Fast Approximation Algorithms for Fractional Packing

Blake Sisson

This is an outline of the main ideas and results from 'Fast Approximation Algorithms For Fractional Packing And Covering Problems' by Plotkin, Shmoys, Tardos.

1 Introduction

We are interested in solving very large cases of the fractional packing problem (FP):

$$\text{find } x \in P \text{ such that } Ax \leq b \tag{1}$$

where $P \in \mathbb{R}^n$ is a polytope, $A \in \mathbb{R}^{m \times n}$, $b > 0$, and for any $x \in P$, we have $Ax \geq 0$. Moreover, we assume we have a fast subroutine for optimizing a nonnegative cost over P. This problem naturally arises when solving multi-commodity flow problems, such as those seen in power systems operation and vehicle routing.

In this setting, traditional linear programming tools (such as the simplex and interior point methods), which rely on expensive matrix computations (e.g. inversion and Chloesky factorization), often require inordinate amount of time and are thus no longer feasible. Instead we employ a particular first order algorithm based on a potential reduction strategy. The benefit of this approach is that we bypass the need for expensive matrix storage and computation, allowing faster iterations and manageable running times.

2 Relaxation and Definitions

We now present a relaxation for the fractional packing problem and some useful definitions. Relax (FP) into an optimization problem over $P \times \mathbb{R}^+$ given by

$$\min \lambda \text{ s.t. } Ax \leq \lambda b \tag{2}$$

with optimal value λ^*. For a given x, denote its corresponding minimum lambda as λ_x, i.e. $\lambda_x = argmin\{\lambda | Ax \leq \lambda b\}$. Note, (x, λ_x) is clearly feasible and it is easy to see that

$$\lambda_x = \max_{i=1,\ldots,m} \frac{a_i x}{b_i}$$

where a_i is the ith row of A.

For a given tolerance $\epsilon > 0$, let us denote an ϵ-approximate solution for (1) as a point $x \in P$ such that $Ax \leq (1+\epsilon)b$. In contrast, we say $x \in P$ is ϵ-optimal for (2) if $\lambda_x \leq (1+\epsilon)\lambda^*$. Note that if x is ϵ-optimal but $\lambda_x > 1+\epsilon$, then there are no exact solutions to (FP). On the other hand, if $\lambda_x \leq 1+\epsilon$, then x is an ϵ-approximate solution to (FP).

3 Assuring relaxed optimality

We would like to be able to certify the ϵ-optimality of a point without needing λ^*, whose value is unknown. To this end, denote $C_P(y)$ as the minimum cost $y^T Ax$ over $x \in P$, for any nonnegative $y \in \mathbb{R}^m$. Now for any feasible $x \in P$ and any $y \geq 0$, we have

$$C_P(y) \leq y^T Ax \leq \lambda_x y^T b \tag{3}$$

This implies that $\lambda_x \geq \frac{C_P(y)}{y^T b}$, and since $C_P(y)$ and $y^T b$ are independent of x, we see that

$$\lambda^* \geq \frac{C_P(y)}{y^T b}$$

Thus the ϵ-optimality of $x \in P$ is shown by the existence of a $\hat{y} \geq 0$ so that $\lambda_x \leq (1+\epsilon)C_P(\hat{y})/\hat{y}^T b$. For algorithmic convenience we will decompose this into two separate conditions. First, we observe the following fact:

Fact 3.1. *There exists a $y^* \geq 0$ such that, at the optimal (x^*, λ^*), we have*

$$C_P(y^*) = (y^*)^T Ax^* = \lambda^*(y^*)^T b \qquad i.e. \quad \lambda^* = \frac{C_P(y^*)}{(y^*)^T b}$$

Proof. Using Lagrangian relaxation we have

$$\begin{aligned}
\lambda^* &= \min_{x \in P, \lambda \geq 0} \{\lambda \mid Ax \leq \lambda b\} = \min_{x \in P, \lambda \geq 0} \max_{y \geq 0} \lambda + y^T(Ax - \lambda b) \\
&\geq \max_{y \geq 0} \min_{x \in P, \lambda \geq 0} y^T Ax + \lambda(1 - y^T b) = \max_{y \geq 0} \{\min_{x \in P} y^T Ax \mid y^T b \leq 1\} \\
&= \max_{y \geq 0} \{C_P(y) \mid y^T b \leq 1\}
\end{aligned}$$

