

# On the Number of Iterations for Dantzig-Wolfe Optimization and Packing-Covering Approximation Algorithms

Philip Klein<sup>1,\*</sup> and Neal Young<sup>2,\*\*</sup>

<sup>1</sup> Brown University, Providence, RI, USA; [klein@cs.brown.edu](mailto:klein@cs.brown.edu)

<sup>2</sup> Dartmouth College, Hanover, NH, USA; [ney@cs.dartmouth.edu](mailto:ney@cs.dartmouth.edu)

## 1 Introduction

We start with definitions given by Plotkin, Shmoys, and Tardos [16]. Given  $A \in \mathbb{R}^{m \times n}$ ,  $b \in \mathbb{R}^m$  and a polytope  $P \subseteq \mathbb{R}^n$ , the *fractional packing problem* is to find an  $x \in P$  such that  $Ax \leq b$  if such an  $x$  exists. An  $\epsilon$ -approximate solution to this problem is an  $x \in P$  such that  $Ax \leq (1 + \epsilon)b$ . An  $\epsilon$ -relaxed decision procedure always finds an  $\epsilon$ -approximate solution if an exact solution exists.

A **Dantzig-Wolfe-type** algorithm for a fractional packing problem  $x \in P$ ,  $Ax \leq b$  is an algorithm that accesses  $P$  only by *queries to  $P$*  of the following form: “given a vector  $c$ , what is an  $x \in P$  minimizing  $c \cdot x$ ?”

There are Dantzig-Wolfe-type  $\epsilon$ -relaxed decision procedures (e.g. [16]) that require  $O(\rho\epsilon^{-2} \log m)$  queries to  $P$ , where  $\rho$  is the *width* of the problem instance, defined as follows:

$$\rho(A, P) = \max_{x \in P} \max_i A_i \cdot x / b_i$$

where  $A_i$  denotes the  $i^{\text{th}}$  row of  $A$ .

In this paper we give a natural probability distribution of fractional packing instances such that, for an instance chosen at random, with probability  $1 - o(1)$  any Dantzig-Wolfe-type  $\epsilon$ -relaxed procedure must make *at least*  $\Omega(\rho\epsilon^{-2} \log m)$  queries to  $P$ . This lower bound matches the aforementioned upper bound, providing evidence that the unfortunate linear dependence of the running times of these algorithms on the width and on  $\epsilon^{-2}$  is an inherent aspect of the Dantzig-Wolfe approach.

The specific probability distribution we study here is as follows. Given  $m$  and  $\rho$ , let  $A$  be a random  $\{0, 1\}$ -matrix with  $m$  rows and  $n = \sqrt{m}$  columns, where each entry of  $A$  has probability  $1/\rho$  of being 1. Let  $P$  be the  $n$ -simplex, and let  $b$  be the  $m$ -vector whose every entry is some  $v$ , where  $v$  is as small as possible so that  $Ax \leq b$  for some  $x \in P$ .

The class of Dantzig-Wolfe-type algorithms encompasses algorithms and algorithmic methods that have been actively studied since the 1950’s through the current time, including:

---

\* Research supported by NSF Grant CCR-9700146.

\*\* Research supported by NSF Career award CCR-9720664.

- an algorithm by Ford and Fulkerson for multicommodity flow [14],
- Dantzig-Wolfe decomposition (generalized linear programming) [4],
- Benders’ decomposition [3],
- the Lagrangean relaxation method developed by Held and Karp and applied to obtaining lower bounds for the traveling salesman problem [9, 10],
- the multicommodity flow approximation algorithms of Shahrokhi and Matula [18], of Klein et al. [13], and of Leighton et al. [15],
- the covering and packing approximation algorithms of Plotkin, Shmoys, and Tardos [16] and the approximation algorithms of Grigoriadis and Khachiyan [8] for block-angular convex programs, and many subsequent works (e.g. [20, 6]).

In a later section we discuss some of the history of the above algorithms and methods and how they relate to the fractional packing problem studied here.

