3-26-1999

Greedy Approximation Algorithms for K-Medians by Randomized Rounding

Neal E. Young
Dartmouth College

Greedy Approximation Algorithms for K-Medians by Randomized Rounding

Neal E. Young*

Technical Report PCS-TR99-344

Department of Computer Science
Dartmouth College
Hanover, NH 03755-3510
ney@cs.dartmouth.edu

March 26, 1999

Abstract

We give an improved approximation algorithm for the general k-medians problem. Given any $\epsilon > 0$, the algorithm finds a solution of total distance at most $D(1+\epsilon)$ using at most $k \ln(n + n/\epsilon)$ medians (a.k.a. sites), provided some solution of total distance D using k medians exists. This improves over the best previous bound (w.r.t. the number of medians) by a factor of $\Omega(1/\epsilon)$ provided $1/\epsilon = n^{O(1)}$. The algorithm is a greedy algorithm, derived using the method of oblivious randomized rounding. It requires at most $k \ln(n+n/\epsilon)$ linear-time iterations. We also derive algorithms for fractional and weighted variants of the problem.

*Research partially funded by NSF CAREER award CCR-9720664.

1 Introduction

The input for the k-medians problem is a collection of *sites*, a collection of n *elements*, and for each site s and element e, a non-negative *distance* $\mathrm{dist}(s,e)$. A solution is a set S of *size* at most k and an assignment of elements to the chosen sites so as to minimize the total of the *distances* between each element and its assigned site.

By a k-medians *decision* problem, we mean a problem instance where an additional parameter $D \geq 0$ is given, and the problem is to decide whether a solution of size at most k and total distance at most D exists. By a *distance-constrained* medians problem, we mean a problem instance where D is given but not k, and the goal is to find a solution minimizing the size subject to the constraint that the total distance is at most D.

Fractional Variants. The *fractional* k-medians problem is that of solving the following linear program, which we call the *k-medians LP* [1].

$$\text{minimize}_x \ D \qquad (1)$$

$$\text{subject to} \begin{cases} \sum_s x(s) & = & k & \\ \sum_{se} x(s,e)\mathrm{dist}(s,e) & = & D & \\ \sum_s x(s,e) & \geq & 1 & (\forall e) \\ x(s,e) & \leq & x(s) & (\forall s,e) \\ x(s,e) & \geq & 0 & (\forall s,e) \end{cases}$$

Above each $\mathrm{dist}(s,e)$ (for each e and s) is given, as is k. The variable $x(s)$ indicates whether (or rather the extent to which) site s is chosen, while variable $x(s,e)$ indicates whether element e uses site s. To "solve the LP" means to assign values to the variables so that the constraints are met and the objective function (in this case D) is minimized.

By a *fractional* solution to the k-medians problem, we mean an x that is a feasible solution to the above LP for some k and D. We define the *(total) distance* of x to be $\mathrm{dist}(x) \doteq \sum_{se} x(s,e)\,\mathrm{dist}(s,e)$ and the *size* of x to be $\sum_s x(s)$. If x takes on only values in $\{0,1\}$ we call x an *integer* solution. The linear program is a relaxation of the k-medians problem, meaning that the solutions to the k-medians problem correspond to the integer solutions of the LP and the correspondence preserves total distance and size.

Weighted Variants. By the *weighted* k-medians problem, we mean the generalization of the k-medians problem in which each site s has a specified *size* $\text{size}(s) \geq 0$ and we redefine the size of a set S of sites to be $\text{size}(S) \doteq \sum_{s \in S} \text{size}(s)$. The goal is then to find a minimum-distance solution among all of the solutions of size at most k. The *fractional* weighted k-medians problem is defined by the "weighted" k-medians LP, which is the same as the k-medians LP except that the constraint "$\sum_s x(s) = k$" is replaced by "$\sum_s x(s)\text{size}(s) = k$". We define the weighted k-medians decision problem and the distance-constrained weighted k-medians analogously to their unweighted counterparts. In all cases, if each $\text{size}(s) = 1$ the weighted problem reduces to the unweighted problem.

Approximate Solutions. The k-medians problem is NP-complete (if each $\text{dist}(s, e) \in \{0, \infty\}$ we have the set cover problem), so we study approximation algorithms. In contrast the fractional k-medians problem is solvable in polynomial time by linear programming. Nonetheless for this problem we also study approximation algorithms, with the hope that they are faster.

For either type of problem, by an $(\alpha(D), \beta(k))$-approximation algorithm, we mean an algorithm that, given an instance for which there exists some fractional solution of total distance D and size at most k, produces a solution (fractional or not, as appropriate) of total distance at most $\alpha(D)$ and size at most $\beta(k)$.

Results. We derive greedy approximation algorithm for the basic unweighted k-medians problem (our main result). We prove a better performance guarantee than was previously known for the problem. We also give algorithms for the weighted and fractional variants of k-medians. The results are summarized in Figure 1.

Method. A basic goal of this paper is to try to use probabilistic methods [2], including randomized rounding [13] and the method of conditional expectations [12], to understand common principles underlying diverse "greedy" [8, 10, 5] and Lagrangian-relaxation approximation algorithms [11].

To this end we use a variant of randomized rounding called *oblivious* randomized rounding [15]. The basic technique is that we start with a randomized rounding scheme (a random experiment for converting a fractional solution x^* to an integer (or, as we'll see later, fractional!) solution $\tilde{x}$ that approximates x^*). We analyze the rounding scheme using probabilistic methods to show a performance guarantee. We then use the method of con-

k-medians variant	**iterations**	**distance**	**size**
distance constrained	$k\ln(n+n/\epsilon)$	$D\,(1+\epsilon)$	$\lceil k\ \ln(n+n/\epsilon)k\rceil$
weighted dist. constr.	m	$D\,(1+\epsilon)$	$k\,[1+\ln(n+n/\epsilon)]$
fractional dec. prob.	$\frac{k\ln(n+n/\epsilon)}{\Theta(\min\{1,\epsilon^2\})}$	$(1+\epsilon)^2D$	$(1+\epsilon)k$

Figure 1: Summary of results. "Size" and "distance" refer to the performance guarantee — for instance the first algorithm returns a solution of total distance at most $D\,(1+\epsilon)$ and size at most $\lceil\ln(n+n/\epsilon)k\rceil$, provided there exists *some* solution of total distance D and size k. The number of elements and sites are n and m, respectively. Each iteration can be implemented in linear time. The algorithm for the weighted problem always terminates within m iterations, but if each $\text{size}(s)\geq 1$, the number of iterations is at most $k\ln(n+n/\epsilon)$.

ditional expectations to derive an algorithm that has the same performance guarantee.

The unusual aspect of oblivious randomized rounding is that we set up the initial rounding scheme so that the final algorithm can be implemented *without being given or computing the original fractional solution.* This trick enables is to use probabilistic methods to derive both greedy algorithms for NP-hard problems and Lagrangian-relaxation algorithms for fractional problems. (Without this trick, the resulting algorithms would have to start by solving the LP directly by some other method.)

We also discuss how to solve the fractional k-medians problem using the general packing/covering algorithm due to Plotkin, Shmoys, and Tardos [11]. We show that this solves the problem as claimed provided each $\text{dist}(s,e)=O(D)$ — a reasonable assumption if the problem being solved is in fact being interpreted as a relaxation of some integer k-medians problem. Technically, the direct algorithm we give is different in the way it handles the distance constraint; this is why it does not suffer from this restriction.

Previous Work. Lin and Vitter gave a polynomial-time $((1+\epsilon)D,(1+1/\epsilon)(\ln n+1))$-approximation algorithm for the k-medians problem [9]. The guarantee presented here improves on that guarantee (with respect to the size) by a factor of $\Theta(1/\epsilon)$ for any $\epsilon\geq n^{-O(1)}$.