Strong duality implies there exists a $y^* \geq 0$ such that $\lambda^* = \lambda^* + (y^*)^T(Ax^* - \lambda^* b) = (y^*)^T Ax^* + \lambda^*(1 - (y^*)^T b) = C_P(y^*)$, which gives $(y^*)^T Ax^* = \lambda^*(y^*)^T b$ and $\lambda^* = \lambda^*(y^*)^T b$ by the first and third equality respectively. Hence the claim holds. $\square$

This fact suggests that a point and a dual solution are close optimality if all three terms in 3 are nearly equal. We can quantify this via the following two relaxed optimality conditions:

$$\begin{aligned}
&(P1) \quad (1-\epsilon)\lambda_x y^T b \leq y^T Ax \\
&(P2) \quad y^T Ax - C_P(y) \leq \epsilon(y^T Ax + \lambda_x y^T b)
\end{aligned}$$

Condition $(P1)$ guarantees that $y^T Ax$ is within ϵ of $\lambda_x y^T b$, while $(P2)$ bounds the relative difference between $y^T Ax$ and $C_P(y)$ by ϵ. Now, as the following result confirms, whether some $y \geq 0$ certifies the $O(\epsilon)$-optimality of given $x \in P$ can be verified by checking if x, y, ϵ satisfy these two conditions.

Lemma 3.2. *If $x \in P$, $y \geq 0$ satisfy the relaxed optimality conditions $(P1)$ and $(P2)$ for $0 \leq \epsilon \leq 1/6$, then x is 6ϵ-optimal.*

Proof. We first lower bound $C_P(y)$,

$$\begin{aligned} C_P(y) &\geq (1-\epsilon)y^T Ax - \epsilon\lambda_x y^T b && \text{by } (P2) \\ &\geq (1-\epsilon)^2\lambda_x y^T b - \epsilon\lambda_x y^T b && \text{by } (P1) \\ &= (1 - 3\epsilon + \epsilon^2)\lambda_x y^T b \\ &\geq (1-3\epsilon)\lambda_x y^T b && \text{since } \lambda_x y^T b \geq 0 \end{aligned}$$

Rearranging gives

$$\lambda_x \leq (1-3\epsilon)^{-1}\frac{C_P(y)}{y^T b}$$

Note that $(1-3\epsilon)^{-1} \leq 1 + 6\epsilon$ since, because $\epsilon \leq 1/6$, $\frac{1}{1-3\epsilon} - 1 = \frac{3\epsilon}{1-3\epsilon} \leq 6\epsilon$. Now, recalling that $\frac{C_P(y)}{y^T b}$ lower bounds λ^*, we have

$$\lambda_x \leq (1+6\epsilon)\lambda^*$$

as desired.

$\square$

4 Improving Solutions

With the goal of developing an iterative procedure, say that we are given a point $x \in P$ and would like to incrementally improve it. Choose $y \geq 0$ defined by $y_i = (1/b_i)e^{\alpha a_i x/b_i}$, for a constant α. As the following result shows, given a sufficiently large α we can guarantee that this x and y satisfy $(P1)$ for a given tolerance. We call such y the <u>corresponding dual solution</u> to x.

Lemma 4.1. *If $\alpha \geq 2\lambda_x^{-1}\epsilon^{-1}\log(2m\epsilon^{-1})$, then x and its corresponding dual solution y satisfy $(P1)$ for tolerance ϵ.*

Proof. Consider a localized version of $(P1)$

$$(\hat{P}1) \quad \forall i = 1, \ldots, m \text{ either } (1 - \frac{\epsilon}{2})\lambda_x b_i \leq a_i x \text{ or } y_i b_i \leq \frac{\epsilon}{2m} y^T b$$

We first show $(\hat{P}1)$ implies $(P1)$, and then that the hypothesis of the lemma implies $(\hat{P}1)$.