To prove the lower bound we use a probabilistic, discrepancy-theory argument to characterize the values of random  $m \times s$  zero-sum games when  $s$  is much smaller than  $m$ . From the point of view proposed in [20], where fractional packing algorithms are derived using randomized rounding (and in particular the Chernoff bound), the intuition for the lower bound here is that it comes from the fact that the Chernoff bound is essentially tight.

Some of the multicommodity flow algorithms, and subsequently the algorithms of Plotkin, Shmoys, Tardos and of Grigoriadis and Khachiyan, use a more general model than the one described above. This model assumes the polytope  $P$  is the cross-product  $P = P^1 \times \dots \times P^k$  of  $k$  polytopes. In this model, each iteration involves optimizing a linear function over one of the polytopes  $P^i$ . It is straightforward to extend the our lower bound to this model by making  $A$  block-diagonal, thus forcing each subproblem to be independently solved. In this general case, the lower bound shows that the number of iterations must be  $\Omega(\epsilon^{-2}(\sum_i \rho_i) \log m)$ , where  $\rho_i$  is the width of  $P^i$ . This lower bound is also tight within a constant factor, as it matches the upper bounds of Plotkin, Shmoys, and Tardos.

**Previous Lower Bounds.** In 1977, Khachiyan [12] proved an  $\Omega(\epsilon^{-1})$  lower bound on the number of iterations to achieve an error of  $\epsilon$

In 1994, Grigoriadis and Khachiyan proved an  $\Omega(m)$  lower bound on the number of iterations to achieve a relative error of  $\epsilon = 1$ . They did not consider the dependence of the number of iterations on  $\epsilon$  for smaller values of  $\epsilon$ .

Freund and Schapire [5], in an independent work in the context of learning theory, prove a lower bound on the net “regret” of any adaptive strategy for playing repeated zero-sum games against an adversary. This result is related to, but different from, the result proved here.

## 2 Proof of Main Result

For any  $m$ -row  $n$ -column matrix  $A$ , define the *value* of  $A$  (considered as a two-player zero-sum matrix game) to be

$$V(A) \doteq \min_x \max_{1 \leq i \leq m} A_i \cdot x$$

where  $A_i$  denotes the  $i$ th row of  $A$  and  $x$  ranges over the  $n$ -vectors with non-negative entries summing to 1.

**Theorem 1.** *For  $m \in \mathbb{N}$ ,  $n = \Theta(m^{0.5})$ , and  $p \in (0, 1/2)$ , let  $A$  be a random  $\{0, 1\}$   $m \times n$  matrix with i.i.d. entries, each being 1 with probability  $p$ . Let  $\bar{\epsilon} > 0$ . With probability  $1 - o(1)$ ,*

- $V(A) = \Omega(p)$ , and
- for  $s \leq \min\{(\ln m)/(p\bar{\epsilon}^2), m^{0.5-\delta}\}$  (where  $\delta > 0$  is fixed), every  $m \times s$  submatrix  $B$  of  $A$  satisfies

$$V(B) > (1 + c\bar{\epsilon})V(A)$$

where  $c$  is a constant depending on  $\delta$ .

Our main result follows as a corollary.

**Corollary 1.** *Let  $m \in \mathbb{N}$ ,  $\rho > 2$ , and  $\epsilon > 0$  be given such that  $\rho\epsilon^{-2} = O(m^{0.5-\delta})$  for some constant  $\delta > 0$ .*

*For  $p = 1/\rho$ , and  $n = m^{0.5}$ , let  $A$  be a random  $\{0, 1\}$   $m \times n$  matrix as in the theorem. Let  $b$  denote the  $m$ -element vector whose every element is  $V(A)$ . Let  $P = \{x \in \mathbf{R}^n : x \geq 0, \sum_i x_i = 1\}$  be the  $n$ -simplex.*

*Then with probability  $1 - o(1)$ , the fractional packing problem instance  $x \in P, Ax \leq b$  has width  $O(\rho)$ , and any Dantzig-Wolfe-type  $\epsilon$ -relaxed decision procedure requires at least  $\Omega(\rho\epsilon^{-2} \log m)$  queries to  $P$  when given the instance as input.*

Assuming the theorem for a moment, we prove the corollary. Suppose that the matrix  $A$  indeed has the two properties that hold with probability  $1 - o(1)$  according to the theorem. It follows from the definition of  $V(A)$  that there exists  $x^* \in P^n$  such that  $Ax^* \leq b$ . That is, there exists a (non-approximate) solution to the fractional packing problem.

