Approximation algorithms for facility location problems

David B. Shmoys* Éva Tardos† Karen Aardal‡

Abstract

We present new approximation algorithms for several facility location problems. In each facility location problem that we study, there is a set of locations at which we may build a facility (such as a warehouse), where the cost of building at location i is f_i; furthermore, there is a set of client locations (such as stores) that require to be serviced by a facility, and if a client at location j is assigned to a facility at location i, a cost of c_{ij} is incurred. The objective is to determine a set of locations at which to open facilities so as to minimize the total facility and assignment costs. In the uncapacitated case, each facility can service an unlimited number of clients, whereas in the capacitated case, each facility can serve, for example, at most u clients. These models and a number of closely related ones have been studied extensively in the Operations Research literature.

We shall consider the case in which the assignment costs are symmetric and satisfy the triangle inequality. For the uncapacitated facility location, we give a polynomial-time algorithm that finds a solution within a factor of 3.16 of the optimal. This is the first constant performance guarantee known for this problem. We also present approximation algorithms with constant performance guarantees for a number of capacitated models as well as a generalization in which there is a 2-level hierarchy of facilities. Our results are based on the filtering and rounding technique of Lin & Vitter. We also give a randomized variant of this technique that can then be derandomized to yield improved performance guarantees.

1 Introduction

We shall present approximation algorithms for a variety of facility location problems. One of the most well-studied problems in the Operations Research literature is the *uncapacitated facility location problem*, dating back to the work of Balinski [2], Kuehn & Hamburger [16], Manne [20], and Stollsteimer [25, 26] in the early 60's. In its simplest form, the problem is as follows: we wish to find optimal locations at which to build facilities (such as warehouses) to serve a given set of n client locations (such as stores); we are also given a set of locations at which facilities may be built, where building a facility at location i incurs a cost of f_i; each client j must

*`shmoys@cs.cornell.edu`. School of Operations Research & Industrial Engineering and Department of Computer Science, Cornell University, Ithaca, NY 14853. Research partially supported by NSF grants CCR-9307391 and DMS-9505155 and ONR grant N00014-96-1-0050O.

†`eva@cs.cornell.edu`. Department of Computer Science and School of Operations Research & Industrial Engineering, Cornell University, Ithaca, NY 14853. Research partially supported by NSF grants DMI-9157199 and DMS-9505155 and ONR grant N00014-96-1-0050O.

‡`aardal@cs.ruu.nl`. Department of Computer Science, Utrecht University, Utrecht, The Netherlands. Research partially supported by NSF grant CCR-9307391, and by ESPRIT Long Term Research Project No. 20244 (Project ALCOM-IT: *Algorithms and Complexity in Information Technology*).

be assigned to one facility, thereby incurring a cost of c_{ij}, the distance between locations i and j; the objective is to find a solution of minimum total cost. The main result of this paper is an approximation algorithm that finds a solution of cost within a factor of 3.16 of the optimum, provided the distances between the locations are symmetric and satisfy the triangle inequality. This is the first approximation algorithm for this problem with a constant performance guarantee.

This $\mathcal{NP}$-hard problem has been studied from, among others, the perspective of worst-case performance guarantees, probabilistic analysis of the average-case performance, polyhedral characterizations, and the empirical investigation of heuristics. Its prominence in the literature is due to the fact that there are a wide variety of applications as well as its appealing simplicity. For an extensive survey of work on this, and closely related problems, the reader is referred to the textbook edited by Mirchandani & Francis [21], and in particular, the chapter by Cornúejols, Nemhauser, and Wolsey [6]. For a more in-depth explanation of results known for these models, there is an extensive discussion in the textbook of Nemhauser & Wolsey [22].

We shall briefly survey the results known on approximation algorithms for the uncapacitated facility location problem. Throughout this paper, a *ρ-approximation algorithm* is a polynomial-time algorithm that always finds a feasible solution with objective function value within a factor of ρ of optimal. Hochbaum [12] showed that the greedy algorithm is an $O(\log n)$-approximation algorithm for this problem, and provided instances to verify that this analysis is asymptotically tight. This provided a stark contrast to earlier results of Cornúejols, Fisher, & Nemhauser [5], who considered a problem that is equivalent from the perspective of optimization, but not approximation: their objective was to find a solution so as to maximize the difference between the assignment "costs" (which they interpreted as profits) and the facility costs. For this objective, Cornúejols, Fisher, & Nemhauser showed that the greedy algorithm, in effect, came within a constant factor of optimal. Although they justified their variant with an application for computing an optimal strategy for gaining profit from interest accrued by delays in clearing checks, the original objective is much more natural for the typical network design type of setting in which the uncapacitated facility location problem usually arises.

Lin & Vitter [19] gave an elegant technique, called filtering, for rounding fractional solutions to linear programming relaxations, and as one application of this technique for designing approximation algorithms, gave another $O(\log n)$-approximation algorithm for the uncapacitated facility location problem. Furthermore, Lin & Vitter considered the *k-median problem*, where facility costs are replaced by a constraint that limits the number of facilities to k; that is, there are n locations, and one is allowed to build facilities at no more than k of them to serve all n locations; the objective is to minimize the total assignment costs. They gave an algorithm that finds a solution for which the objective is within a factor of $1+\epsilon$ of the optimum, but is infeasible since it opens $(1+1/\epsilon)(\ln n+1)k$ facilities. Lin & Vitter [18] also showed that in the special case of the k-median problem where the assignment costs are symmetric and satisfy the triangle inequality, one can find a solution of cost no more than $2(1+\epsilon)$ times the optimum, while using at most $(1+1/\epsilon)k$ facilities.

All of the problems discussed above are min-sum problems, in that the sum of the assignment costs enters into the objective function. Much stronger approximation results are known for min-max facility location problems. The *k-center problem* is the min-max analogue of the k-median problem: one builds facilities at k locations out of n, so as to minimize the maximum distance that an unselected location is from its nearest facility. Hochbaum & Shmoys [13] and subsequently Dyer & Frieze [7] gave 2-approximation algorithms for this problem, and also gave extensions for weighted variants. Bar-Ilan, Kortsarz, & Peleg [3] considered a capacitated variant, in which each facility can serve at most u locations, and gave a 10-approximation algorithm for this problem. Khuller & Sussmann [15] recently improved this to give a 6-approximation algorithm. They also considered a variant in which one can build multiple facilities of capacity u at a location, for which they gave a 5-approximation algorithm.

Our results for min-sum facility location problems are filtering and rounding algorithms that build on the results of Lin and Vitter [18, 19]. In addition to our algorithm for the uncapacitated facility location problem, we will give approximation algorithms for several capacitated variants of this problem. We shall assume that each location has a given demand that must be serviced by some facility, and each facility can service a total demand that is at most u. In assigning locations to facilities, we can either require that each location have its entire demand serviced by a unique facility, or else we can allow a client's demand to be split among several open facilities. For both settings, we will give an algorithm that finds a solution of cost within a constant factor of optimal, but uses facilities that have a constant factor greater capacity than u (and are proportionately more expensive). Finally, we also consider the variant of the problem in which we may build multiple facilities at a location, each of capacity u, and give an approximation algorithm with constant performance guarantee. All of the constants are relatively small (less than 10); for example, in the setting in which we may build multiple facilities at a location and may split a client's demand among several facilities, we give a 5.69-approximation algorithm. Our strongest performance guarantees are based on a randomized variant of the filtering technique of Lin & Vitter, which yields deterministic algorithms with improved performance guarantees.

2 The uncapacitated facility location problem

In this section, we will consider the following problem: we are given a set of locations $N = \{1, \ldots, n\}$, and distances between them, c_{ij}, $i, j = 1, \ldots, n$; there is a subset $F \subseteq N$ of locations at which we may open a facility, and a subset $D \subseteq N$ of locations that must be assigned to some open facility; for each location $j \in D$, there is a positive integral demand d_j that must be shipped to its assigned location. For each location $i \in F$, the non-negative cost of opening a facility at i is f_i. The cost of assigning location i to an open facility at j is c_{ij} per unit of demand shipped. We shall assume that these costs are non-negative, symmetric, and satisfy the triangle inequality: that is, $c_{ij} = c_{ji}$ for all $i, j \in N$, and $c_{ij} + c_{jk} \geq c_{ik}$ for all $i, j, k \in N$. We wish to find a feasible assignment of each location in D to an open facility so as to minimize the total cost incurred. This is the *metric uncapacitated facility location problem.*

This problem can be stated as the following integer program, where the 0-1 variable y_i, $i \in F$ indicates if a facility is opened at location i, and the 0-1 variable x_{ij}, $i \in F$, $j \in D$, indicates if location j is assigned to a facility at i:

$$\text{minimize} \sum_{i \in F} f_i y_i + \sum_{i \in F} \sum_{j \in D} d_j c_{ij} x_{ij} \tag{1}$$

subject to

$$\begin{aligned}
\sum_{i \in F} x_{ij} &= 1, && \text{for each } j \in D, && (2)\\
x_{ij} &\leq y_i, && \text{for each } i \in F,\ j \in D, && (3)\\
x_{ij} &\in \{0,1\}, && \text{for each } i \in F,\ j \in D, && (4)\\
y_i &\in \{0,1\}, && \text{for each } i \in F. && (5)
\end{aligned}$$

The constraints (2) ensure that each location $j \in D$ is assigned to some location $i \in F$, and the constraints (3) ensure that whenever a location j is assigned to location i, then a facility must have been opened at i (and paid for). For notational simplicity, we shall refer to 0-1 variables x_{ij} for each $i, j \in N$, with the understanding that if $i \notin F$ or $j \notin D$, then $x_{ij} = 0$; similarly, we shall refer to variables y_i, for each $i \notin F$, with the understanding that $y_i = 0$ in this case.