- Assume $(\hat{P}1)$ holds and define $I = \{i : (1-\frac{\epsilon}{2})\lambda_x b_i \le a_i x\}$. Now,

  $$\lambda_x y^T b = \sum_{i\in I} \lambda_x y_i b_i + \sum_{i\notin I} \lambda_x y_i b_i \le \frac{1}{1-\frac{\epsilon}{2}} \sum_{i\in I} y_i a_i x + \frac{\epsilon}{2m}\lambda_x y^T bm = \frac{1}{1-\frac{\epsilon}{2}} y^T Ax + \frac{\epsilon}{2}\lambda_x y^T b$$

  which gives

  $$(1-\frac{\epsilon}{2})^2 \lambda_x y^T b \le y^T Ax$$

  This readily implies $(P1)$ since

  $$(1-\epsilon)\lambda_x y^T b \le (1-\epsilon+\frac{\epsilon^2}{4})\lambda_x y^T b = (1-\frac{\epsilon}{2})^2 \lambda_x y^T b \le y^T Ax$$

- First note $y^T b = \sum_i e^{\alpha a_i x/b_i} \ge e^{\alpha\lambda_x}$ by the definition of λ_x and since a positive series is greater in value than its largest term. Now consider an index i with $(1-\frac{\epsilon}{2})\lambda_x b_i > a_i x$. Here, $y_i b_i = e^{a_i x/b_i} < e^{(1-\frac{\epsilon}{2})\alpha\lambda_x}$. And so,

  $$\frac{y_i b_i}{y^T b} \le \frac{e^{(1-\frac{\epsilon}{2})\alpha\lambda_x}}{e^{\alpha\lambda_x}} = e^{-\frac{\epsilon}{2}\alpha\lambda_x} \le e^{-\log(2m\epsilon^{-1})} = \frac{\epsilon}{2m}$$

  where the second inequality holds by our assumption on α. Thus, by rearranging,

  $$y_i b_i \le \frac{\epsilon}{2m} y^T b$$

  □

Now, it is unlikely that x and y also satisfy $(P2)$ for this same tolerance, so we are not yet at an optimal point. We now focus on selecting a new point in P that reduces the slack between $y^T Ax$ and $C_P(y)$ while decreasing the value of λ_x, or, at the very least, assuring λ_x decreases over many consecutive iterations. To guide our selection process, we introduce a potential function, Φ, that measures how much a point violates the packing constraints

$$\Phi(x) = y^T b = \sum_i e^{\alpha a_i x/b_i}$$

Notice that large violations of $Ax \le b$ will result in greater fractions, $a_i x/b_i$, and cause the exponential functions in the summation to blow up. Moreover, a step along a descent direction of Φ will reduce the average constraint violation. A sequence of such steps, as we will show, gradually decreases λ_x. Furthermore, it is easy to check that $\nabla\Phi^T = y^T A$. Hence, taking a Franke-Wolfe iteration to decrease Φ over P will, by definition, decrease $y^T Ax$ and thus shrink the gap with $C_P(y)$.

Now, let $\tilde{x}$ achieve the minimum cost $\nabla\Phi^T x = y^T Ax$ over $x \in P$. As the following result shows, choosing a small enough step in the direction of $\tilde{x}$ gives a significant decrease in the potential function.

For our use in this Lemma and the results to come, we define the width of P relative to the constraints $Ax \le b$ as

$$\rho = \max_{x\in P} \lambda_x\,.$$

Lemma 4.2. *Let $x \in P$, $y \geq 0$ be its corresponding dual solution, and $0 < \epsilon \leq 1$ be a given tolerance. Assume x, y, ϵ do not satisfy $(P2)$. Let $\tilde{x}$ achieve the minimum $C_P(y)$. Taking step size $\sigma \leq \epsilon/(4\alpha\rho)$ we have that a new solution $\hat{x} = x + \sigma(\tilde{x} - x)$ satisfies $\Phi(x) - \Phi(\hat{x}) \geq \alpha\sigma\epsilon\lambda_x\Phi$*

Proof. By definition of the width ρ, $Ax \leq \rho b$ and $A\tilde{x} \leq \rho b$. Thus, by our assumption of σ and the positivity of Ax and ρb,

$$\frac{\alpha\sigma|a_i x - a_i\tilde{x}|}{b_i} \leq \frac{\epsilon|a_i x - a_i\tilde{x}|}{4\rho b_i} \leq \frac{\epsilon}{4} < \frac{1}{4}$$