To bound the width, let  $\bar{x}$  be any vector in  $P$ . By definition of  $P$  and  $A$ , for any row  $A_j$  of  $A$  we have  $A_j \cdot \bar{x} \leq 1$ . On the other hand, from the theorem we know that  $V(A) = \Omega(p) = \Omega(1/\rho)$ . Since  $b_j = V(A)$ , it follows that  $A_j \cdot \bar{x}/b_j$  is  $O(\rho)$ . Since this is true for every  $j$  and  $\bar{x} \in P$ , this bounds the width.

Now consider any Dantzig-Wolfe-type  $\epsilon$ -relaxed decision procedure. Suppose for a contradiction that it makes no more than  $s \leq \rho(c\epsilon)^{-2} \ln m$  calls to the oracle that optimizes over  $P$ . In each of these calls, the oracle returns a vertex of  $P$ , i.e. a vector of the form

$$(0, 0, \dots, 0, 1, 0, \dots, 0, 0)$$

Let  $S$  be the set of vertices returned, and let  $P(S)$  be the convex hull of these vertices. Every vector in  $P(S)$  has at most  $s$  non-zero entries, for its only non-zero entries can occur in positions for which there is a vector in  $S$  having a 1 in that position. Hence, by the theorem with  $\bar{\epsilon} = \epsilon/c$ , there is no vector  $x \in P(S)$  that satisfies  $Ax \leq (1 + \epsilon)b$ .

Consider running the same algorithm on the fractional packing problem  $Ax \leq b, x \in P(S)$ , i.e. with  $P(S)$  replacing  $P$ . The procedure makes all the same queries to  $P$  as before, and receives all the same answers, and hence must give the same output, namely that an  $\epsilon$ -approximate solution exists. This is an incorrect output, which contradicts the definition of a relaxed decision procedure.

### 3 Proof of Theorem 1

For any  $m$ -row  $n$ -column matrix  $A$ , define the *value* of  $A$  (considered as a two-player zero-sum matrix game) to be

$$V(A) = \min_x \max_{1 \leq i \leq m} A_i \cdot x$$

where  $A_i$  denotes the  $i$ th row of  $A$  and  $x$  ranges over the  $n$ -vectors with non-negative entries summing to 1.

Before we give the proof of Theorem 1, we introduce some simple tools for reasoning about  $V(X)$  for a random  $\{0, 1\}$  matrix  $X$ .

By the definition of  $V$ ,  $V(X)$  is at most the maximum, over all rows, of the average of the row's entries. Suppose each entry in  $X$  is 1 with probability  $q$ , and within any row of  $X$  the entries are independent. Then for any  $\delta$  with  $0 < \delta < 1$ , a standard Chernoff bound implies that the probability that a given row's average exceeds  $(1 + \delta)q$  is  $\exp(-\Theta(\delta^2 q n_X))$ , where  $n_X$  is the number of columns of  $X$ . Thus, by a naive union bound  $\Pr[V(X) \geq (1 + \delta)q] \leq m_X \exp(-\Theta(\delta^2 q n_X))$  where  $m_X$  is the number of rows of  $X$ . For convenience we rewrite this bound as follows. For any  $q \in [0, 1]$  and  $\beta \in (0, 1]$ , assuming  $m_X/\beta \rightarrow \infty$ ,

$$\Pr[V(X) \geq (1 + \delta)q] = o(\beta) \text{ for some } \delta = O\left(\sqrt{\frac{\ln(m_X/\beta)}{q n_X}}\right). \quad (1)$$

We use an analogous *lower* bound on  $V(X)$ . By von Neumann's Min-Max Theorem

$$V(X) = \max_y \min_i X'_i \cdot y$$

(where  $X'$  denotes the transpose of  $X$ ). Thus, reasoning similarly, if within any *column* of  $X$  (instead of any row) the entries are independent,

$$\Pr[V(X) \leq (1 - \delta)q] = o(\beta) \text{ for some } \delta = O\left(\sqrt{\frac{\ln(n_X/\beta)}{q m_X}}\right), \quad (2)$$

assuming  $n_X/\beta \rightarrow \infty$ . We will refer to (1) and (2) as the *naive upper and lower bounds* on  $V(X)$ , respectively.