The work here on the fractional k-medians problem is motivated by recent attention given to the *metric* k-medians problem, in which the distance function is assumed to satisfy the triangle inequality. Charikar, Guha, Tardos and Shmoys [4] recently gave a polynomial-time $(O(D),k)$-approximation

input: *sites, n elements,* dist*, ϵ, frac. soln. x^* of size k and dist. D.*
output: *integer solution $\tilde{x}$ that with positive probability approximates x^**

1. For each e, s do $\tilde{x}(s) := 0; \tilde{x}(s,e) := 0$.
2. $T := \lceil k \ln(n + n/\epsilon) \rceil$
3. For $t := 1$ to T do:
4. Choose a single site s at random so that $\Pr(s \text{ chosen}) = x^*(s)/k$.
5. Set $\tilde{x}(s) := 1$.
6. For each element e independently with probability $x^*(s,e)/x^*(s)$:
7. Choose e with s: set $\tilde{x}(s,e) := 1$ and $\tilde{x}(s',e) := 0$ for $s' \neq s$.
8. Return $\tilde{x}$.

Figure 2: K-medians rounding scheme. The rounding scheme, given a fractional solution x^*, repeatedly "samples" at random from x^* enough times to get an integer solution $\tilde{x}$ that, with positive probability, is a good enough approximation to x^*. We use similar rounding schemes to derive algorithms for the weighted and fractional variants of k-medians.

algorithm. Recently polynomial-time $O(1)$-approximation algorithms have also been shown for the closely related (but easier) *uncapacitated facilities-location* problem [6, 14]. Most of these recent approximation algorithms require first computing a solution to the *fractional* problem, and then non-trivially rounding that solution. This motivates us to study algorithms for the fractional problem. We plan to describe algorithms for the fractional weighted set-cover problem and the fractional facilities-location problem in a separate paper.

It is not clear how to reduce the weighted k-medians problem (in contrast to facilities location [7]) to the weighted set-cover problem. Conversely, the weighted k-medians problem generalizes the weighted set-cover problem (by taking each $\text{dist}(s,e) \in \{0, \infty\}$, $D = 0$, and any $\epsilon > 0$). In fact the algorithm presented here generalizes the traditional greedy weighted set-cover algorithm [8, 10, 5].

2 Greedy K-Medians Algorithm

In this section we derive and analyse a greedy $((1+\epsilon)D, \ln(n + n/\epsilon)k)$-approximation algorithm for the integer k-medians problem. The algorithm is based on the rounding scheme shown in Figure 2. In each iteration, the rounding scheme chooses a single random site s. The probability of choosing s is $x^*(s)/k$, where x^* is a given fractional solution. Then for each element

e independently, the rounding scheme chooses e with s with probability $x^*(e,s)/x^*(s)$. After $\ln(n+n/\epsilon)k$ iterations, it returns the chosen sites, with each element assigned to the site it was most recently chosen with.

In the derivation of the algorithm, we assume for notational convenience that $D > 0$. With appropriate notation for the special cases, the derivation can be made correct for the case $D = 0$, but we take the less combersome approach of reconsidering the special case $D = 0$ after the derivation of the algorithm and arguing directly that this case is correctly handled.

Guarantee 1 *Let $\tilde{x}$ be the output of the k-medians rounding scheme. Then with positive probability $\tilde{x}$ has* $\mathrm{size}(\tilde{x}) \leq \lceil k\ln(n+n/\epsilon)\rceil$ *and* $\mathrm{dist}(\tilde{x})/D < 1+\epsilon$.

Proof: The bound on the size always holds, because each iteration adds at most one new site. It remains to show that with positive probability all elements are chosen with some site and the total distance is at most $(1+\epsilon)\,D$.

Say that an element e is *chosen* in an iteration if e is chosen with some site s in that iteration. Let $\#\mathrm{unc}(x)$ denote the number of elements not yet chosen, i.e. the number of elements e such that $\forall s\, x(s,e) = 0$. Observe the following basic facts about each iteration:

1. The probability of e being chosen with a particular s is
$$x^*(s)k^{-1}x^*(s,e)/x^*(s) = x^*(s,e)/k.$$
2. The probability of e being chosen with any site is $\sum_s x^*(s,e)/k = 1/k$.
3. *Given* that e is chosen, the probability of it being chosen with s is $x^*(s,e)$.

Based on these facts we make the following two observations about the final solution $\tilde{x}$:
$$E[\#\mathrm{unc}(\tilde{x})] \;=\; n(1-1/k)^T \;<\; n\exp(-T/k)$$
(here we use $1+z < \exp(z)$ for $z \neq 0$) and
$$\begin{aligned} E[\mathrm{dist}(\tilde{x})] &= \textstyle\sum_e [1-(1-1/k)^T]\sum_s x^*(s,e)\mathrm{dist}(s,e) \\ &= [1-(1-1/k)^T]D \\ &\leq D. \end{aligned}$$

Now, by the "naive union" and Markov bounds (Lemmas 6 and 7 in the Appendix), we have
$$\Pr[\#\mathrm{unc}(\tilde{x}) \geq 1 \textbf{ or } \mathrm{dist}(\tilde{x})/D \geq 1+\epsilon] \;<\; n\exp(-T/k) + 1/(1+\epsilon).$$

By the choice of T the right-hand side is at most 1. This proves the guarantee. $\square$

The quantity of interest. Define random variable $\tilde{x}_t$ to be the value of $\tilde{x}$ after the tth iteration of the rounding scheme. Recall that T is the number of iterations of the rounding scheme.

The following *quantity of interest* is implicit in the analysis of the rounding scheme:

$$\mathcal{Q} \doteq \#\mathrm{unc}(\tilde{x}_T) + \frac{\mathrm{dist}(\tilde{x}_T)/D}{1+\epsilon}. \tag{2}$$

That analysis hinges on the random variable $\mathcal{Q}$ in the sense that it shows

i. If $\mathcal{Q} < 1$, then $\#\mathrm{unc}(\tilde{x}_T) = 0$ and $\mathrm{dist}(\tilde{x}_T)/D < 1+\epsilon$

ii. $E[\mathcal{Q}] < 1$.

As the rounding scheme proceeds, the expectation of $\mathcal{Q}$, *conditioned on the choices made so far* may rise and fall. We identify the state of the rounding scheme as it proceeds by a tuple (t, x), where t is the most recent iteration and x is the value of $\tilde{x}$ at the end of that iteration. Given a configuration (t, x), define $\Phi(t, x)$ to be the expectation of $\mathcal{Q}$ given that the random process goes through configuration (t, x). That is, $\Phi(t, x) \doteq E[\mathcal{Q} \mid \tilde{x}_t = x]$. In terms of Φ, facts (i) and (ii) above can be restated

i. For any x such that $\Phi(T, x) < 1$, the performance guarantee holds.

ii. $\Phi(0, \tilde{x}_0) < 1$.

Abstract explanation of the method of conditional expectations. We are going to apply the method of conditional expectations to the k-medians random rounding scheme, with respect to the quantity of interest $\mathcal{Q}$ defined above. Abstractly, what this entails is modifying the rounding scheme so that if in the tth iteration it starts in some configuration $(t-1, x)$, rather than choosing a *random* site s and set of elements S to arrive at the next configuration (t, x_R), we have it choose the site s and set of elements S *deterministically* so as to arrive at some particular configuration (t, x_D). The choice of x_D must satisfy the following *essential property*:

$$\Phi(t, x_D) \leq \Phi(t-1, x).$$

One "standard" way to choose x_D so that this property is satisfied is to choose x_D to *minimize* the resulting value of $\Phi(t, x_D)$. Then the essential property holds because

$$\Phi(t-1, x) = E_{x_R}[\Phi(t, x_R)] \geq \min_{x_R} \Phi(t, x_R) = \Phi(t, x_D),$$

where (t, x_R) ranges over the configurations reachable in one step from $(t-1, x)$. We use this standard choice in deriving two of the algorithms in this paper.