We will derive an approximation algorithm for the uncapacitated facility location problem that is based on solving the linear relaxation of this integer program, and rounding the fractional solution to an integer solution that increases its cost by a relatively small constant factor. This rounding algorithm consists of two phases. We apply the filtering and rounding technique of Lin & Vitter [19] to obtain a new *fractional* solution, where the new solution has the property that whenever a location j is fractionally assigned to a (partially opened) facility i, the cost c_{ij} associated with that assignment is not too big. We then show how a fractional solution with this *closeness property* can be rounded to a near-optimal integer solution.

Consider the linear relaxation to the integer program (1)-(5), where the 0-1 constraints (4) and (5) are replaced, respectively, with

$$\begin{aligned} x_{ij} &\geq 0, && \text{for each } i \in F,\ j \in D, && (6)\\ y_i &\geq 0, && \text{for each } i \in F. && (7) \end{aligned}$$

Given g_j, for each $j \in D$, we shall say that a feasible solution (x, y) to this linear program is *g-close* if it satisfies the property

$$x_{ij} > 0 \Rightarrow c_{ij} \leq g_j. \qquad (8)$$

The following lemma is proved by applying the filtering technique of Lin & Vitter [19]. Given a feasible fractional solution (x, y), we shall define the *α-point*, $c_j(\alpha)$, for each location $j \in D$. Focus on a location $j \in D$, and let π be a permutation such that $c_{\pi(1)j} \leq c_{\pi(2)j} \leq \cdots \leq c_{\pi(n)j}$. Recall that if $i \notin F$, then $x_{ij} = 0$. We then set $c_j(\alpha) = c_{\pi(i^*)j}$, where $i^* = \min\{i' : \sum_{i=1}^{i'} x_{\pi(i)j} \geq \alpha\}$.

Lemma 1 *Let α be a fixed value in the interval $(0,1)$. Given a feasible fractional solution (x, y), we can find a g-close feasible fractional solution $(\bar{x}, \bar{y})$ in polynomial time, such that*

1. *$g_j \leq c_j(\alpha)$, for each $j \in D$;*
2. *$\sum_{i \in F} f_i \bar{y}_i \leq (1/\alpha) \sum_{i \in F} f_i y_i$.*

Proof: The proof of this lemma is quite simple. For each $j \in D$, let $\alpha_j = \sum_{i \in F:\ c_{ij} \leq c_j(\alpha)} x_{ij}$; clearly, $\alpha_j \geq \alpha$. We merely set

$$\bar{x}_{ij} = \begin{cases} x_{ij}/\alpha_j & \text{if } c_{ij} < c_j(\alpha); \\ 0 & \text{otherwise.} \end{cases}$$

For each $i \in F$, we set $\bar{y}_i = \min\{1, y_i/\alpha\}$. The definition of $\bar{x}$ is set up exactly to ensure that the first condition holds. Furthermore, since $\bar{y}_i \leq (1/\alpha) y_i$, the second condition hold as well. Finally, a straightforward calculation verifies that $(\bar{x}, \bar{y})$ is a feasible fractional solution. ■

If we let $S = \{i : c_{ij} \geq c_j(\alpha)\}$, then the definition of $c_j(\alpha)$ implies that $\sum_{i \in S} x_{ij} \geq 1 - \alpha$. Hence,

$$\sum_{i \in F} c_{ij} x_{ij} \geq \sum_{i \in S} c_{ij} x_{ij} \geq (1 - \alpha) c_j(\alpha),$$

or equivalently,

$$c_j(\alpha) \leq \frac{1}{1-\alpha} \sum_{i \in F} c_{ij} x_{ij} \qquad (9)$$

We will show how to exploit the closeness property in rounding fractional solutions to near-optimal integer solutions. This result generalizes a similar claim used by Lin & Vitter [18] to obtain their results for the metric k-median problem.

Lemma 2 *Given a feasible fractional g-close solution $(\bar{x}, \bar{y})$, we can find a feasible integer $3g$-close solution $(\hat{x}, \hat{y})$ such that*

$$\sum_{i \in F} f_i \hat{y}_i \leq \sum_{i \in F} f_i \bar{y}_i.$$

Proof: We shall first present the rounding algorithm, and then prove that it yields the lemma. We are given g_j, $j \in D$, and a feasible fractional solution $(\bar{x}, \bar{y})$ that is g-close. The algorithm iteratively converts this solution into a $3g$-close integer solution $(\hat{x}, \hat{y})$, without increasing the total facility cost.

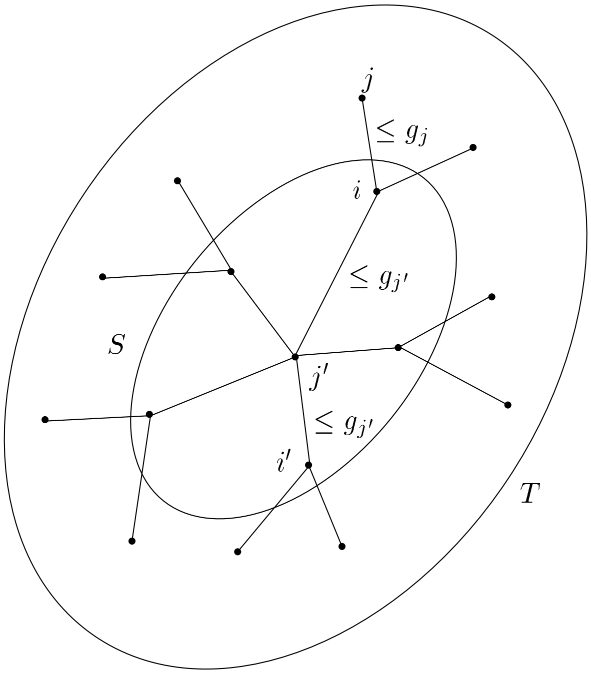

Figure 1: Rounding the solution near j', where edges correspond to positive components of $\hat{x}$

The algorithm maintains a feasible fractional solution $(\hat{x}, \hat{y})$; initially, we set $(\hat{x}, \hat{y}) = (\bar{x}, \bar{y})$. Throughout the execution of the algorithm, $\hat{F}$ will denote the set of partially opened facility locations for the current solution; that is, $\hat{F} = \{i \in F : 0 < \hat{y}_i < 1\}$. We shall also let $\hat{D}$ denote the set of those locations j that are assigned only to facilities in $\hat{F}$; that is, $\hat{x}_{ij} > 0$ implies that $i \in \hat{F}$. In each iteration, we first find the location $j \in \hat{D}$ for which g_j is smallest; let j' denote this location. Let S be the set of facilities $i \in F$ for which $\hat{x}_{ij'} > 0$ (see Figure 1); that is,

$$S = \{i \in \hat{F} : \hat{x}_{ij'} > 0\}.$$

We will assign j' to the location $i \in S$ for which f_i is smallest; let i' denote this location. We round the values $\{\hat{y}_i\}_{i \in S}$ by setting $\hat{y}_{i'} = 1$, and $\hat{y}_i = 0$ for each

$i \in S - \{i'\}$. Let T denote the set of locations that are partially assigned by $\hat{x}$ to locations in S; that is,

$$T = \{j : \exists i \in S \text{ such that } \hat{x}_{ij} > 0\}.$$

We assign each location $j \in T$ to the facility opened at i'; that is, we set $\hat{x}_{i'j} = 1$ and $\hat{x}_{ij} = 0$ for each $i \neq i'$. When $\hat{D}$ becomes empty, then for each location $j \in D$, there exists i' such that $\hat{x}_{i'j} > 0$ and $\hat{y}_{i'} = 1$, and so j can be assigned to i'; that is, we round $\hat{x}$ by setting $\hat{x}_{i'j} = 1$ and $\hat{x}_{ij} = 0$ for each $i \neq i'$. We shall argue that the algorithm maintains the following properties:

$(P1)$ $(\hat{x}, \hat{y})$ is a feasible fractional solution;

$(P2)$ $\sum_{i \in F} f_i \hat{y}_i \leq \sum_{i \in F} f_i \bar{y}_i$;

$(P3)$ $\hat{x}_{ij} > 0$ and $i \in \hat{F} \Rightarrow c_{ij} \leq g_j$;

$(P4)$ $\hat{x}_{ij} > 0$ and $i \notin \hat{F} \Rightarrow c_{ij} \leq 3g_j$.

These properties certainly hold when the algorithm starts. Furthermore, if they hold when the algorithm stops (and so property $(P3)$ becomes vacuous), then we have proved Lemma 2.

We shall show that these properties are maintained by the algorithm in each iteration. Property $(P1)$ is clearly maintained: the algorithm only assigns a location $j \in D$ to an opened facility, and when we set any variable $\hat{y}_i$ to 0, we also set each variable $\hat{x}_{ij}$ to 0. Property $(P3)$ is trivially maintained, since the algorithm never sets a variable $\hat{x}_{ij}$ to be in the interval (0,1) nor adds a location to $\hat{F}$.