*Proof of Theorem 1.*

The naive lower bound to  $V(A)$  shows that

$$\Pr[V(A) \leq p(1 - \delta_0)] = o(1) \text{ for some } \delta_0 = O\left(\sqrt{\frac{\ln n}{pm}}\right) = o(1). \quad (3)$$

Thus,  $V(A) \geq \Omega(p)$  with probability  $1 - o(1)$ .

Let  $s = \min\{(\ln m)/(p\bar{\epsilon}^2), m^{0.5-\delta}\}$ . Assume without generality that  $s = (\ln m)/(p\bar{\epsilon}^2)$  (by increasing  $\bar{\epsilon}$  if necessary).

We will show that with probability  $1 - o(1)$  any  $m \times s$  submatrix  $B$  of  $A$  has value

$$V(B) > (1 + c\bar{\epsilon})V(A) \tag{4}$$

The definition of value implies that  $V(B') \geq V(B)$  for any  $m \times s'$  submatrix  $B'$  of  $B$  (where  $s' \leq s$ ). Thus we obtain (4) for such submatrices  $B'$  as well.

For any of the  $m$  rows of  $B$ , the expected value of the average of the  $s$  entries is  $p$ . We will show at least  $r = s^2 \ln n$  of the rows have a higher than average number of ones and by focusing on these rows we will show that  $V(B)$  is likely to be significantly higher than  $V(A)$ .

For appropriately chosen  $\delta_1$ , the probability that a given row of  $B$  has at least  $\ell = (1 + \delta_1)ps$  ones is at least  $\binom{s}{\ell} p^\ell (1-p)^{s-\ell} = \exp(-O(\delta_1^2 ps))$ . (That is, the Chernoff bound is essentially tight here up to constant factors in the exponent.) Call any such row *good* and let  $G$  denote the number of good rows. In particular choosing some

$$\delta_1 = \Omega\left(\sqrt{\frac{\ln(m/r)}{ps}}\right) = \Omega\left(\sqrt{\frac{\ln m}{ps}}\right) = \Omega(\epsilon)$$

the probability that any given row is good is at least  $2r/m$  and the expectation of  $G$  is at least  $2r$ . Since  $G$  is a sum of independent random  $\{0, 1\}$  random variables,  $\Pr[G < r] < \exp(-r/8)$ .

By the choice of  $r$ , this is  $o(1/n^s)$ , so with probability  $1 - o(1/n^s)$ ,  $B$  has at least  $r$  good rows.

Suppose this is indeed the case and select any  $r$  good rows. Let  $C$  be the  $r \times s$  submatrix of  $B$  formed by the chosen rows. In any column of  $C$ , the entries are independent and by symmetry each has probability at least  $p(1 + \delta_1)$  of being 1. Applying the naive lower bound (2) to  $V(C)$ , we find

$$\Pr[V(C) \leq p(1 + \delta_1)(1 - \delta_2)] = o(1/n^s) \text{ for some } \delta_2 = O\left(\sqrt{\frac{s \ln n}{pr}}\right). \tag{5}$$

By the choice of  $r$ ,  $\delta_2 = o(\delta_1)$ . Thus  $(1 + \delta_1)(1 - \delta_2) = 1 + \Omega(\delta_1)$ . Since  $V(B) \geq V(C)$ , we find that, for any  $m \times s$  submatrix  $B$ ,  $V(B) \geq p(1 + \Omega(\delta_1))$  with probability  $1 - o(1/n^s)$ .