Second-order Taylor's Theorem tells us that for $\delta < 0$ with $|\delta| \leq \frac{\epsilon}{4} \leq \frac{1}{4}$ and any x,

$$\begin{aligned}
e^{x+\delta} &= e^x + \delta e^x + \frac{1}{2}\delta^2 e^{x+t\delta} && t \in (0,1) \\
&\leq e^x + \delta e^x + \frac{1}{2}|\delta||\delta|e^x && \text{(since } exp(\cdot) \text{ is monotonically increasing)} \\
&\leq e^x + \delta e^x + \frac{\epsilon}{8}|\delta|e^x && (|\delta| \leq \frac{\epsilon}{4}) \\
&\leq e^x + \delta e^x + \frac{\epsilon}{2}|\delta|e^x
\end{aligned}$$

Set $\delta = \alpha\sigma(a_i\tilde{x} - a_i x)/b_i$, which we know is negative since $a_i x$ must be minimal for $\tilde{x}$ by assumption. This gives

$$\begin{aligned}
\hat{y}_i &= \frac{1}{b_i} e^{\alpha a_i(x + \sigma(\tilde{x} - x))/b_i} \\
&= \frac{1}{b_i} e^{\alpha a_i x/b_i + \alpha\sigma(a_i\tilde{x} - a_i x)/b_i} \\
&= y_i + \frac{1}{b_i}\frac{\alpha\sigma(a_i\tilde{x} - a_i x)}{b_i} e^{\alpha a_i x/b_i} + \frac{1}{b_i}\frac{\epsilon\alpha\sigma|a_i\tilde{x} - a_i x|}{2b_i} e^{\alpha a_i x/b_i} \\
&\leq y_i + \alpha\sigma\frac{1}{b_i}(a_i\tilde{x} - a_i x)y_i + \epsilon\alpha\sigma\frac{1}{2b_i}(a_i\tilde{x} + a_i x)y_i
\end{aligned}$$

where the third equality holds by second-order Taylor's Theorem and the inequality holds by triangle inequality. Applying this to bound the change in potential, we get

$$\begin{aligned}
\Phi(x) - \Phi(\hat{x}) &= \sum_i (y_i - \hat{y}_i) b_i \\
&\geq \alpha\sigma \sum_i (a_i x - a_i\tilde{x}) y_i - \alpha\sigma\frac{\epsilon}{2}\sum_i (a_i\tilde{x} + a_i x) y_i \\
&= \alpha\sigma(y^T Ax - y^T A\tilde{x}) - \alpha\sigma\frac{\epsilon}{2}(y^T A\tilde{x} + y^T Ax) \\
&\geq \alpha\sigma(y^T Ax - C_P(y)) - \alpha\sigma\frac{\epsilon}{2}(y^T Ax + y^T Ax) \\
&= \alpha\sigma(y^T Ax - C_P(y)) - \alpha\sigma\epsilon y^T Ax \\
&> \alpha\sigma\epsilon(y^T Ax + \lambda_x y^T b) - \alpha\sigma\epsilon y^T Ax \\
&= \alpha\sigma\epsilon\lambda_x\Phi(x)
\end{aligned}$$

where the second inequality holds by definition on $\tilde{x}$ and the last inequality is true since $(P2)$ is assumed to be unsatisfied.

$\square$

5 The Improve-Packing Subroutine

Motivated by our previous discussion, we propose the following subroutine which, given an initial point $x_0 \in P$ and tolerance $\epsilon > 0$, finds a new point, $x \in P$, that is either 6ϵ-optimal or has $\lambda_x \leq \lambda_{x_0}/2$

Algorithm 1 Improve-Packing(x, ϵ)

```
Initialize λ0 ← max_i a_i x/b_i;  α ← 4λ0^{-1} ε^{-1} log(2mε^{-1});  σ ← ε/(4αρ)
While λ_x := max_i a_i x/b_i ≥ λ0/2
   For each i = 1,...,m: set y_i ← (1/b_i) e^{α a_i x/b_i}
   If x and y satisfy (P2)
      Break
   Find a min-cost point x̃ ∈ P for costs y^T A
   Update x ← (1 − σ)x + σx̃
Return x
```

Note that the choice of $\alpha = 4\lambda_0^{-1}\epsilon^{-1}\log(2m\epsilon^{-1})$ assures that the relaxed optimally condition $(P1)$ is satisfied throughout the subroutine's execution as desired. Indeed, let x_k be the point in P after the kth iteration of the while loop, y_k be its corresponding dual solution, and $\lambda_k := \lambda_{x_k}$. Assuming the we enter the next loop, i.e. $\lambda_k \geq \lambda_0$ holds, we have that x_k and y_k satisfy the hypothesis for lemma 4.1 since $2\lambda_{k+1}^{-1}\epsilon^{-1}\log(2m\epsilon^{-1}) \leq 4\lambda_0^{-1}\epsilon^{-1}\log(2m\epsilon^{-1}) = \alpha\,.$

In addition, by lemma 4.2, the choice of $\sigma = \epsilon/(4\alpha\rho)$ guarantees that each iteration makes progress towards satisfying $(P2)$, so at termination we either have a 6ϵ-optimal solution or a point whose maximum constraint violation has been reduced by at least half.