To show that property $(P4)$ is maintained during an iteration, consider some variable $\hat{x}_{i'j}$ that is set to 1 during it. We examine the situation at the start of this iteration as depicted in Figure 1. Since j must be in T, there must exist $i \in S$ such that $\hat{x}_{ij} > 0$. Furthermore, both $\hat{x}_{ij'} > 0$ and $\hat{x}_{i'j'} > 0$, since $i, i' \in S$. But $S \subseteq \hat{F}$, and hence by $(P3)$, we have that $c_{ij} \leq g_j$, $c_{ij'} \leq g_{j'}$, and $c_{i'j'} \leq g_{j'}$. By the triangle inequality, we have that $c_{i'j} \leq c_{i'j'} + c_{j'i} + c_{ij} \leq 2g_{j'} + g_j \leq 3g_j$, where the last inequality follows from our choice of j'. Hence, property $(P4)$ is maintained by the algorithm.

To show that $(P2)$ is maintained, we note that

$$f_{i'} = \min_{i \in S} f_i \leq \sum_{i \in S} f_i \hat{x}_{ij},$$

where the inequality follows from the fact that

$$\sum_{i \in S} \hat{x}_{ij} = 1,$$

and that the minimum of a set of numbers is never more than their weighted average. Finally, $\hat{x}_{ij} \leq \hat{y}_i$, and so we have that $f_{i'} \leq \sum_{i \in S} f_i \hat{y}_i$. But this inequality implies that the facility cost of $\hat{y}$ never increases throughout the execution of the algorithm, which proves that $(P2)$ is maintained.

Finally, we note that the simple rounding performed when $\hat{D}$ is empty also maintains these four properties. This completes the proof of the lemma. ■

If we start with a feasible fractional solution (x, y) and apply Lemma 1 to get $(\bar{x}, \bar{y})$, and then apply Lemma 2 to $(\bar{x}, \bar{y})$, the resulting feasible integer solution $(\hat{x}, \hat{y})$ has facility cost at most

$$\sum_{i \in F} f_i \hat{y}_i \leq \sum_{i \in F} f_i \bar{y}_i \leq (1/\alpha) \sum_{i \in F} f_i y_i.$$

On the other hand, for each location $j \in D$, its assignment cost in $\hat{x}$ is at most $3g_j \leq 3c_j(\alpha) \leq \frac{3}{1-\alpha}\sum_{i\in F} c_{ij}x_{ij}$. By combining these two bounds, we see that the total cost of $(\hat{x}, \hat{y})$

$$
\begin{aligned}
&= \sum_{i\in F} f_i \hat{y}_i + \sum_{i\in F}\sum_{j\in D} d_j c_{ij}\hat{x}_{ij} && (10)\\
&\leq \frac{1}{\alpha}\sum_{i\in F} f_i y_i + 3\sum_{j\in D} d_j c_j(\alpha) && (11)\\
&\leq \max\{\frac{1}{\alpha}, \frac{3}{1-\alpha}\}(\sum_{i\in F} f_i y_i + \sum_{i\in F}\sum_{j\in D} d_j c_{ij} x_{ij}). && (12)
\end{aligned}
$$

If we set $\alpha = 1/4$, then we see that the total cost of $(\hat{x}, \hat{y})$ is within a factor of 4 of the cost of (x, y). By rounding an optimal solution (x, y) to the linear relaxation, we get the following theorem.

Theorem 3 *For the metric uncapacitated facility location problem, filtering and rounding yields a 4-approximation algorithm.*

In Section 5, we will give an algorithm with a somewhat better performance guarantee, by refining this analysis. Nonetheless, we do not know very much about the extent to which there is an inherent gap between integer and fractional optimal solutions to this formulation for the metric uncapacitated location problem.

3 The capacitated facility location problem

In this section, we consider the case in which each open facility can be assigned to serve a total demand that is at most u, where u is a positive integer. We will show how to adapt our algorithm for the uncapacitated case to this more general setting.

In the uncapacitated case, if we are given the optimal value of y, then it is trivial to find the corresponding x: we simply assign each location $j \in D$ to the location i for which c_{ij} is the minimum among all possibilities where $y_i = 1$. In the capacitated case, the situation is somewhat more complicated. First of all, there are two variants of the problem, depending on whether each location's demand must be assigned to only one facility, or the demand may be fractionally split among more than one (completely) open facility.

We will first focus on the latter case. If we are given the optimal value of y, the problem of finding a minimum-cost assignment that satisfies each location's demand, while assigning at most u to each open facility is an instance of the transportation problem. (For a review of the basics for this problem see, e.g., the textbook of Lawler [17].) Briefly, the optimal solution to this problem can be found in polynomial time, and if u and the demands d_j, $j \in D$, are integers, then the flow values $d_j x_{ij}$ in the solution found are also integral. For example, this implies that in the case that the demands are all 1 and u is an integer, there is no distinction between the two capacitated variants mentioned above: we always find an assignment that routes each demand to a unique open facility.

Our algorithm is based on rounding an optimal solution to its linear programming relaxation. This linear programming relaxation is identical to the one used in the uncapacitated case, except we must explicitly require that

$$0 \leq y_i \leq 1, \qquad \text{for each } i \in F, \tag{13}$$

and we must impose capacity constraints

$$\sum_{j\in D} d_j x_{ij} \leq u y_i, \qquad \text{for each } i \in F. \tag{14}$$

It is not possible to design an approximation algorithm for the capacitated problem based solely on this linear programming relaxation, since the ratio between its integer and fractional optimal is unbounded. To see this, consider an instance with $u+1$ locations that are all distance 0 from each other with fixed costs $f_1 = 0$ and $f_i = 1$, $i = 2, \ldots, u+1$. There is the following fractional solution: set $y_1 = 1$, $y_2 = 1/u$, $x_{1j} = u/(u+1)$ and $x_{2j} = 1/(u+1)$, $j = 1, \ldots, u+1$. The cost of this solution is $1/u$, whereas the optimal integer solution has cost 1. However, if we also allow the near-optimal solution to slightly overuse any facility then clearly one can, at least in this instance, find an integer solution of cost nearly equal to that for the optimal fractional one.

Motivated by this discussion, we shall call an algorithm for the metric capacitated facility location problem a (ρ, ρ')-*approximation algorithm* if it finds, in polynomial time, a solution of total cost within a factor of ρ of the true optimum, but each facility $i \in F$ is expanded to have capacity $\rho' u$ at a cost of $\rho' f_i$. In this section, we present a $(7, 7/2)$-approximation algorithm. We will express the relaxation in the capacity constraint by allowing $0 \leq y_i \leq \rho'$, for each $i \in F$. If (x, y) is a feasible fractional solution to this modified linear program, then it is ρ'-*relaxed*. Furthermore, the analogue of an integer solution with this relaxation is that y_i is either 0 or at least 1, for each $i \in F$; if (x, y) is a ρ'-relaxed solution with this additional property, then we will call it a ρ'-*relaxed integer solution* (even though it is not really integer at all).

Once again, our algorithm is based on first filtering, and then rounding. It is quite straightforward to generalize Lemma 1 to obtain the following result.

Lemma 4 *Let α be a fixed value in the interval $(0, 1)$. Given a feasible fractional solution (x, y), we can find a g-close fractional solution $(\bar{x}, \bar{y})$ in polynomial time, such that*

1. $g_j \leq c_j(\alpha)$, *for each* $j \in D$;
2. $\sum_{i \in F} f_i \bar{y}_i \leq (1/\alpha) \sum_{i \in F} f_i y_i$;
3. $(\bar{x}, \bar{y})$ *is* $1/\alpha$-*relaxed.* ■

On the other hand, the rounding algorithm becomes a bit more complicated, since the uncapacitated algorithm takes great advantage of the fact that there are no capacities: *all* demand fractionally routed to *any* location in S ends up being assigned to j' (using the notation in the proof of Lemma 2). We next prove the following analogue of Lemma 2.

Lemma 5 *Given a ρ'-relaxed fractional g-close solution $(\bar{x}, \bar{y})$, we can find a $2\rho'$-relaxed integer $3g$-close solution $(\hat{x}, \hat{y})$ in polynomial time, such that*

$$\sum_{i \in F} f_i \hat{y}_i \leq 4 \sum_{i \in F} f_i \bar{y}_i.$$

Proof: We first describe the rounding algorithm in detail, and then prove that it produces the claimed solution. As in the uncapacitated case, we maintain a solution $(\hat{x}, \hat{y})$ and the algorithm gradually rounds each $0 < \hat{y}_i < 1$ to either 0 or 1; initially, we set $\hat{x} = \bar{x}$, we set $\hat{y}_i = 1$ for each i such that $\bar{y}_i \in [1/2, 1)$, and we set $\hat{y}_i = \bar{y}_i$ otherwise. We also maintain a set $\hat{F} \subseteq F$ of facilities i for which $0 < \hat{y}_i < 1$ (but due to the previous step, this will be equivalent to restricting $0 < \hat{y}_i < 1/2$). For each $j \in D$, the algorithm keeps track of the fraction of the demand for location j that is satisfied by locations in $\hat{F}$: let $\beta_j = \sum_{i \in \hat{F}} \hat{x}_{ij}$ for each $j \in D$. In this case, we let $\hat{D} \subseteq D$ be the set of locations j for which $\beta_j > 1/2$. (In the uncapacitated case, the restriction for $\hat{D}$ was, in effect, that $\beta_j = 1$.)