The modified process is then deterministic, that is, on any particular input, it goes through a fixed sequence of configurations that we denote

$$(0, \hat{x}_0) \mapsto (1, \hat{x}_1) \mapsto \cdots \mapsto (T, \hat{x}_T).$$

We call the modified process "the algorithm" to distinguish it from the rounding scheme. By the essential property,

$$1 > \Phi(0, \hat{x}_0) \geq \Phi(1, \hat{x}_1) \geq \cdots \geq \Phi(T, \hat{x}_T).$$

This implies that the algorithm is guaranteed to produce an outcome $\hat{x}_T$ such that $\Phi(T, \hat{x}_T) < 1$, i.e. an x meeting the performance guarantee.

Deriving the algorithm. Having described the essentials of the method abstractly, we now demonstrate it concretely by applying it to the k-medians rounding scheme. The first important question is how to choose s and S to minimize Φ. To answer this question we take a close look at $\Phi(t, x)$. Recall that $\Phi(t, x) \doteq E[\mathcal{Q} \mid \tilde{x}_t = x]$ and

$$\mathcal{Q} \doteq \#\text{unc}(\tilde{x}_T) + \frac{\text{dist}(\tilde{x}_T)/D}{1+\epsilon}.$$

Conditioned on $\tilde{x}_t = x$, the expected number of elements left unchosen by the Tth iteration is the number left unchosen (in x) times $(1-1/k)^{T-t}$. This takes care of the first term in Φ.

Conditioned on $\tilde{x}_t = x$, what is the expectation of $\text{dist}(\tilde{x}_T)$? For any s and e, the conditional expectation of $\tilde{x}_T(s, e)$ is

$$(1-1/k)^{T-t} x(s, e) + [1 - (1-1/k)^{T-t}] x^*(s, e),$$

because the chance that e is not chosen in any of the $T-t$ remaining rounds is $(1-1/k)^{T-t}$, whereas if it is, the probability that it is finally assigned to

s is $x^*(s,e)$. Thus, the conditional expectation of $\mathrm{dist}(\tilde{x}_T)$ is

$$
\begin{aligned}
&\sum_{se} E[\tilde{x}_T(s,e) \mid \tilde{x}_t = x]\mathrm{dist}(s,e) \\
&\quad = \sum_{se} \Big((1-1/k)^{T-t}x(s,e) + [1-(1-1/k)^{T-t}]x^*(s,e)\Big)\,\mathrm{dist}(s,e) \\
&\quad = (1-1/k)^{T-t}\mathrm{dist}(x_t) + [1-(1-1/k)^{T-t}]D.
\end{aligned}
$$

This takes care of the second term, giving

$$
\begin{aligned}
\Phi(t,x) &= \#\mathrm{unc}(x) \times (1-1/k)^{T-t} \\
&\quad + \frac{\mathrm{dist}(x)/D}{1+\epsilon}(1-1/k)^{T-t} + \frac{1-(1-1/k)^{T-t}}{1+\epsilon}.
\end{aligned} \tag{3}
$$

At this point we can see that $\Phi(t,x)$ is independent of x^*, except for the parameter D. This independence is a consequence of the careful design of the original rounding scheme, and is what will enable us to implement the algorithm without computing x^*.

Recall that $(t-1, \hat{x}_{t-1})$ denotes the configuration of the algorithm before the tth iteration. Given that $\hat{x}_{t-1}$ is determined, choosing s and S to minimize $\Phi(t,\hat{x}_t)$ is the same as choosing them to minimize

$$
\begin{aligned}
&[\Phi(t,\hat{x}_t) - \Phi(t,\hat{x}_{t-1})]/(1-1/k)^{T-t} \\
&\quad = \#\mathrm{unc}(\hat{x}_t) - \#\mathrm{unc}(\hat{x}_{t-1}) + \frac{\mathrm{dist}(\hat{x}_t)/D}{1+\epsilon} - \frac{\mathrm{dist}(\hat{x}_{t-1})/D}{1+\epsilon} \\
&\quad = \sum_{e\in S} \frac{\mathrm{dist}(s,e)/D}{1+\epsilon} - \begin{cases} \frac{\mathrm{dist}(s'(e),e)/D}{1+\epsilon} & \text{if } \exists s'(e) \text{ s.t. } \hat{x}_{t-1}(s'(e),e)=1 \\ 1 & \text{otherwise.} \end{cases}
\end{aligned}
$$

For any particular site choice s, it is easy to find a set S minimizing the above expression — namely, the set $S(s)$ containing those elements e such that $\mathrm{dist}(s,e)/(D(1+\epsilon))$ is either less than 1 (if e was not previously chosen) or less than $\mathrm{dist}(s',e)/(D(1+\epsilon))$ where s' is the site it was most recently chosen with. Thus, to find the best s and S, it suffices for the algorithm to enumerate each possible site s with the single corresponding set $S(s)$. This gives us enough information to implement the algorithm efficiently.

The algorithm is shown in Figure 3. For convenience it maintains variables d_e so that $d_e = \mathrm{dist}(s',e)/(D/(1+\epsilon))$ if e is currently assigned to some site s', and otherwise $d_e = 1$. Strictly speaking, the termination condition differs from the one in the algorithm obtained by applying the method of conditional probabilities, but we show below that the modified termination condition suffices.

input: *sites, n elements,* dist, ϵ, D.
output: *integer solution $\hat{x}$ that approximates x^**

1. Let $\hat{x}(s) := 0$ and $\hat{x}(s,e) := 0$ for each s, e. Let $d_e := 1$.
2. Define $\text{dist}'(s,e) = \frac{\text{dist}(s,e)/D}{1+\epsilon}$, where $0/0 \doteq 0$ in case $D = 0$.
3. Until $\sum_e d_e < 1$ do:
4. Choose a single site s to minimize $\sum_e \min\{0, \text{dist}'(s,e) - d_e\}$.
5. For each e such that $\text{dist}'(e,s) < d_e$ do:
6. Set $d_e := \text{dist}'(e,s)$.
7. Set $\hat{x}(s,e) := 1$ and $\hat{x}(s',e) := 0$ for each $s' \neq s$.
8. Return $\hat{x}$.

Figure 3: The greedy k-medians algorithm. The algorithm depends on D but not k. The algorithm terminates within $T \doteq \lceil k \ln(n+n\epsilon) \rceil$ linear-time iterations and returns a solution of distance $D(1+\epsilon)$ and size $\lceil k \ln(n+n\epsilon) \rceil$, provided some feasible fractional solution of distance D and size k exists.

Note that the algorithm does not compute a solution to the fractional problem first. Also, it depends on D but not k.

Guarantee 2 *Fix an instance of the k-medians problem having a fractional solution of cost D and size k. Let $T \doteq \lceil k \ln(n + n/\epsilon) \rceil$.*

Given the instance, D, and ϵ, the greedy k-medians algorithm returns a solution $\hat{x}$ such that $\text{dist}(\hat{x})/D \leq 1 + \epsilon$ *and* $\text{size}(\hat{x}) \leq T$.

The algorithm requires T iterations, each of which can be implemented in linear time.

Proof: With the exception of the termination condition "$\sum_e d_e < 1$", the algorithm comes from the rounding scheme by applying the method of conditional expectations with respect to $\mathcal{Q}$. This means that it maintains the invariant $\Phi(t, \hat{x}) < 1$ after each iteration.

Let $T = \lceil k \ln(n + n/\epsilon) \rceil$ as in the rounding scheme. Since the size of $\hat{x}$ is at most the number of iterations, and the algorithm terminates iff all elements have been chosen and $\text{dist}(\hat{x}) < D(1+\epsilon)$, we know that if the algorithm terminates in T or fewer iterations, then the guarantee we are proving holds.

Suppose the algorithm reaches the Tth iteration. Then $\Phi(T, \hat{x}_T) < 1$. But by inspection of Φ, this implies that all elements must have been chosen and $\text{dist}(\hat{x}_T)/D < 1 + \epsilon$. Thus in this case the algorithm terminates after the Tth iteration.

