IMPROVED APPROXIMATION ALGORITHMS FOR THE UNCAPACITATED FACILITY LOCATION PROBLEM

FABIÁN A. CHUDAK* AND DAVID B. SHMOYS†

Abstract. We consider the uncapacitated facility location problem. In this problem, there is a set of locations at which facilities can be built; a fixed cost f_i is incurred if a facility is opened at location i. Furthermore, there is a set of demand locations to be serviced by the opened facilities; if the demand location j is assigned to a facility at location i, then there is an associated service cost proportional to the distance between i and j, c_{ij}. The objective is to determine which facilities to open and an assignment of demand points to the opened facilities, so as to minimize the total cost. We assume that the distance function c is symmetric and satisfies the triangle inequality. For this problem we obtain a $(1 + 2/e)$-approximation algorithm, where $1 + 2/e \approx 1.736$, which is a significant improvement on the previously known approximation guarantees.

The algorithm works by rounding an optimal fractional solution to a linear programming relaxation. Our techniques use properties of optimal solutions to the linear program, randomized rounding, as well as a generalization of the decomposition techniques of Shmoys, Tardos & Aardal.

Key words. Facility location, approximation algorithms, randomized rounding

AMS subject classifications.

1. Introduction.

The study of the location of facilities to serve clients at minimum cost has been one of the most studied themes in the field of Operations Research (see, e.g., the textbook edited by Mirchandani & Francis [MF90]). In this paper, we focus on one of its simplest variants, *the uncapacitated facility location problem*, also known as *the simple plant location problem*, which has been extensively treated in the literature (see, e.g., the survey by Cornuéjols, Nemhauser & Wolsey [CNW90]). This problem can be described as follows. There is a set of potential facility locations $\mathcal{F}$; building a facility at location $i \in \mathcal{F}$ has an associated nonnegative fixed cost f_i, and any open facility can provide an unlimited amount of a certain commodity. There also is a set of clients or demand points $\mathcal{D}$ that require service; client $j \in \mathcal{D}$ has a positive demand of commodity d_j that must be shipped from one of the open facilities. If a facility at location $i \in \mathcal{F}$ is used to satisfy the demand of client $j \in \mathcal{D}$, the service or transportation cost incurred per unit is proportional to the distance from i to j, c_{ij} . The goal is to determine a subset of the set of potential facility locations at which to open facilities and an assignment of clients to these facilities so as to minimize the overall total cost, that is, the fixed costs of opening the facilities plus the total service cost. We will only consider the *metric* variant of the problem in which the distance function c is nonnegative, symmetric and satisfies the triangle inequality. Throughout this paper, a ρ*-approximation algorithm* is a polynomial-time algorithm that delivers a feasible solution within a factor of ρ of optimum. Our main result is a 1.736-approximation algorithm for the metric uncapacitated facility location problem.

In contrast to the uncapacitated facility location problem, Cornuéjols, Fisher & Nemhauser [CFN77] studied the problem in which the objective is to maximize the difference between assignment and facility costs. They showed that with this objective, the problem can be thought of as a bank account location problem as follows. A company seeks to maximize its available funds by paying bills using checks drawn on banks at different locations. More precisely, if a bill is incurred in location j, and is paid with a check from location i, there is delay in the clearing time that generates a profit, such as interest, c_{ij}. On the other hand, maintaining an account at location i has a fixed cost f_i. Thus the company would like to choose a subset of locations at which to open accounts so as to maximize the difference between the total profit from clearing times minus the total fixed cost of maintaining the accounts. Notice that even though the maximization and minimization problems are equivalent from the point of view of optimization, they are not equivalent from the point of view of approximation: the maximization problem can be approximated within a constant factor, whereas the minimization problem with an arbitrary distance function is as hard as the set cover problem, and thus a c-approximation algorithm with $c = o(\log |\mathcal{D}|)$ is unlikely to exist (see [Fei98] for details). Interestingly, Cornuéjols, Fisher & Nemhauser showed that for the maximization problem, the greedy procedure that iteratively tries to open the facility that most improves the objective function yields a solution of value within a constant factor of optimum. In contrast, Hochbaum [Hoc82] showed that the greedy algorithm is an $\Theta(\log |\mathcal{D}|)$-approximation algorithm for the minimization problem with an arbitrary

*`chudak@ifor.math.ethz.ch`. Institute for Operations Research, Swiss Federal Institute of Technology, ETH-Zürich, Switzerland. This research was done while the author was a graduate student at the School of Operations Research & Industrial Engineering, Cornell University, Ithaca, NY 14853. Research partially supported by NSF grants DMS-9505155 and CCR-9700029 and by ONR grant N00014-96-1-00500.

†`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-9912422, CCR-9700029 & DMS-9505155 and ONR grant N00014-96-1-0050O.

distance function.