The following result bounds the number of iterations required by a call to the Improve-packing subroutine.

Theorem 5.1. *The Improve-Packing procedure terminates after $O(\epsilon^{-3}\lambda_0^{-1}\rho\log(m\epsilon^{-1}))$ iterations. Moreover, if the input is $O(\epsilon)$-optimal, then Improve-Packing terminates after $O(\epsilon^{-2}\lambda_0^{-1}\rho\log(m\epsilon^{-1}))$ iterations.*

Proof. We first note that the decrease in potential function due to a single iteration is

$$\Omega((\epsilon^2\lambda_0/\rho)\Phi)$$

This is because, by lemma 4.2, the decrease is bounded by $\alpha\sigma\epsilon\lambda\Phi = \epsilon^2\lambda/(4\rho)\Phi \geq \epsilon^2\lambda_0/(8\rho)\Phi$ where we have employed the definition of σ and the relation $\lambda \geq \lambda_0/2\,.$ Additionally, through the execution of Improve-Packing we have

$$e^{\alpha\lambda_0/2} \leq \Phi(x) \leq me^{\alpha\lambda_0}$$

because $e^{\alpha\lambda_0/2} \leq e^{\alpha\lambda_x} \leq \sum_i e^{\alpha a_i x/b_i} = \Phi(x) \leq \Phi(x_0) = \sum_i e^{\alpha a_i x_0/b_i} \leq me^{\alpha\lambda_0}$.

Now, a simple inductive argument shows that after k iterations the value of the potential function is

$$(1 - \frac{\epsilon^2\lambda_0}{\rho})^k \, \Phi(x_0) \leq (1 - \frac{\epsilon^2\lambda_0}{\rho})^k \, me^{\alpha\lambda_0}$$

Letting $r := 1 - \frac{\epsilon^2\lambda_0}{\rho}$ we see that

$$\begin{aligned}
& r^k me^{\alpha\lambda_0} < e^{\alpha\lambda_0/2} = \Phi_{\text{lower bound}} \\
& \iff k \log r + \log m + \alpha\lambda_0 < \frac{\alpha\lambda_0}{2} \\
& \iff \log m + \frac{\alpha\lambda_0}{2} < -k \log r \leq Ckr = Ck\frac{\epsilon^2\lambda_0}{\rho} \\
& \iff k > \frac{\rho \log m}{\epsilon^2\lambda_0} + \alpha\frac{\rho}{2\epsilon^2} = \frac{\rho \log m}{\epsilon^2\lambda_0} + 2\frac{1}{\epsilon^3\lambda_0}\rho \log(\frac{2m}{\epsilon})
\end{aligned}$$

where constant $C > 0$ exists because $-\log(1-x) \approx x$ for small x (more precisely, $-\log(1-x) = O(x)$ when $0 < x << 1$), which can readily be shown by Taylor expanding. Therefore, since the second term $2\frac{1}{\epsilon^3\lambda_0}\rho\log(\frac{2m}{\epsilon})$ dominates $\frac{\rho \log m}{\epsilon^2\lambda_0}$, Improve-Packing must terminate after $k = O(\epsilon^{-3}\lambda_0^{-1}\rho\log(m\epsilon^{-1}))$ iterations.

In the case where the input is $O(\epsilon)$-optimal we have must have $\lambda_x \geq \lambda_0/(1 + O(\epsilon))$, for otherwise $\lambda_x < \lambda_0/(1 + O(\epsilon)) \leq \lambda^*$ which cannot happen. Using this tighter bound we can improve the lower bound on the potential function throughout the execution as so

$$e^{\alpha(1+O(\epsilon))^{-1}\lambda_0} \leq \Phi(x)$$

And from there a similar proof follows.

