symposium on Discrete Algorithms, 1995, pp. 170–178.