This proves the guarantee assuming $D > 0$. In the special case $D = 0$, define $D' > 0$ so that D' is smaller than the smallest non-zero $\text{dist}(s,e)/(1+$

input: *sites, n elements,* dist, size, ϵ, *frac. soln.* x^* *of size* k *and dist.* D.
output: *integer solution* $\tilde{x}$ *that with positive probability approximates* x^*

1. For each e, s do $\tilde{x}(s) := 0; \tilde{x}(s,e) := 0$.
2. Until $\text{size}(\tilde{x}) \geq k\lceil \ln(n + n/\epsilon) \rceil$
3. **or** all elements have been chosen and $\text{dist}(\tilde{x}) < D(1+\epsilon)$:
4. Choose a single site s at random so that $\Pr(s \text{ chosen}) = x^*(s)/|x^*|$.
5. Set $\tilde{x}(s) := 1$.
6. For each element e independently with probability $x^*(s,e)/x^*(s)$ do:
7. Choose e with s: set $\tilde{x}(s,e) := 1$ and $\tilde{x}(s',e) := 0$ for $s' \neq s$.
8. Return $\tilde{x}$.

Figure 4: Weighted k-medians rounding scheme. The rounding scheme is the same as for the unweighted problem, except for the termination condition. Note $|x| \doteq \sum_s x(s)$.

ϵ). Running the algorithm with $D = D'$ gives $\text{dist}(\hat{x}) < (1+\epsilon)D'$, which by the choice of D' implies $\text{dist}(\hat{x}) = 0$. Running the algorithm with $D = 0$, it is easy to verify the algorithm does the same thing, because in either case the algorithm never assigns an element e to any site s such that $\text{dist}(s,e) > D(1+\epsilon)$. □

With appropriate assumptions about the representation of the input, the algorithm can be implemented so that each of the $k\ln(n + n/\epsilon)$ iterations takes an amount of time that is linear in the number of sites plus the number of pairs (s,e) such that $\text{dist}(s,e)/D < 1+\epsilon$. In some cases (e.g. if the bipartite graph representing the (site,element) relationships is of bounded degree, or if the distances are all small integers), faster implementations using standard data structures may be possible.

Note that in general all that is required to implement the algorithm is a subroutine that, given a set of sites C (the currently chosen sites), returns the site s minimizing $\sum_e \min\{1+\epsilon, \text{dist}(C,e)/D, \text{dist}(s,e)/D\}$. This may be useful in cases where the sites and/or elements are not concretely given, but instead are implicitly defined in some manner.

A natural question is how to adapt the algorithm to the case when k is known in advance, rather than D. I do not know how.

3 Weighted K-Medians Algorithm

In this section we derive and analyse a greedy $((1+\epsilon)D, [1+\ln(n+n/\epsilon)]k)$-approximation algorithm for the *weighted* integer k-medians problem. The algorithm is based on the rounding scheme shown in Figure 4, which is the same as the rounding scheme for the unweighted problem except that instead of terminating after a fixed number of rounds, it terminates when the total size of the chosen sites exceeds a threshold.

A goal of this section is to study the issues that arise when generalizing a rounding scheme for an unweighted problem to the weighted version. Generally, this entails modifying a rounding scheme that has a fixed number of rounds so that the number of rounds becomes a random variable. This complicates the probabilistic analysis, but the following lemma seems to capture what we need.

Lemma 1 *Let $X_0, X_1, X_2, \ldots$ be a sequence of random variables and let $T \in \mathbb{N}_+$ be a random variable with $E[T] < \infty$. Suppose there exist $\Delta \in \mathbb{R}$ such that for all $t \geq 1$*

$$E(X_t - X_{t-1} \,|\, T \geq t) \leq \Delta.$$

Then provided $X_t - X_{t-1} < c$ for some constant c and all $t \leq T$,

$$E(X_T - X_0) \leq \Delta E(T).$$

This lemma is a one-sided version of Wald's equation [3, p. 370]. We give the proof in the Appendix.

Analysis. Define $|x| \doteq \sum_s x(s)$. In the previous derivations, we generally had $|x^*| = k$ but this no longer holds in general. As before we assume $D > 0$ and consider the case $D = 0$ at the end.

Guarantee 3 *Let $\tilde{x}$ be the output of the weighted k-medians rounding scheme. Then with positive probability $\tilde{x}$ satisfies $\mathrm{size}(\tilde{x}) < k\ln(n+n/\epsilon)+\max_s \mathrm{size}(s)$ and $\mathrm{dist}(\tilde{x})/D < 1+\epsilon$.*

Note that no integer solution of size k uses any s with $\mathrm{size}(s) > k$, so if our goal is an approximately optimal integer solution, we can assume w.l.o.g. that $\mathrm{size}(s) \leq k$ for all s.

Proof: Let random variable T denote the number of iterations of the rounding scheme. Let $\tilde{x}_t$ denote the value of $\tilde{x}$ after the tth iteration of the rounding scheme.

Claim 1: $E[\text{size}(\tilde{x}_t \mid \tilde{x}_{t-1})] \le \text{size}(x_{t-1}) + k/|x^*|$. This is because the expected increase in $\text{size}(\tilde{x})$ in any iteration is at most

$$\sum_s \frac{x^*(s)}{|x^*|}\text{size}(s) = \frac{\text{size}(x^*)}{|x^*|} = k/|x^*|.$$

Define[1] $d(x) \doteq \#\text{unc}(x) + \dfrac{\text{dist}(x)/D}{1+\epsilon} - \dfrac{1}{1+\epsilon}$. Note $d(x_t) > 0$ for $t < T$.

Claim 2: $E[\ln d(\tilde{x}_t) \mid x_{t-1}] \le \ln d(\tilde{x}_{t-1}) - 1/|x^*|$. This is because, as in the analysis on page 6 of the unweighted rounding scheme, a calculation shows that

$$E[\#\text{unc}(\tilde{x}_t) \mid \tilde{x}_{t-1}] = (1 - 1/|x^*|)\#\text{unc}(\tilde{x}_{t-1})$$

and

$$E[\text{dist}(\tilde{x}_t) \mid \tilde{x}_{t-1}] = (1 - 1/|x^*|)\text{dist}(\tilde{x}_{t-1}) + D/|x^*|.$$

From these another calculation shows $E[d(\tilde{x}_t) \mid x_{t-1}] = (1-1/|x^*|)d(\tilde{x}_{t-1})$. Thus, using $\ln(z) \le z - 1$,

$$\begin{aligned} E[\ln d(\tilde{x}_t) \mid \tilde{x}_{t-1}] &= \ln d(\tilde{x}_{t-1}) + E\left[\ln \frac{d(\tilde{x}_t)}{d(\tilde{x}_{t-1})} \;\middle|\; \tilde{x}_{t-1}\right] \\ &\le \ln d(\tilde{x}_{t-1}) + E\left[\frac{d(\tilde{x}_t)}{d(\tilde{x}_{t-1})} - 1 \;\middle|\; \tilde{x}_{t-1}\right] \\ &= \ln d(\tilde{x}_{t-1}) - 1/|x^*|. \end{aligned}$$

Combining the two claims we get

$$E[\ln d(\tilde{x}_t) \;+\; \text{size}(\tilde{x}_t)/k \mid x_{t-1}] \le \ln d(\tilde{x}_{t-1}) \;+\; \text{size}(\tilde{x}_{t-1})/k$$

Thus, by Lemma 1,

$$E[\ln d(\tilde{x}_T) \;+\; \text{size}(\tilde{x}_T)/k] \le \ln d(\tilde{x}_0) \;+\; \text{size}(x_0) \;<\; \ln n. \tag{4}$$

Thus, with positive probability, $\ln d(\tilde{x}_T) + \text{size}(\tilde{x}_T)/k < \ln(n)$. Assume this event occurs.