In each iteration, we first select the location $j \in \hat{D}$ for which g_j is minimum, and let j' denote this location. Again, we let

$$S = \{i \in \hat{F} : \hat{x}_{ij'} > 0\}$$

and

$$T = \{j \in D : \exists i \in S \text{ such that } \hat{x}_{ij} > 0\}.$$

We do not open just one facility in S, but open the cheapest $\lceil \sum_{i \in S} \hat{y}_i \rceil$ facilities in S instead; let O denote this set of facilities. For each $i \in O$, we update $\hat{y}_i = 1$, and for each $i \in S - O$, we update $\hat{y}_i = 0$. (Thus, $\hat{F}$ will be reset to $\hat{F} - S$ in the next iteration.)

For each location $j \in T$, there is a total demand $\hat{d}_j$ currently assigned to locations in S, where

$$\hat{d}_j = d_j \sum_{i \in S} \hat{x}_{ij};$$

this demand will be rerouted to go only to those facilities in O. The problem of assigning the demand $\hat{d}_j$ at each location $j \in T$ to facilities in O, each of which is capable of handling total demand at most u, is an instance of the transportation problem (analogous to the discussion at the beginning of this section). Our analysis will show that *any* feasible solution suffices; however, it is natural to exploit the fact that a minimum-cost solution can be found in polynomial time. For each $i \in O$, $j \in T$, let z_{ij} be the amount of j's demand that is assigned to i by an optimal solution to this instance of the transportation problem. We update our solution by resetting $\hat{x}_{ij} = z_{ij}/\hat{d}_j$ for each $i \in O$, $j \in T$, and $\hat{x}_{ij} = 0$ for each $i \in S - O$, $j \in D$. (All other components of $\hat{x}$ remain unchanged.)

When $\hat{D}$ becomes empty, we have satisfied at least half of the demand for each location $j \in D$, by assigning it to locations for which the component of $\hat{y}$ is at least 1. To compute the solution claimed by the lemma, we will simply ignore the β_j fraction of j's demand that is still assigned to the remaining facilities in $\hat{F}$, and rescale the part of $\hat{x}$ specifying the assignment to facilities not in $\hat{F}$. That is, for each $i \notin \hat{F}$, we reset $\hat{y}_i$ to be $2\hat{y}_i$, and reset $\hat{x}_{ij}$ to be $\hat{x}_{ij}/(1 - \beta_j)$ for each $j \in D$. For each $i \in \hat{F}$, we set $\hat{y}_i = 0$ and set $\hat{x}_{ij} = 0$, for each $j \in D$.

The proof that this algorithm delivers a suitable solution follows the same outline as the proof of Lemma 2. We show that until the point at which $\hat{D}$ becomes empty, the algorithm maintains invariants

$(P1')$ $(\hat{x}, \hat{y})$ is a ρ'-relaxed solution;

$(P2')$ $\sum_{i \notin \hat{F}} f_i \hat{y}_i \leq 2 \sum_{i \notin \hat{F}} f_i \bar{y}_i$;

as well as $(P3)$ and $(P4)$.

Of course, we must also show that the algorithm is well-defined. In each iteration, we rely on an optimal solution to an auxilliary input to the transportation problem, and so we must show that a feasible solution exists to this input. An input to the transportation problem has a feasible solution provided that the total demand is no more than the total supply. That is, we must show that the total demand for T, $\sum_{j \in T} \hat{d}_j$, is not more than the total supply for O, $|O|u$. But since the solution $(\hat{x}, \hat{y})$ maintained by the algorithm is a ρ'-relaxed solution, we have that $(\hat{x}, \hat{y})$ satisfies the inequality

$$\sum_{j \in T} d_j \hat{x}_{ij} \leq \sum_{j \in D} d_j \hat{x}_{ij} \leq u \hat{y}_i, \quad \text{for each } i \in S,$$

and hence

$$\sum_{j \in T} \hat{d}_j = \sum_{j \in T} \sum_{i \in S} d_j \hat{x}_{ij} \leq \sum_{i \in S} u \hat{y}_i \leq u|O|.$$

Hence, the algorithm is well-defined. Furthermore, it is clear that this solution of the transportation problem is precisely what is required to maintain the fact that $(\hat{x}, \hat{y})$ remains a ρ'-relaxed solution. Hence, property $(P1')$ is maintained.

As in the uncapacitated case, property $(P3)$ is trivially maintained, since the algorithm never sets $\hat{x}_{ij} > 0$ while maintaining $i \in \hat{F}$. The proof for property $(P4)$ is identical to its proof in the uncapacitated case: for each $i \in S$ and $j \in T$, $c_{ij} \leq 3g_j$.

It remains only to prove that property $(P2')$ is maintained by the algorithm. This property is true initially, since the only locations $i \notin \hat{F}$ either have $\hat{y}_i = \bar{y}_i$, or else $\bar{y}_i \geq 1/2$ and $\hat{y}_i = 1$, and hence $\hat{y}_i \leq 2\bar{y}_i$. Next consider the set of locations S removed from F in some iteration. At the end of this iteration, we will set $\hat{y}_i = 1$ for each $i \in O$, and $\hat{y}_i = 0$ for each $i \in S - O$. Until this iteration, for each $i \in S$, we have not changed $\hat{y}_i$, and hence, $\hat{y}_i = \bar{y}_i$. Thus, to prove that property $(P2')$ is maintained by this iteration, it suffices to show that the inequality

$$\sum_{i \in O} f_i \leq 2 \sum_{i \in S} f_i \hat{y}_i \tag{15}$$

holds for the value of $\hat{y}$ at the start of this iteration.

Observe that since O was selected in order of cheapest fixed costs, we have that

$$\sum_{i \in O} f_i \leq \sum_{i \in S} z_i f_i, \tag{16}$$

provided $0 \leq z_i \leq 1$, for each $i \in S$, and $\sum_{i \in S} z_i = |O|$. If we set

$$z_i = \hat{y}_i \cdot \frac{|O|}{\sum_{i \in S} \hat{y}_i}, \text{ for each } i \in S, \tag{17}$$

then clearly $\sum_{i \in S} z_i = |O|$. Since $i \in \hat{F}$, $\hat{y}_i < 1/2$. Furthermore, $j' \in \hat{D}$ implies that

$$1/2 < \beta_{j'} = \sum_{i \in \hat{F}} \hat{x}_{ij'} = \sum_{i \in \hat{S}} \hat{x}_{ij'}.$$

Since $\hat{x}_{ij'} \leq \hat{y}_i$, we can conclude that

$$1/2 < \sum_{i \in \hat{S}} \hat{y}_i.$$

Hence,

$$\frac{|O|}{\sum_{i \in S} \hat{y}_i} = \frac{\lceil \sum_{i \in S} \hat{y}_i \rceil}{\sum_{i \in S} \hat{y}_i} < 2, \tag{18}$$

and so $z_i < 1$, for each $i \in S$. By combining (16), (17), and (18), we see that

$$\sum_{i \in O} f_i \leq \sum_{i \in S} z_i f_i < \sum_{i \in S} \hat{y}_i \cdot 2 \cdot f_i,$$

and so 15 holds; property $(P2')$ is maintained.

Next consider the situation when when $\hat{D}$ becomes empty. At this point, property $(P1')$ implies that $\hat{y}_i \leq \rho'$, for each $i \in F$. Since we now multiply $\hat{y}$ by at most 2, and we have ensured that there does not exist some $\hat{y}_i \in (0, 1)$, we see that the solution is a $2\rho'$-relaxed integer solution. Furthermore, since before $\hat{y}$ is multiplied by 2, we know that $(P2')$ holds, then the final solution $\hat{y}$ must have facility cost at most 4 times the cost of $\bar{y}$, and this completes the proof of the lemma. ■

Next we show how to combine Lemmas 4 and 5 to obtain a $(7, 7/2)$-approximation algorithm for the capacitated facility location problem. Let (x, y) denote an optimal solution to the linear relaxation of the capacitated facility location problem. We apply Lemma 4 to (x, y), to obtain a $1/\alpha$-relaxed solution $(\bar{x}, \bar{y})$, and then apply Lemma 5 to yield the $2/\alpha$-relaxed integer solution $(\hat{x}, \hat{y})$. For each $i \in F$ with $\hat{y}_i > 0$, we open a facility of capacity $\hat{y}_i u$ and assign to it a fraction $\hat{x}_{ij}$ of the demand d_j at location j. The facility cost of this solution is at most

$$\sum_{i \in F} f_i \hat{y}_i \leq 4 \sum_{i \in F} f_i \bar{y}_i \leq \frac{4}{\alpha} \sum_{i \in F} f_i y_i. \tag{19}$$

Furthermore, the assignment costs are at most

$$\begin{aligned} \sum_{j \in D} c_{ij} d_j \hat{x}_{ij} &\leq 3 \sum_{j \in D} d_j g_j \\ &\leq 3 \sum_{j \in D} d_j c_j(\alpha) \\ &\leq \frac{3}{1-\alpha} \sum_{j \in D} d_j \sum_{i \in F} c_{ij} x_{ij}. \end{aligned} \tag{20}$$

Hence, we have found a solution of total cost at most

$$\frac{4}{\alpha} \sum_{i \in F} f_i y_i + \frac{3}{1-\alpha} \sum_{j \in D} d_j \sum_{i \in F} c_{ij} x_{ij}.$$

If we set $\alpha = 4/7$, then we see that the total cost of the solution found is within a factor of 7 of the cost of the optimal solution to the linear relaxation. Since the solution is $2/\alpha$-relaxed, we obtain the following theorem.