□

6 The Algorithm

Using the Improve-Packing subroutine, we now present an algorithm that finds an ϵ_0-approximate solution, for any desired tolerance $0 \leq \epsilon_0 < 1$, or concludes that no exact solution exists and tries again on the relaxed problem $Ax \leq b'$ with $b' = 2b$.

The algorithm first attempts to find a 1-optimal solution by taking an arbitrary $x \in P$ and repeatedly calling Improving-Packing with tolerance $1/6$ and the output of each previous call. This procedure terminates either when $\lambda_x \leq 2$ or when output x and its dual solution, y_x, satisfy lemma 3.2. If the latter termination condition holds but not the former, then, since x is 1-optimal, we have $2 < \lambda_x \leq 2\lambda^*$ which implies $\lambda^* > 1$. Hence in this case no exact solution to the packing problem exists, so we relax the problem by doubling b and try again.

The next phase of the algorithm, denoted the ϵ-scaling phase, repeatedly halves the tolerance and calls Improve-Packing until an ϵ_0 approximate solution is found or it is shown that none exist. Throughout the ϵ-scaling phase every call to Improve-Packing returns a 6ϵ-optimal solution. This is because λ_x will not be reduced by more than 2 since we start this phase with a 1-optimal solution, i.e. the termination condition $\lambda_x < \lambda_0/2$ cannot occur. So after each call, if $\lambda_x > 1 + 6\epsilon$ then, using the same reasoning as above, no exact solution to the packing problem exists, so we relax the problem by doubling b and try again. Finally, if the algorithm makes it to the end of the ϵ-phase with $\epsilon \leq \epsilon_0$, then $\lambda_x \leq 1 + 6\epsilon \leq 1 + \epsilon_0$ which implies that x is an ϵ_0-approximate solution.

Algorithm 2 Solve-Packing(ϵ_0)

```
Initialize ϵ ← 1/6;  x ∈ P arbitrary
While λ_x > 2 and x, y_x do not satisfy (P2)
  x ← Improve-Packing(x, ϵ)
If λ_x > 2
  Return None; call Solve-Packing(ϵ_0) with b ← 2b
While TRUE
  ϵ ← ϵ/2
  x ← Improve-Packing(x, ϵ)
  If λ_x > 1 + 6ϵ
    Return None; call Solve-Packing(ϵ_0) with b ← 2b
  If ϵ ≤ ϵ_0
    Return x
```

The next result guarantees the finite running time of the Solve-Packing algorithm.

Theorem 6.1. *The Solve-Packing procedure terminates after $O(\epsilon_0^{-2}\rho\log(m\epsilon_0^{-1}))$ iterations of the Improve-Packing subroutine.*

Proof. First consider the first phase of the algorithm (before the ϵ-scaling phase). By the first part of theorem 5.1 the number of iterations during a call to Improve-Packing with tolerance 1 is $O(\lambda_0^{-1}\rho\log(m)$. Observe that this bound is proportional to λ_0^{-1} and at least doubles each call. (The second observation holds because only the last call terminates with x and y satisfying $(P2)$. This is because, in this case, we would then have a 1-optimal solution and the algorithm would move to the next phase. Every other call reduces the next λ_0 by half, and so λ_0^{-1} at least doubles.) Thus the number of iterates during the final call to Improve-Packing dominates the total of the previous calls. Hence the number Improve-Packing iterations during this phase is $O(\rho\log(m))$.

Now consider the ϵ-scaling phase of the algorithm. For each ϵ-phase, the input is a 12ϵ-optimal solution. So by the second part of theorem 5.1 the number of iterations of Improve-Packing needed to convert the input into a 6ϵ-optimal solution is $O(\epsilon^{-2}\rho\log(m\epsilon^{-1}))$, since λ_0^{-1} is a constant. Observe that this bound is proportional to the current value ϵ^{-2} and more than doubles each phase since ϵ is halved. Thus the bound for the final ϵ- phase dominates the total for all other phases. Now since $\epsilon > \epsilon_0$ at termination, the number Improve-Packing iterations during the ϵ-scaling phase is $O(\epsilon_0^{-2}\rho\log(m\epsilon_0^{-1}))$. $\square$

7 References

Plotkin, Serge A., et al. "Fast Approximation Algorithms for Fractional Packing and Covering Problems." Mathematics of Operations Research, vol. 20, no. 2, 1995, pp. 257–301.