[1] The definition of d is motivated by the proof of Guarantee 1. Namely, $d(x)$ decreases by a constant factor $(1 - 1/|x|)$ in expectation each iteration, and if $d(x) < 1 - 1/(1+\epsilon)$, then x covers all elements and $\text{dist}(x)/D < 1 + \epsilon$.

The subsequent use of $\ln d()$ is motivated by the unfortunate fact that since T is a random variable, we only know how to compute expectations (at termination) of quantities that decrease or increase by some fixed additive amount in each round.

If the rounding scheme terminates because all elements are covered and $\mathrm{dist}(\tilde{x}) < D(1+\epsilon)$, then clearly the performance guarantee holds.

Otherwise the algorithm terminates because $\mathrm{size}(\tilde{x}) \geq k\ln(n+n/\epsilon)$. This lower bound on the size and the occurence of the event "$\ln d(\tilde{x}_T) + \mathrm{size}(\tilde{x}_T)/k < \ln n$" ensure that $d(\tilde{x}_T) < 1/(1+1/\epsilon) = 1 - 1/(1+\epsilon)$. But by inspection of d, this means that $\#\mathrm{unc}(\tilde{x}_T) < 1$ and $\mathrm{dist}(\tilde{x}_T) < D(1+\epsilon)$.

Also, $\mathrm{size}(\tilde{x}_T) \leq k\ln(n+n/\epsilon) + \max_s \mathrm{size}(s)$, because the last round increases the size of $\tilde{x}$ by at most $\max_s \mathrm{size}(s)$. $\square$

Algorithm Derivation and Analysis. In this derivation we use *pessimistic estimators* — upper bounds on the conditional expectation that take the place of the true conditional expectation. Also, we do not use the "standard" method of *minimizing* the conditional expectation (or rather, pessimistic estimator for it). Rather, we find another way to keep it from increasing.

Let $\tilde{x}_t$, T, K, and $d()$ be defined as in the rounding scheme and proof. The quantity of interest in that proof is the random variable

$$\mathcal{Q} \doteq \ln d(\tilde{x}_T) \tag{5}$$

That proof showed that $E[\mathcal{Q}] < \ln \epsilon/(1+\epsilon)$, and that if $\mathcal{Q} < \ln\epsilon/(1+\epsilon)$ then $\tilde{x}_T$ meets the performance guarantee. We can easily easily generalize the proofs of the two claims in the analysis of the rounding scheme to show:

Claim 1: $E[T-t \,|\, \tilde{x}_t = x] \geq |x^*|[K - \mathrm{size}(x)]/k$.

Claim 2: $E[\ln d(\tilde{x}_T) \,|\, \tilde{x}_t = x] \leq \ln d(x) - E[T-t]/|x^*|$.

We omit the proofs because we verify the algorithm independently below. Combining the two claims we get

$$E[\ln d(\tilde{x}_T) \,|\, \tilde{x}_t = x] \leq \ln d(x) - [K - \mathrm{size}(x)]/k.$$

Define $\widehat{\Phi}(x)$ to be this upper bound on the conditional expectation, that is $\widehat{\Phi}(x) \doteq \ln d(x) - [K - \mathrm{size}(x)]/k$. In the original analysis we showed that $\widehat{\Phi}(\tilde{x}_0) < \ln\epsilon/(1+\epsilon)$ and that if $\widehat{\Phi}(x) < \ln\epsilon/(1+\epsilon)$ and $\mathrm{size}(x) \geq K$ then $\#\mathrm{unc}(x) = 0$ and $\mathrm{dist}(x)/D < 1+\epsilon$.

In each iteration, the algorithm will choose a site s and a set of elements S so that $\widehat{\Phi}(x)$ does not increase. Let x and x', respectively, denote the

input: *sites, n elements,* dist, size, ϵ, D.
output: *integer solution $\hat{x}$ that approximates x^**

```
Let x̂(s) := 0 and x̂(s,e) := 0 for each s,e. Let d_e := 1.
Define dist'(s,e) ≐ (dist(s,e)/D)/(1+ε), where 0/0 ≐ 0 in case D = 0.
Until ∑_e d_e < 1 do:
    Choose a single site s to minimize ∑_e min{0, dist'(e,s) − d_e}/size(s).
    For each e such that dist'(e,s) ≤ d_e do:
        Set d_e := dist'(e,s).
        Set x̂(s,e) := 1 and x̂(s',e) := 0 for each s' ≠ s.
Return x̂.
```

Figure 5: The weighted greedy k-medians algorithm. The algorithm depends on D but not k. The algorithm returns a solution of distance $D(1+\epsilon)$ and size $\lceil k \ln(n+n\epsilon)\rceil$, provided some feasible fractional solution of distance D and size k exists.

configuration of the algorithm before and after a particular iteration. Then

$$\begin{aligned}\widehat{\Phi}(x') - \widehat{\Phi}(x) &= \frac{\text{size}(s)}{k} + \ln d(x')/d(x) \\ &< \frac{\text{size}(s)}{k} + \frac{d(x') - d(x)}{d(x)}. \end{aligned} \tag{6}$$

So it suffices to choose s and S so that the right-hand side of (6) above is non-positive. We know from the proof that if s and S are chosen randomly as in the rounding scheme, then $E[\text{size}(s)] = k/|x^*|$ and $E[d(x') - d(x)] = -d(x)/|x^*|$, so the expectation of the right-hand side of (6) is zero. Thus, there is *some* choice of s and S which makes it non-positive. Multiplying through by $d(x)/\text{size}(s)$, it suffices if

$$\frac{d(x)}{k} + \frac{d(x') - d(x)}{\text{size}(s)} \leq 0.$$

Thus, it suffices to choose s and S to minimize $[d(x') - d(x)]/\text{size}(s)$. This is guaranteed to make $\widehat{\Phi}(x') < \widehat{\Phi}(x)$.

The algorithm is shown in Figure 5. For convenience it maintains variables d_e so that $d(\hat{x}) = (\sum_e d_e) - 1/(1+\epsilon)$. The termination condition is thus equivalent to "$d(\hat{x}) < 1 - 1/(1+\epsilon)$". Strictly speaking, this termination condition differs from the one obtained by applying the method of conditional probabilities, but the modified termination condition suffices because

the analysis shows that whatever k is, the algorithm will terminate after the first iteration such that $\text{size}(\hat{x}) \geq k \ln(n + n\epsilon)$.

Like the algorithm for the unweighted case, this algorithm does not compute a solution to the fractional problem first, and it depends on D but not k. In the special case $D = 0$, the argument given in the proof of Guarantee 2 applies to show that the algorithm here is correct.

Guarantee 4 *Fix an instance of the weighted k-medians problem having a fractional solution of cost D and size k.*

Given the instance, D, and ϵ, the greedy weighted k-medians algorithm returns a solution $\hat{x}$ such that $\text{size}(\hat{x}) \leq k \ln(n + n/\epsilon) + \max_s \text{size}(s)$ *and* $\text{dist}(\hat{x})/D < 1 + \epsilon$.

The number of iterations is at most the number of sites because each iteration chooses a new site. If each $\text{size}(s) \geq 1$, then the number of iterations is at most $1 + k \ln(n + n/\epsilon)$, because each iteration increases the size by at least 1.

Because we can assume that $\max_s \text{size}(s) \leq k$ when the LP is a relaxation of a particular weighted k-median problem, we have:

Corollary 1 *The greedy weighted k-medians algorithm is a $(D(1+\epsilon), k(1+\ln(n+n/\epsilon)))$-approximation algorithm for the weighted k-medians problem.*

4 Fractional K-Medians Algorithm

In this section we derive and analyze a $((1+\epsilon)^2 D, (1+\epsilon)k)$-approximation, Lagrangian-relaxation algorithm for the fractional k-medians problem. The analysis here reuses much of the analysis of the greedy algorithm for the integer k-medians problem, so we assume the reader understands that derivation.

A goal of this section is to study the connection between greedy algorithms for the integer versions of problems and Lagrangian-relaxation algorithms for the fractional versions.