Theorem 6 *For the metric capacitated facility location problem, filtering and rounding yields a $(7, 7/2)$-approximation algorithm.*

Next we turn our attention to the model in which the entire demand of each location must be assigned to the same facility. We shall call this problem the *metric capacitated location problem with unsplittable flows.* We will show that the solution found by algorithm of Theorem 6 can be adjusted to satisfy this more stringent condition, while only slightly increasing the performance guarantees.

The extension to the model with unsplittable flows is based on a rounding theorem of Shmoys & Tardos [24] for the generalized assignment problem. This theorem can be explained as follows. Suppose that there is a collection of jobs J, each of which is to be assigned to exactly one machine among the set M; if job $j \in J$ is assigned to machine $i \in M$, then it requires p_{ij} units of processing, and incurs a cost r_{ij}. Each machine $i \in M$ can be assigned jobs that require a total of at most P_i units of processing on it, and the total cost of the assignment must be at most R, where R and P_i, for each $i \in M$, are given as part of the input. The aim is to decide if there is a feasible assignment. If there is such an assignment, then there must also be a feasible solution to the following linear program, where x_{ij} is the relaxation of a 0-1 variable that indicates whether job j is assigned to machine i:

$$\sum_{i \in M} x_{ij} = 1, \quad \text{for each } j \in J; \tag{21}$$

$$\sum_{j \in J} p_{ij} x_{ij} \leq P_i, \quad \text{for each } i \in M; \tag{22}$$

$$\sum_{i \in M} \sum_{j \in J} r_{ij} x_{ij} \leq R, \tag{23}$$

$$x_{ij} \geq 0, \quad \text{for each } i \in M,\ j \in J. \tag{24}$$

Shmoys and Tardos [24] show that any feasible solution x can be rounded, in polynomial time, to an integer solution that is feasible if the right-hand side of (22) is relaxed to $P_i + \max_{j \in J} p_{ij}$.

We show next how to apply this rounding theorem to produce a solution for the capacitated version with unsplittable flows. Consider the algorithm of Theorem 6 without specifying the choice of α. Suppose that we apply the algorithm starting with an optimal solution (x, y) to the linear relaxation of the capacitated facility location problem (that is, the linear program given by (1), (2), (3), (6), (13), and (14).) The algorithm delivers a $2/\alpha$-relaxed integer solution $(\hat{x}, \hat{y})$, where the facility cost and the assignment cost are, respectively, within a factor of $4/\alpha$ and $3/(1-\alpha)$ of the analogous costs for (x, y). Let O denote the set of facilities opened by the solution $(\hat{x}, \hat{y})$; that is,

$$O = \{i \in F : \hat{y}_i \geq 1\}.$$

We can view each facility $i \in O$ as a machine of processing capacity $\hat{y}_i u$, and each location $j \in D$ as a job that requires a total of d_j units of processing (independent of the machine to which it is assigned) and incurs a cost $d_j c_{ij}$ when assigned to machine (facility) i. Therefore, if we set $M = O$, $J = D$, $P_i = \hat{y}_i u$ for each $i \in M$,

$$R = \sum_{i \in F} \sum_{j \in D} d_j c_{ij} \hat{x}_{ij},$$

as well as $p_{ij} = d_j$ and $r_{ij} = d_j c_{ij}$ for each $i \in M$, $j \in D$, then $\hat{x}$ is a feasible solution to the linear program (21)-(24).

The rounding theorem for the generalized assignment problem implies that we can round $\hat{x}$ into an integer solution $\tilde{x}$ such that each facility $i \in O$ is assigned a total demand at most $P_i + \max_{j \in D} d_j$ and the assignment cost of this solution is

$$\sum_{i \in O} \sum_{j \in D} d_j c_{ij} \tilde{x}_{ij} \leq \sum_{i \in F} \sum_{j \in D} d_j c_{ij} \hat{x}_{ij} \leq \frac{3}{1-\alpha} \sum_{i \in F} \sum_{j \in D} d_j c_{ij} x_{ij},$$

where the last inequality follows from (20). Note that, in order for there to exist a feasible solution with unsplittable flows, the demand d_j must be at most u, for each $j \in D$; hence, we assume that our instance has this property. We can conclude that the rounded solution $\tilde{x}$ assigns a total demand to each facility $i \in O$ that is at most

$$\max_{j \in D} d_j + \hat{y}_i u \leq (1 + \hat{y}_i) u.$$

Hence, if we consider the solution $(\tilde{x}, \tilde{y})$ where $\tilde{y}_i = \hat{y}_i + 1$, for each $i \in O$ and $\tilde{y}_i = \hat{y}_i$ otherwise, then we see that it is a $1 + 2/\alpha$-relaxed integer solution. Finally, since $\hat{y}_i \geq 2$ for each $i \in O$ (due to the final doubling when $\hat{D}$ becomes empty), we see that $\tilde{y}_i \leq (3/2)\hat{y}_i$, for each $i \in D$. This implies that the facility cost of $(\tilde{x}, \tilde{y})$ is

$$\sum_{i \in F} f_i \tilde{y}_i \leq (3/2) \sum_{i \in F} f_i \hat{y}_i \leq \frac{6}{\alpha} \sum_{i \in F} f_i y_i,$$

where the last inequality follows from (19). Thus, if we compare the solution $(\tilde{x}, \tilde{y})$ to the optimal fractional solution (x, y) from which we started, we have shown that the facility cost increases by at most a factor of $6/\alpha$, and the assignment cost increases by at most a factor of $3/(1-\alpha)$. If we set $\alpha = 2/3$, then both of these bounds are equal to 9, and so we obtain the following theorem.

Theorem 7 *For the metric capacitated facility location problem with unsplittable flows, filtering and rounding yields a* $(9, 4)$*-approximation algorithm.*

Khuller & Sussmann [15] have introduced the notion that one can open multiple facilities of capacity u at each location (in the context of the capacitated k-center problem). We can also obtain analogues of Theorems 6 and 7 for this variant of the capacitated facility location problem. In other words, we are now interested in obtaining solutions in which each y_i is an integer. We start by solving the linear relaxation, which is identical to the one used above, except that we replace (13) with just $y_i \geq 0$, for each $i \in F$. Lemma 4 must now be modified to reflect that we obtain a solution $(\bar{x}, \bar{y})$ that is feasible for the new linear relaxation, but still has the property that $\bar{y}_i \leq (1/\alpha) y_i$, for each $i \in F$; otherwise Lemma 4 remains unaffected. The statement of Lemma 5 must also be modified; we no longer require $(\bar{x}, \bar{y})$ to be a ρ'-relaxed integer solution, but now require that the solution $(\hat{x}, \hat{y})$ be such that each $\hat{y}_i$, $i \in F$, is an integer. This apparently stronger claim can be obtained by essentially the same proof. The only modification needed is in the initialization of $\hat{y}$: at the start of the algorithm, we set $\hat{y}_i = \lceil \bar{y}_i \rceil$ for each i such that $\bar{y}_i \geq 1/2$, and as before, we set $\hat{y}_i = \bar{y}_i$ for each i such that $\bar{y}_i < 1/2$, This also maintains property $(P2')$, since this initial rounding increases the cost incurred for each facility location $i \notin \hat{F}$ by at most a factor of 2. Of course, we no longer need to maintain property $(P1')$. By using these modified lemmas, we can obtain the following analogue of Theorems 6 and 7.

Theorem 8 *For the metric capacitated facility location problem with multiple facilities allowed, filtering and rounding yields a 7-approximation algorithm with splittable flows, and a 9-approximation algorithm with unsplittable flows.*

Since the performance guarantees have not become worse by imposing this additional restriction that the capacity used for each location is an integer multiple of u, one might wonder why we have not stated Theorems 6 and 7 in this stronger way. The reason is that by maintaining this integerized capacity, we do need to introduce a greater relaxation of the capacity bound. For example, in Theorem 6 we would produce a $2\lceil \rho' \rceil$-relaxed solution, rather than simply a $2\rho'$-relaxed solution.

4 The 2-level uncapacitated facility location problem

Another more general version of the facility location problems that we consider is the setting in which there is a 2-level hierarchy of facilities. Such 2-level facility location problems have been considered extensively in the literature (see, for example, [1, 14, 27, 28]).