Since there are at most  $\binom{n}{s} \leq n^s$  distinct  $m \times s$  submatrices  $B$  of  $A$ , the probability that all of them have value  $p(1 + \Omega(\delta_1))$  is  $1 - o(1)$ . Finally, applying the naive upper bound to  $V(A)$  shows that

$$\Pr[V(A) \geq p(1 + \delta_3)] = o(1) \text{ for some } \delta_3 = O\left(\sqrt{\frac{\ln m}{pn}}\right). \tag{6}$$

Since  $\delta_3 = o(\delta_1)$ , the result follows. □

## 4 Historical Discussion

Historically, there are three lines of research within what we might call the Dantzig-Wolfe model. One line of work began with a method proposed by Ford and Fulkerson for computing multicommodity flow. Dantzig and Wolfe noticed that this method was not specific to multicommodity flow; they suggested decomposing an arbitrary linear program into two sets of constraints, writing it as

$$\min\{cx : x \geq 0, Ax \geq b, x \in P\} \quad (7)$$

and solving the linear program by an iterative procedure: each iteration involves optimizing over the polytope  $P$ . This approach, now called Dantzig-Wolfe decomposition, is especially useful when  $P$  can be written as a cross-product  $P_1 \times \cdots \times P_k$ , for in this case minimization over  $P$  can be accomplished by minimizing separately over each  $P_i$ . Often, for example, distinct  $P_i$ 's constrain disjoint subsets of variables. In practice, this method tends to require many iterations to obtain a solution with value optimum or nearly optimum, often too many to be useful.

### Lagrangian Relaxation

A second line of research is represented by the work of Held and Karp [9, 10]. In 1970 they proposed a method for estimating the minimum cost of a traveling-salesman tour. Their method was based on the concept of a *1-tree*, which is a slight variant of a spanning tree. They proposed two ways to calculate this estimate; one involved formulating the estimate as the solution to the mathematical program

$$\max_u \left[ ub + \min_{x \in P} (c - uA)x \right] \quad (8)$$

where  $P$  is the polytope whose vertices are the 1-trees. They suggested an iterative method to find an optimal or near-optimal solution: While they given some initial assignment to  $u$ , find a minimum-cost 1-tree with respect to the edge-costs  $c - uA$ . Next, update the node-prices  $u$  based on the degrees of the nodes in the 1-tree found. Find a min-cost 1-tree with respect to the modified costs, update the node-prices accordingly, and so on.

Like Dantzig and Wolfe's method, this method's only dependence on the polytope  $P$  is via repeatedly optimizing over it. In the case of Held and Karp's estimate, optimizing over  $P$  amounts to finding a minimum-cost spanning tree. Their method of obtaining an estimate for the solution to a discrete-optimization problem came to be known as Lagrangian relaxation, and has been applied to a variety of other problems.

Held and Karp's method for finding the optimal or near-optimal solution to (8) turns out to be the *subgradient method*, which dates back to the early sixties. Under certain conditions this method can be shown to converge in the limit, but, like Dantzig and Wolfe's method it can be rather slow. (One author refers to the "the correct combination of artistic expertise and luck" [19] needed to make progress in subgradient optimization.)

## Fractional Packing and Covering

The third line of research, unlike the first two, provided guaranteed convergence rates. Shahrokhi and Matula [18] gave an approximation algorithm for a special case of multicommodity flow. Their algorithm was improved and generalized by Klein, Plotkin, Stein, and Tardos [13], Leighton et al. [15], and others. Plotkin, Shmoys, and Tardos [16] noticed that the technique could be generalized to apply to the problem of finding an element of the set

$$\{x : Ax \leq b, x \in P\} \quad (9)$$

where  $P$  is a convex set and  $A$  is a matrix such that  $Ax \geq 0$  for every  $x \in P$ . In particular, as discussed in the introduction, they gave an  **$\epsilon$ -relaxed decision procedure** that required  $O(\rho\epsilon^{-2} \log m)$  queries to  $P$ , where  $\rho$  is the *width* of the problem instance.

A similar result was obtained independently by Grigoriadis and Khachiyan [8]. Many subsequent algorithms (e.g. [20, 6]) built on these results. Furthermore, many applications for these results have been proposed.

This method of Plotkin, Shmoys, Tardos and Grigoriadis, Khachiyan improves on Dantzig-Wolfe decomposition and subgradient optimization in that it does not require artistry to achieve convergence, and it is effective for reasonably large values of  $\epsilon$ . However, for small  $\epsilon$  the method is frustratingly slow. Might there be an algorithm in the Dantzig-Wolfe model that converges more quickly?