The rounding scheme is shown in Figure 6. It differs from the rounding scheme for the integer problem in that it has more iterations and rounds in smaller increments (of size $O(\min(1, \epsilon^2)/\ln(n + n/\epsilon))$) instead of rounding to integers. It also does not "reassign" elements to sites. Finally, in the analysis, we use the following Chernoff bound (Lemma 2) to bound the probability of failure.

input: *sites, n elements,* dist, ϵ, *frac. soln.* x^* *of size* k *and distance* D.
output: *frac. solution* $\tilde{x}$ *that with pos. probability approximates* x^*.

1. For each e, s do $\tilde{x}(e) := \tilde{x}(s) := \tilde{x}(s,e) := 0$.
2. $\delta := \epsilon/(1+\epsilon)$; $T := \lceil k \ln(n + n/\epsilon)/\mathrm{chern}(-\delta)\rceil$; $\Delta := k(1+\epsilon)/T$.
3. For $t := 1$ to T do:
4. Choose a single site s at random so that $\Pr(s \text{ chosen}) = x^*(s)/k$.
5. Set $\tilde{x}(s) := \tilde{x}(s) + \Delta$.
6. For each element e independently with probability $x^*(s,e)/x^*(s)$ do:
7. Increment $\tilde{x}(s,e)$ and $\tilde{x}(e)$ by Δ.
8. Return $\tilde{x}$.

Figure 6: Fractional k-medians rounding scheme. Note $\mathrm{chern}(-\delta) \doteq (1-\delta)\ln(1-\delta) + \delta = \Theta(\min\{1, \epsilon^2\})$. The variables $\tilde{x}(e)$ are used only for the analysis.

Lemma 2 (Chernoff Bound [12]) *Let $X_1, X_2, \ldots, X_k$ be a sequence of independent random variables in* $[0,1]$ *with* $E(X_i) \geq \mu_i$ *and* $\sum \mu_i = \mu > 0$. *Define* $\mathrm{chern}(\epsilon) \doteq (1+\epsilon)\ln(1+\epsilon) - \epsilon$. *Let* $\epsilon > 0$. *Then*

$$\Pr\left[\textstyle\sum_i X_i \leq \mu(1-\epsilon)\right] < \exp(-\mathrm{chern}(-\epsilon)\mu).$$

A proof is given in the Appendix (see Lemma 5).

Guarantee 5 *Let $\tilde{x}$ be the output of the fractional k-medians rounding scheme. Then with positive probability $\tilde{x}$ has* $\mathrm{size}(\tilde{x}) \leq (1+\epsilon)k$ *and* $\mathrm{dist}(\tilde{x})/D \leq (1+\epsilon)^2$.

Proof: The bound on the size always holds, because each of the T iterations adds $\Delta = (1+\epsilon)k/T$ to the size of $\tilde{x}$. Define the *coverage* of e to be $\tilde{x}(e) \doteq \sum_s \tilde{x}(s,e)$. It remains to show that with positive probability $\mathrm{dist}(\tilde{x})/D < (1+\epsilon)^2$ and for each e the coverage $\tilde{x}(e)$ is at least 1.

Say that an element e is *chosen with s* in an iteration if $\tilde{x}(s,e)$ is incremented in that iteration. Say that e is *chosen* in an iteration if it is chosen with any site during the iteration. Observe the following basic facts about each iteration (just as in the integer k-medians rounding scheme):

1. The probability of e being chosen with s is $x^*(s,e)/k$.
2. The probability of e being chosen (with any site) is $1/k$.
3. *Given* that e is chosen, the probability of e being chosen with s is $x^*(s,e)$.

Based on these facts we make the following two observations.

- For each element e, the expected number of iterations in which e will be chosen is T/k. The coverage of e will be at least one provided e is chosen at least $(1-\delta)T/k = 1/\Delta$ times. (Recall that $1-\delta = 1/(1+\epsilon)$.) Since each iteration is independent, by the Chernoff bound (Lemma 2), the probability that e is chosen fewer than $(1-\delta)T/k$ times is less than $\exp(-\mathrm{chern}(-\delta)T/k)$.

- In each iteration, the expected increase in $\mathrm{dist}(\tilde{x})$ is $\sum_{es}[x^*(s,e)/k]\,\Delta\mathrm{dist}(e,s)$, which equals $D\,\Delta/k$. Thus, after T rounds, $E[\mathrm{dist}(\tilde{x})] = TD\Delta/k = (1+\epsilon)D$.

Now, by the "naive union" and Markov bounds, we have

$$\Pr[\exists e\, \tilde{x}(e) \leq 1 \textbf{ or } \mathrm{dist}(\tilde{x})/D \geq (1+\epsilon)^2] < n\exp(-\mathrm{chern}(-\delta)T/k) + \frac{1+\epsilon}{(1+\epsilon)^2}$$

By the choice of T, the right-hand side above is at most 1. □

Algorithm Derivation and Analysis. Let $\tilde{x}_t$, T, D and k be as in the rounding scheme and proof.

Define X_{et} to be an indicator random variable for the event that element e is chosen in round t. The quantity of interest in the analysis is

$$\mathcal{Q} \doteq \frac{\mathrm{dist}(\tilde{x}_T)/D}{(1+\epsilon)^2} + \sum_e \frac{\prod_{t=1}^{T} 1 - \delta X_{et}}{(1-\delta)^{(1-\delta)T/k}}. \tag{7}$$

The fraction on the left comes from the Markov bound on the cost. The terms on the right come from the Chernoff bound, which bounds the probability of failure by the expectation of an exponential "penalty" function. These are added together because of the use of the naive union and Markov bounds.

The following facts are implicit in the proof of Guarantee 5. A proof is given in the Appendix. The reader may wish to verify these facts as an exercise, after reviewing the proof of the Chernoff bound.

Lemma 3 *1. If $\mathcal{Q} < 1$, then the rounding scheme succeeds.*

2. $E[\mathcal{Q} < 1]$.

Define $\Phi(t, x) \doteq E[\mathcal{Q} \mid \tilde{x}_t = x]$. We will apply the standard method of conditional expectations to the fractional rounding scheme with respect to $\mathcal{Q}$. The key question is how to choose s and S to minimize Φ. Recall that

$$\mathcal{Q} \doteq \frac{\mathrm{dist}(\tilde{x}_T)/D}{(1+\epsilon)^2} + \sum_e \frac{\prod_{t=1}^T 1 - \delta X_{et}}{(1-\delta)^{(1-\delta)T/k}}.$$

A careful calculation shows

$$\Phi(t, x) = \frac{\mathrm{dist}(x)/D + (T-t)\Delta/k}{(1+\epsilon)^2} + \sum_e \frac{(1-\delta)^{x(e)/\Delta}(1-\delta/k)^{T-t}}{(1-\delta)^{(1-\delta)T/k}}. \quad (8)$$

Let $\hat{x}_t$ denote the value of $\hat{x}$ after the tth iteration of the algorithm. Choosing a site s and a set of elements S so as to minimize $\Phi(t, \hat{x}_t)$ is the same as minimizing $\Phi(t, \hat{x}_t) - \Phi(t, \hat{x}_{t-1})$, which a calculation shows is equal to

$$\frac{\sum_{e \in S} \mathrm{dist}(s, e)\Delta/D}{(1+\epsilon)^2} - \delta \sum_{e \in S} \frac{(1-\delta)^{\hat{x}_{t-1}(e)/\Delta}(1-\delta/k)^{T-t}}{(1-\delta)^{(1-\delta)T/k}}. \quad (9)$$

Define $\hat{y}_t(e)$ to be $\delta(1+\epsilon)^2/\Delta$ times the term for e in the sum on the right-hand side of expression (9) above. Then given a particular site s, the set S minimizing expression (9) is $S(s) \doteq \{e : \mathrm{dist}(s, e)/D - \hat{y}_t(e) < 0\}$. Thus, to find the best s and S, it suffices for the algorithm to enumerate each possible site s with the single corresponding set $S(s)$. This gives us enough information to implement the algorithm efficiently.