We shall only consider the 2-level version of the uncapacitated problem, but it is possible to obtain similar extensions for the capacitated models as well. In the 2-level uncapacitated facility location problem, there is, as before, a set of demand points D, and a set of locations F where hub facilities can be built. However, each unit of demand at a point in D must now be shipped from a hub facility via an intermediate transit station; let E denote the set of locations at which one of these transit stations may be built. We shall consider the metric case in which the unit cost of shipping between two locations $i, j \in D \cup E \cup F$ is equal to c_{ij}; that is, these costs are non-negative, symmetric, and satisfy the triangle inequality, and so for any $i, j, k \in D \cup E \cup F$, $c_{ij} + c_{jk} \geq c_{ik}$. Each location $k \in D$ has a specified demand d_k. For each $i \in F$, the cost of building a hub facility at location i is f_i and for each $j \in E$, the cost of building a transit station at location j is e_j. Each unit of demand at location $k \in D$ must be shipped from some location $i \in F$ at which a hub is built via a location $j \in E$ at which a transit station is built, incurring a shipping cost of $c_{ij} + c_{jk}$. We shall let c_{ijk} denote the shipping cost $c_{ij} + c_{jk}$. The aim is to determine which hubs and transit stations to build so that the total building and shipping cost is minimized. We will show how to extend Theorem 3 to obtain a 4-approximation algorithm for this more general model.

First, we give a linear programming relaxation of the 2-level uncapacitated facility location problem. All of the variables in this linear program are relaxations of 0-1 decision variables, and there are three types of variables: the variables x_{ijk}, $i \in F$, $j \in E$, $k \in D$, indicate whether the demand at location k is routed through a transit station at location j from a hub facility at location i; the variables y_i, $i \in F$, indicate if a hub facility is opened at location i; and the variables z_j, $j \in E$ indicate if a transit station is opened at location j.

$$\text{minimize} \quad \sum_{i\in F} f_i y_i + \sum_{j\in E} e_j z_j + \sum_{i\in F}\sum_{j\in E}\sum_{k\in D} d_k c_{ijk} x_{ijk} \tag{25}$$

subject to

$$\sum_{i\in F}\sum_{j\in E} x_{ijk} = 1, \quad \text{for each } k \in D, \tag{26}$$

$$\sum_{j\in E} x_{ijk} \le y_i, \quad \text{for each } i \in F,\ k \in D, \tag{27}$$

$$\sum_{i\in F} x_{ijk} \le z_j, \quad \text{for each } j \in E,\ k \in D, \tag{28}$$

$$x_{ijk} \ge 0, \quad \text{for each } i \in F,\ j \in E,\ k \in D, \tag{29}$$

$$y_i \ge 0, \quad \text{for each } i \in F, \tag{30}$$

$$z_j \ge 0, \quad \text{for each } j \in E. \tag{31}$$

As in the single-level setting, we will show that any feasible solution to the linear relaxation of this integer program can be rounded to an integer solution that has objective function value at most 4 times as much. This rounding algorithm will closely resemble the algorithm used to prove Theorem 3. We first modify the definition of g-close. A feasible solution (x, y, z) to this linear relaxation is said to be g-close if it satisfies the property

$$x_{ijk} > 0 \Rightarrow c_{ijk} \le g_k. \tag{32}$$

We shall also modify the notion of an α-point. For each location $k \in D$, we sort the costs c_{ijk} over all pairs $i \in F$, $j \in E$, in nondecreasing order; if we add the associated values x_{ijk} in this sorted order, then we let $c_k(\alpha)$ be the cost associated with the first pair for which this running sum is at least α. It is straightforward to obtain the following extension of Lemma 1.

Lemma 9 *Let α be a fixed value in the interval $(0,1)$. Given a feasible fractional solution (x, y, z), we can find a g-close feasible fractional solution $(\bar{x}, \bar{y}, \bar{z})$ in polynomial time, such that*

1. $g_k \le c_k(\alpha)$, *for each* $k \in D$;
2. $\sum_{i\in F} f_i \bar{y}_i \le (1/\alpha) \sum_{i\in F} f_i y_i$;
3. $\sum_{j\in E} e_j \bar{z}_j \le (1/\alpha) \sum_{j\in E} e_j z_j$. ■

Analogous to (9), it is easy to derive that, for each $k \in D$,

$$c_k(\alpha) \le \frac{1}{1-\alpha} \sum_{i\in F}\sum_{j\in E} c_{ijk} x_{ijk}. \tag{33}$$

Next we prove the following analogue of Lemma 2.

Lemma 10 *Given a feasible fractional g-close solution $(\bar{x}, \bar{y}, \bar{z})$, we can find a feasible integer $3g$-close solution $(\hat{x}, \hat{y}, \hat{z})$ such that*

$$\sum_{i \in F} f_i \hat{y}_i + \sum_{j \in E} e_j \hat{z}_j \leq \sum_{i \in F} f_i \bar{y}_i + + \sum_{j \in E} e_j \bar{z}_j.$$

Proof: We shall first give the rounding algorithm, and then prove that the solution found has the properties claimed by the lemma. The algorithm is quite similar to the one used in the single-level uncapacitated case. We maintain a feasible fractional solution $(\hat{x}, \hat{y}, \hat{z})$ that is initialized to $(\bar{x}, \bar{y}, \bar{z})$. We will maintain a collection R of triples (i, j, k), $i \in F$, $j \in E$, $k \in D$, that have been rounded to have $\hat{x}_{ijk} = 1$ (and hence $\hat{y}_i = \hat{z}_j = 1$). Initially, $R = \emptyset$ (even if some components of $\hat{x}$ are equal to 1). We also maintain a set $\hat{D}$ of locations $k \in D$ that do not participate in any triple in R; that is,

$$\hat{D} = \{\bar{k} \in D : (i, j, k) \in R \Rightarrow \bar{k} \neq k\}.$$

In each iteration, we first find the location $k \in \hat{D}$ for which g_k is smallest; let k' denote this location. Let S denote the set of pairs (i, j) that are used to supply k' in the current solution; that is,

$$S = \{(i, j) : \hat{x}_{ijk'} > 0\}.$$

We also introduce notation for those locations that occur in some pair in S; let

$$S_F = \{i \in F : \exists j \text{ such that } x_{ijk'} > 0\}$$

and

$$S_E = \{j \in E : \exists i \text{ such that } x_{ijk'} > 0\}.$$

We will assign k' to be served by the facility-transit station pair $(i, j) \in S$ for which $f_i + e_j$ is smallest; let (i', j') denote this pair. We round the values $\{\hat{y}_i\}_{i \in S_F}$ by setting $\hat{y}_{i'} = 1$, and $\hat{y}_i = 0$ for each $i \in S_F - \{i'\}$. Similarly, we set $\{\hat{z}_j\}_{j \in S_E}$ by setting $\hat{z}_{j'} = 1$, and $\hat{z}_j = 0$ for each $j \in S_E - \{j'\}$. Let T denote the set of locations that are partially assigned by $\hat{x}$ to use locations in either S_E or S_F; that is,

$$T = \{k \in \hat{D} : \exists \hat{x}_{ijk} > 0 \text{ such that } i \in S_F \text{ or } j \in S_E\}.$$

We assign each location $k \in T$ to the facility opened at i' through the transit station located at j'; that is, for each $k \in T$, we reset $\hat{x}_{i'j'k} = 1$ and $\hat{x}_{ijk} = 0$ for each $(i, j) \neq (i', j')$; furthermore, we add (i', j', k) to R. When $\hat{D}$ becomes empty, then for each location $k \in D$, there exists (i', j') such that $\hat{x}_{i'j'k} = 1$, and so we have computed an integer solution.

We shall argue that the algorithm maintains the following properties:

$(P1)$ $(\hat{x}, \hat{y}, \hat{z})$ is a feasible fractional solution;

$(P2)$ $\sum_{i \in F} f_i \hat{y}_i + \sum_{j \in E} e_j \hat{z}_j \leq \sum_{i \in F} f_i \bar{y}_i + \sum_{j \in E} e_j \bar{z}_j$;

$(P3)$ $\hat{x}_{ijk} > 0$ and $(i, j, k) \notin R \Rightarrow c_{ijk} \leq g_k$;

$(P4)$ $\hat{x}_{ijk} > 0$ and $(i, j, k) \in R \Rightarrow c_{ijk} \leq 3g_k$;

$(P5)$ $(i, j, k) \in R$ and $\hat{x}_{ij\bar{k}} > 0 \Rightarrow (i, j, \bar{k}) \in R$;

$(P6)$ $(i, j, k) \in R \Rightarrow (\hat{x}_{i\bar{j}\bar{k}} = 0$ for each $\bar{j} \neq j$, $\bar{k} \in D$ and
$\hat{x}_{\bar{i}j\bar{k}} = 0$ for each $\bar{i} \neq i$, $\bar{k} \in D$.)

These properties certainly hold when the algorithm starts. Furthermore, if they hold when the algorithm stops (and so property $(P3)$ becomes vacuous), then we have proved Lemma 10. The proof that $(P1)$ is maintained is similar to the proof of property $(P1)$ in Lemma 2: the main observation is that whenever some $\hat{y}_i$ or $\hat{z}_j$ is set to 0, we also set all corresponding variables $\hat{x}_{ijk}$ to 0.

The new properties $(P5)$ and $(P6)$ are straightforward consequences of the way in which the rounding algorithm proceeds. To prove $(P5)$, consider two triples (i,j,k) and $(i,j,\bar{k})$ for which $\hat{x}_{ijk} > 0$ and $\hat{x}_{ij\bar{k}} > 0$ at the start of the algorithm. If either triple is placed in R, then in the same iteration, the algorithm will put the other one in R as well. Since the algorithm never changes a component of $\hat{x}$ from being 0 to being positive, this implies that property $(P5)$ holds.