Our aim in this paper has been to address this question, and to provide evidence that the answer is no. However, our lower bound technique is incapable of proving a lower bound that is superlinear in  $m$ , the number of rows of  $A$ . The reason is that for any  $m$ -row matrix  $A$ , there is an  $m$ -column submatrix  $B$  such that  $V(B) = V(A)$ . This raises the question of whether there is a Dantzig-Wolfe-type method that requires a number of iterations polynomial in  $m$  but subquadratic in  $1/\epsilon$ .

## References

- [1] B. Awerbuch and T. Leighton. A simple local-control approximation algorithm for multicommodity flow. In *Proc. of the 34th IEEE Annual Symp. on Foundation of Computer Science*, pages 459–468, 1993.
- [2] B. Awerbuch and T. Leighton. Improved approximation algorithms for the multicommodity flow problem and local competitive routing in dynamic networks. In *Proc. of the 26th Ann. ACM Symp. on Theory of Computing*, pages 487–495, 1994.
- [3] J. F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik* 4:238–252, 1962.
- [4] G. B. Dantzig and P. Wolfe. Decomposition principle for linear programs. *Operations Res.*, 8:101–111, 1960.
- [5] Y. Freund and R. Schapire. Adaptive game playing using multiplicative weights. *J. Games and Economic Behavior*, to appear.
- [6] N. Garg and J. Könemann. Faster and simpler algorithms for multicommodity flow and other fractional packing problems. In *Proc. of the 39th Annual Symp. on Foundations of Computer Science*, pages 300–309, 1998.

- [7] A. V. Goldberg. A natural randomization strategy for multicommodity flow and related problems. *Information Processing Letters*, 42:249–256, 1992.
- [8] M. D. Grigoriadis and L. G. Khachiyan. A sublinear-time randomized approximation algorithm for matrix games. Technical Report LCSR-TR-222, Rutgers University Computer Science Department, New Brunswick, NJ, April 1994.
- [9] M. Held and R. M. Karp. The traveling salesman problem and minimum spanning trees. *Operations Research*, 18:1138–1162, 1971
- [10] M. Held and R. M. Karp. The traveling salesman problem and minimum spanning trees: Part II. *Mathematical Programming*, 1:6–25, 1971.
- [11] D. Karger and S. Plotkin. Adding multiple cost constraints to combinatorial optimization problems, with applications to multicommodity flows. In *Proc. of the 27th Ann. ACM Symp. on Theory of Computing*, pages 18–25, 1995.
- [12] L. G. Khachiyan. Convergence rate of the game processes for solving matrix games. *Zh. Vychisl. Mat. and Mat. Fiz.*, 17:1421–1431, 1977. English translation in *USSR Comput. Math and Math. Phys.*, 17:78–88, 1978.
- [13] P. Klein, S. Plotkin, C. Stein, and E. Tardos. Faster approximation algorithms for the unit capacity concurrent flow problem with applications to routing and finding sparse cuts. *SIAM Journal on Computing*, 23(3):466–487, June 1994.
- [14] L. R. Ford and D. R. Fulkerson. A suggested computation for maximal multicommodity network flow. *Management Sci.*, 5:97–101, 1958.
- [15] T. Leighton, F. Makedon, S. Plotkin, C. Stein, É. Tardos, and S. Tragoudas. Fast approximation algorithms for multicommodity flow problems. *Journal of Computer and System Sciences*, 50(2):228, 1995.
- [16] S. Plotkin, D. Shmoys, and É. Tardos. Fast approximation algorithms for fractional packing and covering problems. *Mathematics of Operations Research*, 20:257–301, 1995.
- [17] T. Radzik. Fast deterministic approximation for the multicommodity flow problem. In *Proc. of the Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, 486–492, 1995.
- [18] F. Shahrokhi and D. W. Matula. The maximum concurrent flow problem. *JACM*, 37:318–334, 1990.
- [19] J. F. Shapiro. A survey of lagrangean techniques for discrete optimization. *Annals of Discrete Mathematics*, 5:113–138, 1979.
- [20] N. E. Young. Randomized rounding without solving the linear program. In *Proc. of the Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 170–178, 1995.