The algorithm is shown in Figure 7. The algorithm depends on D and k. Because the algorithm comes from applying the standard method of conditional expectations to the fractional k-medians rounding scheme, the algorithm inherits its Guarantee 5:

Guarantee 6 *Fix an instance of the k-medians problem having a fractional solution of cost D and size k. Let $T \doteq \lceil k \ln(n + n/\epsilon) \rceil$.*

Given the instance, D, k, and ϵ, the fractional k-medians algorithm returns a fractional solution having total distance at most $(1+\epsilon)^2 D$ and size at most $(1+\epsilon)k$. The algorithm requires $O(k \ln(n+n/\epsilon)/\min\{1, \epsilon^2\})$ iterations, each of which can be implemented in linear time.

5 Fractional K-Medians by Packing/Covering

The fractional k-medians problem can be modeled as a mixed packing/covering linear programming problem as follows. Let m be the number of sites and

input: *sites, n elements,* dist, ϵ, D, k.
output: *fractional solution $\hat{x}$ that approximates x^*.*

1. For each e, s do $\hat{x}(e) := \tilde{x}(s) := \tilde{x}(s,e) := 0$.
2. $\delta := \epsilon/(1+\epsilon)$; $T := \lceil k \ln(n + n/\epsilon)/\mathrm{chern}(-\delta)\rceil$; $\Delta := k(1+\epsilon)/T$.
3. For each e do $\hat{y}(e) := [\delta(1+\epsilon)^2/\Delta](1-\delta/k)^T/(1-\delta)^{(1-\delta)T/k}$.
4. For $t := 1$ to T do:
5.   For each element e do: set $\hat{y}(e) := \hat{y}(e)/(1-\delta/k)$.
7.   Choose a single site s to minimize $\sum_s \min\{0, \mathrm{dist}(s,e)/D - \hat{y}(e)\}$.
8.   Set $\hat{x}(s) := \hat{x}(s) + \Delta$.
9.   For each element e such that $\mathrm{dist}(s,e)/D - \hat{y}(e) < 0$ do:
10.   Increment $\hat{x}(s,e)$ and $\hat{x}(e)$ by Δ.
11.   Set $\hat{y}(e) := (1-\delta)\hat{y}(e)$.
12. Return $\tilde{x}$.

Figure 7: Fractional k-medians algorithm. In case $D = 0$, interpret $0/0$ as 0 in lines 7 and 9.

n the number of elements. Let $P \subset \mathbb{R}^{m+nm}$ be the polytope of fractional solutions to the k-medians LP that may violate the coverage constraint,, i.e. $P = \{x : \sum_s x(s) = k; 0 \le x(s,e) \le x(s)\}$. Then consider the problem

$$\text{Find } x \in P \text{ such that } \mathrm{dist}(x) \le D \text{ and } \forall e \ \textstyle\sum_s x(s,e) \ge 1.$$

We solve this mixed packing/covering problem using the method of Plotkin, Shmoys, and Tardos (PST) [11]. For this problem, the PST algorithm can be given input $\langle \mathrm{dist}(), k, D, \epsilon \rangle$ and return an approximate solution x such that $\sum_s x(s,e) \ge 1/(1+\epsilon)$ and $\mathrm{dist}(x) \le (1+\epsilon)D$ (provided the original problem is feasible). We scale x, multiplying it by $(1+\epsilon)$, to get the final output.

Implementing the PST algorithm requires a subroutine that, given a vector $y \in \mathbb{R}^{m+mn}$, returns $x \in P$ minimizing $y \cdot x = \sum_s y(s)x(s) + \sum_{se} y(s,e)x(s,e)$. There will always be an optimal x that is a vertex of P, i.e. one such that the only non-zero coordinates of x satisfy $x(s) = x(s,e) = k$ for a single site s and some elements e. Thus an optimal x can be found by enumerating the sites and choosing the site s that minimizes

$$y(s) + \sum_e \min\{0, y(s,e)\mathrm{dist}(s,e)\}.$$

The subroutine then returns the corresponding x (whose only non-zero coordinates satisfy $x(s) = x(s,e) = k$ for e with $y(s,e) < 0$). This subroutine can be implemented in linear time.

The running time of the PST algorithm is dominated by the time spent in this subroutine. The subroutine is called $O(\rho \ln(m)/\epsilon^2)$ times, where ρ is the *width* of the problem instance, which in this case is

$$\max_{s,e}\{k\,\mathrm{dist}(s,e)/D, k\sum_s x(s,e)\}.$$

Under the assumption that each $\mathrm{dist}(s,e) = O(D)$, it follows that $\rho = O(k)$ because $\sum_s x(s,e) \le \sum_s x(s) = k$.

Appendix

The following *inequalities* are analogues of Wald's equation [3, p. 370]. In our applications, the condition $E(X_t - X_{t-1}\,|\,T \ge t) \le \Delta$ holds because $E(X_t - X_{t-1}) \le \Delta$ and T is a *stopping time*. This means in our case that the event "$T = t$" can be determined knowing only what happens in the first t iterations of our rounding schemes.

Lemma 1 (restatement): *Let $X_0, X_1, X_2, \ldots$ be a sequence of random variables and let $T \in \mathbb{N}_+$ be a random variable with $E[T] < \infty$. Suppose there exist $\Delta \in \mathbb{R}$ such that for all $t \ge 1$*

$$E(X_t - X_{t-1}\,|\,T \ge t) \le \Delta.$$

Then provided $X_t - X_{t-1} < c$ for some constant c and all $t \le T$,

$$E(X_T - X_0) \le \Delta E(T).$$

Proof: W.l.o.g. assume $\Delta = 0$, otherwise apply the change of variables $X'_t = X_t - \Delta t$ before proceeding.

If $E(X_T - X_0) = -\infty$ then the claim clearly holds. Otherwise,

$$\begin{aligned} E(X_T - X_0) &= \sum_{t=1}^{\infty} \Pr(T = t) E(X_t - X_0\,|\,T = t) \\ &= \sum_{t=1}^{\infty}\sum_{s=1}^{t} \Pr(T = t) E(X_s - X_{s-1}\,|\,T = t). \end{aligned}$$

Provided the technical condition is met, we claim the above sum is absolutely convergent. This is because $X_s - X_{s-1} \le c$ for some c so the sum of the positive terms is at most $\sum_t \sum_{s \le t} \Pr(T = t)c = \sum_t \Pr(T = t)ct = E[cT] < \infty$. Since we have assumed $E(X_T - X_0) > -\infty$, the sum of the absolute

values of the negative terms must be finite as well. Consequently exchanging the order of summation does not change the sum, and the double sum above equals

$$\begin{aligned}
& \sum_{s=1}^{\infty}\sum_{t=s}^{\infty} \Pr(T=t)E(X_s - X_{s-1} \,|\, T=t) \\
= \; & \sum_{s=1}^{\infty} \Pr(T \geq s)E(X_s - X_{s-1} \,|\, T \geq s) \\
\leq \; & \sum_{s=1}^{\infty} \Pr(T \geq s) \times 0 \; = \; 0
\end{aligned}$$

□

Lemma 4 *Let $X_0, X_1, X_2, \ldots$ be a sequence of random variables and let $T \in \mathbb{N}_+$ be a random variable with $E[T] < \infty$. Suppose there exist $\Delta \in \mathbb{R}$ such that for all $t \geq 1$*

$$E(X_t - X_{t-1} \,|\, T \geq t) \geq \Delta.$$

Then provided $X_t - X_{t-1} > -c$ for some constant c and all $t \leq T$,

$$E(X_T - X_0) \geq \Delta E(T).$$

This follows trivially by taking $X'_t \doteq -X_t$ and $\Delta' = -\Delta$ and applying Lemma 1 to X', Δ', and T.