To prove $(P6)$, consider two triples (i,j,k) and $(i,\bar{\jmath},\bar{k})$, where $\bar{\jmath} \neq j$, for which intially we have that $\hat{x}_{ijk} > 0$ and $\hat{x}_{i\bar{\jmath}\bar{k}} > 0$. If either of these triples is added to R, then in the same iteration, we must also set the variable corresponding to the other triple to 0; in other words, if $(i,j,k) \in R$, then $\hat{x}_{i\bar{\jmath}\bar{k}} = 0$, and so the first half of $(P6)$ has been proved. The proof of the second half is exactly analogous.

The proof that property $(P4)$ is maintained is similar to the proof given for for $(P4)$ in Lemma 2. Consider some variable $\hat{x}_{i'j'k}$ that is set to 1 during some iteration of the algorithm. However, this implies that $k \in T$, since the algorithm only sets to 1 those components of $\hat{x}$ for which the last index is in T. For the location k' used in this iteration (that is, the location in $\hat{D}$ with minimum g_k value), we have that $\hat{x}_{i'j'k'} > 0$; furthermore, (i',j',k) was not in R at the start of this iteration, and hence, by $(P3)$, $c_{i'j'k'} \leq g_{k'}$. Since $k \in T$, we know that there exists $\hat{x}_{ijk} > 0$ such that $i \in S_F$ or $j \in S_E$. We shall consider these two cases separately.

Case 1: $i \in S_F$. It follows from $i \in S_F$ that there exists $\bar{\jmath} \in E$ such that that $\hat{x}_{i\bar{\jmath}k'} > 0$. Since $k' \in \hat{D}$, this implies that $(i,\bar{\jmath},k') \notin R$, and so $c_{i\bar{\jmath}k'} \leq g_{k'}$.

We will show next that $(i,j,k) \notin R$, and hence $c_{ijk} \leq g_k$. Suppose that $\bar{\jmath} \neq j$. Since $\hat{x}_{i\bar{\jmath}k'} > 0$, it follows from $(P5)$ that $(i,j,k) \notin R$. On the other hand, suppose that $j = \bar{\jmath}$. Since $k' \in \hat{D}$, we know that $(i,\bar{\jmath},k') \notin R$, and hence, by $(P6)$, $(i,\bar{\jmath},k) = (i,j,k) \notin R$.

We wish to show that $c_{i'j'k} \leq 3g_k$. However, by the triangle inequality, we can bound $c_{i'j'k}$ by the total cost of the path from i' to j' to k', followed by the path from k' to $\bar{\jmath}$ to i, followed by the path from i to j to k. Hence,

$$c_{i'j'k} \leq c_{i'j'k'} + c_{i\bar{\jmath}k'} + c_{ijk} \leq g_{k'} + g_{k'} + g_k \leq 3g_k.$$

Case 2: $j \in S_E$. Since $j \in S_E$, there exists $\bar{\imath}$ such that $\hat{x}_{\bar{\imath}jk'} > 0$. Again, since $k' \in \hat{D}$, we know that $(\bar{\imath},j,k') \notin R$, and hence $c_{\bar{\imath}jk'} \leq g_{k'}$.

We will show next that $(i,j,k) \notin R$, and hence $c_{ijk} \leq g_k$. Suppose that $\bar{\imath} \neq i$. Since $\hat{x}_{\bar{\imath}jk'} > 0$, it follows from $(P5)$ that $(i,j,k) \notin R$. On the other hand, suppose that $i = \bar{\imath}$. Since $k' \in \hat{D}$, we know that $(i,j,k') = (\bar{\imath},j,k') \notin R$, and hence, by $(P6)$, $(i,j,k) \notin R$. Finally, we can bound $c_{i'j'k}$ by the cost of the path from i' to j' to k' followed by the edge from k' to j, followed by the edge from j to k. Hence,

$$c_{i'j'k} \leq c_{i'j'k'} + c_{\bar{\imath}jk'} + c_{ijk} \leq g_{k'} + g_{k'} + g_k \leq 3g_k,$$

and we have shown that property $(P4)$ is maintained.

To show that $(P2)$ is maintained, we note that

$$f_{i'} + e_{j'} = \min_{(i,j) \in S} f_i + e_j \leq \sum_{(i,j) \in S} (f_i + e_j)\hat{x}_{ijk},$$

where the inequality follows from the fact that the minimum of a set is no more than any convex combination of it. Finally, $\sum_{j \in E} \hat{x}_{ijk} \leq \hat{y}_i$, and $\sum_{i \in F} \hat{x}_{ijk} \leq \hat{z}_j$;

these imply that

$$\sum_{(i,j)\in S} f_i \hat{x}_{ijk} = \sum_{i\in S_F} f_i \sum_{j:(i,j)\in S} \hat{x}_{ijk} \leq \sum_{i\in S_F} f_i \hat{y}_i$$

and

$$\sum_{(i,j)\in S} e_j \hat{x}_{ijk} = \sum_{j\in S_E} e_j \sum_{i:(i,j)\in S} \hat{x}_{ijk} \leq \sum_{j\in S_E} e_j \hat{z}_j.$$

Hence

$$f_{i'} + e_{j'} \leq \sum_{i\in S_F} f_i \hat{y}_i + \sum_{j\in S_E} e_j \hat{z}_j.$$

But this inequality implies that the total of the facility cost and transit station cost of $(\hat{y}, \hat{z})$ never increases throughout the execution of the algorithm, which proves that $(P2)$ is maintained. This completes the proof of the lemma. ∎

By combining Lemmas 9 and 10 in a manner identical to the way in which Lemmas 1 and 2 were used to prove Theorem 3, we obtain the following theorem.

Theorem 11 *For the 2-level uncapacitated facility location problem, filtering and rounding yields a 4-approximation algorithm.*

5 A randomized filtering algorithm

In this section, we will show that by choosing the threshold α at random, we are able to obtain improved performance guarantees. In fact, it will also be straightforward to derandomize these algorithms. This use of randomization is very much in the same spirit as the randomization used in scheduling algorithms by Chekuri, Motwani, Natarajan, & Stein [4] and Goemans [9].

For each of the facility location models that we have discussed in the previous three sections, we have given an approximation algorithm based on a particular choice of α, but it is evident that we can also consider the algorithm for any choice of $\alpha \in (0, 1)$. For each model, the randomized algorithm is quite easy to state: we choose α uniformly in the interval $(\beta, 1)$, where β will be fixed later to optimize the algorithm's performance; then we apply the deterministic algorithm with that value of α. The intuition for cutting off the uniform distribution at some point β is that the filtering step increases the facility cost by a factor of $1/\alpha$, and so we will need to bound $E[1/\alpha]$.

We first analyze this approach for the uncapacitated (single-level) facility location problem. At the core of our analyses is the following simple lemma about the α-point of a cost function, which was first observed by Goemans [8]. Goemans used this observation to show that if one implements the α-point 1-machine scheduling algorithm of Hall, Shmoys, & Wein [11] where $\alpha \in (0, 1)$ is chosen with probability density function $f(\alpha) = 2\alpha$, then its performance guarantee improves from 4 to 2 (which had already been shown in [10] by a less direct approach). Independent of our work, Schulz & Skutella [23] also used this observation for improved performance guarantees for other scheduling models.

Lemma 12 *For each $j \in D$, $\int_0^1 c_j(\alpha)d\alpha = \sum_{i=1}^n c_{ij}x_{ij}$.*

Proof: For simplicity of notation, let us assume that

$$c_{1j} \leq c_{2j} \leq \cdots \leq c_{nj};$$

that is, the permutation π is the identity. The function $c_j(\alpha)$ is a step function, which can be described as follows. Let $i_1 < i_2 < \cdots < i_\ell$ be the indices i for which $x_{ij} > 0$. The function $c_j(\alpha)$ is equal to $c_{i_k j}$ for each α in the interval $(\sum_{s=1}^{k-1} x_{i_s j}, \sum_{s=1}^{k} x_{i_s j}]$. We wish to compute the area under this curve; for the interval from $\sum_{s=1}^{k-1} x_{i_s j}$ to $\sum_{s=1}^{k} x_{i_s j}$, this area is exactly $c_{i_k j} \cdot x_{i_k j}$. Hence the total area is exactly

$$\sum_{k=1}^{\ell} c_{i_k j} \cdot x_{i_k j} = \sum_{i=1}^{n} c_{ij} x_{ij},$$

which proves the lemma. ■

We show next how to apply this lemma. In fact, we have already proved that the filtering and rounding algorithm of Theorem 3 finds a solution of cost at most $\frac{1}{\alpha} \sum_{i \in F} f_i y_i + 3 \sum_{j \in D} d_j c_j(\alpha)$ for any given α (see equation (11)). Hence, we see that the expected cost of the solution found by the randomized algorithm is

$$\begin{aligned}
&\leq E[\frac{1}{\alpha} \sum_{i \in F} f_i y_i + 3 \sum_{j \in D} d_j c_j(\alpha)] \\
&= E[\frac{1}{\alpha}] \sum_{i \in F} f_i y_i + 3 \sum_{j \in D} d_j E[c_j(\alpha)] \\
&= (\int_{\beta}^{1} \frac{1}{1-\beta} \frac{1}{\alpha} d\alpha) \sum_{i \in F} f_i y_i + 3 \sum_{j \in D} d_j (\int_{\beta}^{1} \frac{1}{1-\beta} c_j(\alpha) d\alpha) \\
&\leq \frac{\ln(1/\beta)}{1-\beta} \sum_{i \in F} f_i y_i + \frac{3}{1-\beta} \sum_{j \in D} d_j \int_{0}^{1} c_j(\alpha) d\alpha \\
&= \frac{\ln(1/\beta)}{1-\beta} \sum_{i \in F} f_i y_i + \frac{3}{1-\beta} \sum_{j \in D} d_j \sum_{i \in F} c_{ij} x_{ij}.
\end{aligned}$$

Hence, we wish to choose β so as to minimize $\max\{\frac{\ln(1/\beta)}{1-\beta}, \frac{3}{1-\beta}\}$; that is, we set $\beta = 1/e^3$, to yield the following theorem.

Theorem 13 *For the metric uncapacitated facility location problem, randomized filtering and rounding yields an algorithm that finds a solution whose expected total cost is within a factor of* $3/(1-e^{-3}) < 3.16$ *of the optimum.*

One reinterpretation of the proof of this theorem is that for α selected at random in this manner, we have

$$E[\frac{1}{\alpha} \sum_{i \in F} f_i y_i + 3 \sum_{j \in D} d_j c_j(\alpha)] \leq \rho(\sum_{i \in F} f_i y_i + \sum_{i \in F} \sum_{j \in D} d_j c_{ij} x_{ij}),$$

where $\rho = \frac{3}{(1-e^{-3})}$. Of course, a consequence of this is that there must exist a choice for α for which this function is not greater than its expectation. Thus, if we can find the $\alpha = \alpha^*$ for which $\frac{1}{\alpha} \sum_{i \in F} f_i y_i + 3 \sum_{j \in D} d_j c_j(\alpha)$ is minimized, then by running the deterministic filtering and rounding algorithm with $\alpha = \alpha^*$, we are assured of finding a solution within the expected performance guarantee. Fortunately, the step function nature of $c_j(\alpha)$ makes this a particularly simple function to minimize; we need only check all breakpoints of all of the step functions $c_j(\alpha)$, $j \in D$. This yields the following theorem.

Theorem 14 *For the metric uncapacitated facility location problem, filtering and rounding yields a 3.16-approximation algorithm.*

The same randomization and derandomization technique can be applied to each of the theorems in this paper, yielding somewhat improved constants for each of the performance guarantees. In the capacitated case, for example, if we again choose α uniformly within the interval between $[\beta, 1]$ (where β will be chosen later), then the expected total cost of the solution found by the algorithm is at most

$$\frac{4\ln(1/\beta)}{1-\beta}\sum_{i\in F} f_i y_i + \frac{3}{1-\beta}\sum_{j\in D} d_j \sum_{i\in F} c_{ij}x_{ij},$$

where (x, y) is the optimal solution to the linear relaxation of the capacitated facility location problem. If we set $\beta = e^{-3/4}$, then we see that the expected cost is within a factor of $3/(1-e^{-3/4}) < 5.69$ of the cost of the linear relaxation optimum (x, y). The solution $(\hat{x}, \hat{y})$ found by the algorithm is also guaranteed to be $2/\alpha$-relaxed, and so the expectation of the maximum capacity used at any facility is at most $2uE[1/\alpha] \le \frac{3}{2(1-e^{-3/4})}u \le 2.85u$. When we derandomize this algorithm, by focusing on the optimal choice of α with respect to the bound on the cost of the solution, we cannot simultaneously keep the guarantee for the maximum capacity used close to its expectation, $2.85u$. However, we are choosing α within the interval $[e^{-3/4}, 1]$, and the bound $2/\alpha$ is at most $2e^{3/4} \le 4.24$ throughout this interval. Hence, we obtain the following theorem.

Theorem 15 *For the metric capacitated facility location problem, filtering and rounding yields a* $(5.69, 4.24)$*-approximation algorithm.*

The same approach can be applied to each of the theorems in this paper. In particular, for Theorem 7, the performance guarantee of $(9, 4)$ can be improved to $(3/(1-e^{-1/2}), 1+2e^{1/2}) \le (7.62, 4.29)$; for Theorem 8, the performance guarantees of 7 and 9 can be improved to 5.69 and 7.62, respectively; and for Theorem 11, the performance guarantee of 4 can be improved to 3.16.

Acknowledgments We are grateful to Michel Goemans for sharing with us his randomized analysis of the 1-machine scheduling algorithm of [11], since this ultimately led to the results in Section 5.

References

[1] K. Aardal, M. Labbé, J. Leung, and M. Queyranne. On the two-level uncapacitated facility location problem. *INFORMS J. Comput.*, 8:289–301, 1996.

[2] M. L. Balinksi. On finding integer solutions to linear programs. In *Proceedings of the IBM Scientific Computing Symposium on Combinatorial Problems*, pages 225–248. IBM, 1966.

[3] J. Bar-Ilan, G. Kortsarz, and D. Peleg. How to allocate network centers. *J. of Algorithms*, 15:385–415, 1993.

[4] C. Chekuri, R. Motwani, B. Natarajan, and C. Stein. Approximation techniques for average completion time scheduling. In *Proceedings of the 8th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 609–618, 1997.

[5] G. Cornuéjols, M. L. Fisher, and G. L. Nemhauser. Location of bank accounts to optimize float: An analytic study of exact and approximate algorithms. *Management Sci.*, 8:789–810, 1977.

[6] G. Cornuéjols, G. L. Nemhauser, and L. A. Wolsey. The uncapacitated facility location problem. In P. Mirchandani and R. Francis, editors, *Discrete Location Theory*, pages 119–171. John Wiley and Sons, Inc., New York, 1990.

[7] M. E. Dyer and A. M. Frieze. A simple heuristic for the p-center problem. *Oper. Res. Lett.*, 3:285–288, 1985.

[8] M. X. Goemans. Personal communication. 1996.

[9] M. X. Goemans. Improved approximation algorithms for scheduling with release dates. In *Proceedings of the 8th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 591–598, 1997.

[10] L. A. Hall, A. S. Schulz, D. B. Shmoys, and J. Wein. Scheduling to minimize the average completion time: on-line and off-line approximation algorithms. 1996. Submitted to *Math. Oper. Res.*

[11] L. A. Hall, D. B. Shmoys, and J. Wein. Scheduling to minimize the average completion time: on-line and off-line algorithms. In *Proceedings of the 7th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 142–151, 1996.

[12] D. S. Hochbaum. Heuristics for the fixed cost median problem. *Math. Programming*, 22:148–162, 1982.

[13] D. S. Hochbaum and D. B. Shmoys. A best possible approximation algorithm for the k-center problem. *Math. Oper. Res.*, 10:180–184, 1985.

[14] L. Kaufman, M. vanden Eede, and P. Hansen. A plant and warehouse location problem. *Operational Research Quarterly*, 28:547–557, 1977.

[15] S. Khuller and Y. J. Sussmann. The capacitated k-center problem. In *Proceedings of the 4th Annual European Symposium on Algorithms, Lecture Notes in Computer Science 1136*, pages 152–166, Berlin, 1996. Springer.

[16] A. A. Kuehn and M. J. Hamburger. A heuristic program for locating warehouses. *Management Sci.*, 9:643–666, 1963.

[17] E. L. Lawler. *Combinatorial Optimization: Networks and Matroids.* Holt, Rinehart, and Winston, New York, 1976.

[18] J.-H. Lin and J. S. Vitter. Approximation algorithms for geometric median problems. *Inform. Proc. Lett.*, 44:245–249, 1992.

[19] J.-H. Lin and J. S. Vitter. ϵ-approximations with minimum packing constraint violation. In *Proceedings of the 24th Annual ACM Symposium on Theory of Computing*, pages 771–782, 1992.

[20] A. S. Manne. Plant location under economies-of-scale-decentralization and computation. *Management Sci.*, 11:213–235, 1964.

[21] P. B. Mirchandani and R. L. Francis, eds. *Discrete Location Theory.* John Wiley and Sons, Inc., New York, 1990.

[22] G. L. Nemhauser and L. A. Wolsey. *Integer and Combinatorial Optimization.* John Wiley and Sons, Inc., New York, 1988.

[23] A. S. Schulz and M. Skutella. Randomization strikes in LP-based scheduling: Improved approximations for min-sum criteria. Technical Report 533/1996, Department of Mathematics, Technical University of Berlin, 1996.

[24] D. B. Shmoys and É. Tardos. An improved approximation algorithm for the generalized assignment problem. *Mathematical Programming*, 62:461–474, 1993.

[25] J. F. Stollsteimer. *The effect of technical change and output expansion on the optimum number, size and location of pear marketing facilities in a California pear producing region.* PhD thesis, University of California at Berkeley, Berkeley, California, 1961.

[26] J. F. Stollsteimer. A working model for plant numbers and locations. *J. Farm Econom.*, 45:631–645, 1963.

[27] D. Tcha and B. Lee. A branch-and-bound algorithm for the multi-level uncapacitated location problem. *European J. Oper. Res.*, 18:35–43, 1984.

[28] T. J. Van Roy and D. Erlenkotter. A dual based procedure for dynamic facility location. *Management Sci.*, 28:1091–1105, 1982.