The technical condition is necessary; consider choosing each X_t randomly to be $X_{t-1} \pm 2^t$ (with $X_0 = 0$) and letting $T = \min\{t : X_t = 1\}$. Then $E[X_t - X_{t-1}] \leq 0$, so taking $\Delta = 0$, all conditions for the theorem except "$X_t - X_{t-1} < c$" are met. But the conclusion $E[X_T - X_0] \leq 0$ does not hold, because $E[X_T - X_0] = 1$.

For convenience we also state the Chernoff bounds we use, as well as the naive union bound and the Markov bound.

Lemma 5 (Chernoff Bound [12]) *Let $X_1, X_2, \ldots, X_k$ be a sequence of independent random variables in $[0,1]$ with $E(X_i) \leq \mu_i$ and $\sum \mu_i = \mu > 0$. Let $\epsilon > 0$. Define* $\mathrm{chern}(\epsilon) \doteq (1+\epsilon)\ln(1+\epsilon) - \epsilon$. *Then*

$$\Pr\left[\textstyle\sum_i X_i \geq \mu(1+\epsilon)\right] < \exp(-\mathrm{chern}(\epsilon)\mu).$$

Proof:

$$\begin{aligned}
\Pr\left[\sum_i X_i \geq (1+\epsilon)\mu\right] &= \Pr\left[\prod_i \frac{(1+\epsilon)^{X_i}}{(1+\epsilon)^{(1+\epsilon)\mu_i}} \geq 1\right] \\
&\leq E\left[\prod_i \frac{1+\epsilon X_i}{(1+\epsilon)^{(1+\epsilon)\mu_i}}\right] \\
&= \prod_i \frac{1+\epsilon E(X_i)}{(1+\epsilon)^{(1+\epsilon)\mu_i}} \\
&< \prod_i \frac{e^{\epsilon\mu_i}}{(1+\epsilon)^{(1+\epsilon)\mu_i}}
\end{aligned}$$

The last line equals $\exp[-\mathrm{chern}(\epsilon)\mu]$. The second step uses $(1+\alpha)^z \leq 1+\alpha z$ for $0 \leq z \leq 1$ and Markov's inequality. The last uses $E(X_i) \leq \mu_i$ and $1+z \leq e^z$, which is strict if $z \neq 0$. □ □

Essentially the same proof with "$-\epsilon$" replacing "ϵ" proves Lemma 2.

Lemma 6 (Naive Union Bound) *Let $A_1, A_2, \ldots, A_k$ be a set of random events. Then*

$$\Pr(A_1 \mathrm{or} A_2 \mathrm{or} \cdots \mathrm{or} A_k) \leq \Pr(A_1) + \Pr(A_2) + \cdots + \Pr(A_k).$$

Proof: Define random indicator variables $Y_1, \ldots, Y_k$ so that Y_i is 1 if A_i occurs, and 0 otherwise. Then

$$\Pr(\exists i\, A_i) \;=\; \Pr(\sum_i A_i \geq 1) \;\leq\; E(\sum_i A_i) \;=\; \sum_i \Pr(A_i).$$

The second step follows by applying the Markov Bound (Lemma 7) to the random variable $\sum_i A_i$. □

Lemma 7 (Markov Bound) *Let X be a non-negative random variable and $\alpha \geq 0$. Then*

$$\Pr[X \geq \alpha E(X)] \leq 1/\alpha.$$

Proof: Define random indicator variable Y to be 1 if $X \geq \alpha E(X)$ and 0 otherwise. Then $Y \leq X/[\alpha E(X)]$ so

$$\Pr[X \geq \alpha E(X)] \;=\; E(Y) \;\leq\; E(X/[\alpha E(x)]) \;=\; 1/\alpha.$$

$\square$

Proof of Lemma 3.

Proof: Here is a proof of the first claim. Suppose $\mathcal{Q} < 1$. Then clearly $\text{dist}(\tilde{x}_T)/D < (1+\epsilon)^2$, so the distance bound holds. It remains to show $\forall e\, \tilde{x}_T(e) > 1$. Note that for any e,

$$(1-\delta)^{\tilde{x}_T(e)/\Delta} = (1-\delta)^{\sum_t X_{et}} \leq \prod_{t=1}^{T}(1-\delta X_{et})$$

so that $\mathcal{Q}$ is at least as large as

$$\sum_e \frac{(1-\delta)^{\tilde{x}_T(e)/\Delta}}{(1-\delta)^{(1-\delta)T/k}}.$$

But if any e has $\tilde{x}_T(e) \leq 1 = (1-\delta)T\Delta/k$, then the above sum is at least 1.

To prove the second claim, note that for any e, the X_{et}'s are independent, and that the expectation of a product of independent random variables is the product of the expectations. Thus,

$$E\left[\prod_{t=1}^{T} 1 - \delta X_{et}\right] = \prod_{t=1}^{T} 1 - \delta E[X_{et}] = \prod_{t=1}^{T} 1 - \delta/k < \exp(-T\delta/k).$$

and

$$E\left[\frac{\prod_{t=1}^{T} 1 - \delta X_{et}}{(1-\delta)^{(1-\delta)T/k}}\right] < \frac{\exp(-T\delta/k)}{(1-\delta)^{(1-\delta)T/k}} = \exp(-\text{chern}(-\delta)T/k).$$

This, combined with the last three inequalities in the proof of Guarantee 5, give the claim. $\square$

References

[1] Ahn, Cooper, Cornuejols, and Frieze. Probabilistic analysis of a relaxation for the k-median problem. *Mathematics of Operations Research*, 13:1–31, Feb 1988.

[2] Noga Alon and Joel H. Spencer. *The Probabilistic Method.* John Wiley and Sons, New York, 1992.

[3] Allan Borodin and Ran El-Yaniv. *Online Computation and Competitive Analysis.* Cambridge University Press, 1998.

[4] M. Charikar, S. Guha, E. Tardos, and D.B. Shmoys. A constant-factor approximation algorithm for the k-median problem. In *Proceedings of the Thirty First Annual ACM Symposium on Theory of Computing*, 1999.

[5] V. Chvatal. A greedy heuristic for the set-covering problem. *Mathematics of Operations Research*, 4(3):233–235, 1979.

[6] Sudipto Guha and Samir Khuller. Greedy strikes back: Improved facility location algorithms. In *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 649–657, San Francisco, California, 25–27 January 1998.

[7] Dorit S. Hochbaum. Heuristics for the fixed cost median problem. *Math. Programming*, 22(2):148–162, 1982.

[8] David S. Johnson. Approximation algorithms for combinatorial problems. *Journal of Computer and System Sciences*, 9:256–278, 1974.

[9] Jyh-Han Lin and Jeffrey Scott Vitter. ϵ-approximations with minimum packing constraint violation (extended abstract). In *Proceedings of the Twenty Fourth Annual ACM Symposium on Theory of Computing*, pages 771–782, Victoria, British Columbia, Canada, 4–6 May 1992.

[10] László Lovász. On the ratio of optimal integral and fractional covers. *Discrete Mathematics*, 13:383–390, 1975.

[11] Serge A. Plotkin, David B. Shmoys, and Éva Tardos. Fast approximation algorithms for fractional packing and covering problems. *Math. Oper. Res.*, 20(2):257–301, 1995.

[12] Prabhakar Raghavan. Probabilistic construction of deterministic algorithms approximating packing integer programs. *Journal of Computer and System Sciences*, 37(2):130–143, October 1988.

[13] Prabhakar Raghavan and C. Thompson. Randomized rounding: A technique for provably good algorithms and algorithmic proofs. *Combinatorica*, 7:365–374, 1987.

[14] David B. Shmoys, Éva Tardos, and Karen Aardal. Approximation algorithms for facility location problems (extended abstract). In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, pages 265–274, El Paso, Texas, 4–6 May 1997.

[15] Neal E. Young. Randomized rounding without solving the linear program. In *Proceedings of the Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 170–178, San Francisco, California, 22–24 January 1995.