By filtering a linear programming relaxation, Lin & Vitter [LV92b] also obtained an $O(\log|\mathcal{D}|)$-approximation algorithm for the uncapacitated facility location problem when the distance function c is arbitrary. In addition, they also considered the *k-median problem*, in which only k facilities can be opened, but there are no fixed costs in the objective function. They showed how to find a solution with objective function value within $(1+\epsilon)$ of optimum, but that opens $(1+1/\epsilon)O(\log|\mathcal{D}|)\,k$ facilities. Under the assumption that the distance function is a metric, they have also shown (see [LV92a]) how to find a solution of cost no more than $2(1+\epsilon)$ of optimum, opening at most $(1+1/\epsilon)k$ facilities. This latter result was the starting point of most recent work on metric facility location problems; although limited to the k-median problem, it essentially contains the core ideas needed to obtain a constant approximation algorithm for the metric uncapacitated facility location problem.

The metric uncapacitated facility location problem is known to be *NP*-hard (see [CNW90]). Very recently, Guha & Khuller [GK99] and Sviridenko [Svi98] have shown that it is *MAX-SNP*-hard. In fact, Guha & Khuller have also shown that the existence of a ρ-approximation algorithm for $\rho < 1.463$ implies that $NP \subseteq TIME(n^{O(\log\log n)})$ (see also Feige [Fei98]), and combined with an observation of Sviridenko [Svi98] such an algorithm would also imply that P=NP.

We briefly review previous work on approximation algorithms for the metric uncapacitated facility location problem. The first constant-factor approximation algorithm was given by Shmoys, Tardos & Aardal [STA97], who presented a 3.16-approximation algorithm, based on rounding an optimal solution of a classical linear programming relaxation for the problem, due to Balinski [Bal65]. This bound was subsequently improved by Guha & Khuller [GK99], who provided a 2.408-approximation algorithm. Guha & Khuller's algorithm requires a stronger linear programming relaxation, in which they add to the relaxation a facility budget constraint that separately bounds the total facility cost incurred. After running the algorithms of [STA97], they use a greedy procedure (as in [CFN77] and [Hoc82]) to improve the quality of the solution: iteratively, open one facility at a time if it improves the cost of the solution. However, since the optimal facility cost is unknown, they instead consider all 'reasonable' values of the form $(1+\epsilon)^k$ for some integer k and $\epsilon > 0$. As a consequence, they must solve a weakly polynomial number of linear programs, round the optimal solution to each, and then select the best solution found. In contrast, the 1.736-approximation algorithm presented in this paper requires the solution of just one linear program, the one introduced by Balinski [Bal65], providing as a by-product further evidence of the strength of this linear programming relaxation.

In a different line of work, Korupolu, Plaxton & Rajaraman [KPR00] showed that a simple local improvement heuristic produces a solution within a constant factor of optimum, though the best constant they obtain is 5. Very recently Arora, Raghavan & Rao [ARR98] have presented a quasi-polynomial approximation scheme for the case in which demand and facility points are in $\mathbb{R}^d$, where the dimension d is fixed, and the distance function is the usual Euclidean distance. If $d = 2$, then they obtain a polynomial approximation scheme. Their method is similar to the one used by Arora [Aro98] to produce approximation schemes for the traveling salesman problem. Thus, in contrast with the algorithm presented here, it appears to have only theoretical relevance due to the inefficiency of this approach.

Without loss of generality we shall assume that the set of potential facility locations $\mathcal{F}$ and the set of demand points $\mathcal{D}$ are disjoint; let $\mathcal{N} = \mathcal{F} \cup \mathcal{D}$, $n = |\mathcal{N}|$. Even though all our results hold for the case of arbitrary nonnegative demands, for sake of simplicity of the exposition, we will assume that each demand d_j is 1 ($j \in \mathcal{D}$); thus, the cost of assigning a client j to an open facility at location i is c_{ij}. The distance between any two points $k, \ell \in \mathcal{N}$ is $c_{k\ell}$. We assume that the $n \times n$ distance matrix $(c_{k\ell})$ is nonnegative, symmetric (that is, $c_{k\ell} = c_{\ell k}$, for all $k, \ell \in \mathcal{N}$) and satisfies the triangle inequality (that is, $c_{ij} \le c_{ik} + c_{kj}$, for all $i, j, k \in \mathcal{N}$). The simplest linear programming relaxation (from [Bal65]), which we will refer to as P, is as follows:

$$\text{Minimize} \quad \sum_{j\in\mathcal{D}}\sum_{i\in\mathcal{F}} c_{ij}x_{ij} + \sum_{i\in\mathcal{F}} f_i y_i$$

$$(\mathsf{P}) \qquad \text{subject to} \qquad \sum_{i\in\mathcal{F}} x_{ij} = 1\ , \qquad \text{for each } j \in \mathcal{D}\ , \tag{1.1}$$

$$x_{ij} \le y_i\ , \qquad \text{for each } i \in \mathcal{F},\ j \in \mathcal{D}\ , \tag{1.2}$$

$$x_{ij} \ge 0\ , \qquad \text{for each } i \in \mathcal{F},\ j \in \mathcal{D}. \tag{1.3}$$

Any 0-1 feasible solution corresponds to a feasible solution to the uncapacitated facility location problem: $y_i = 1$ indicates that a facility at location $i \in \mathcal{F}$ is open, whereas $x_{ij} = 1$ means that client $j \in \mathcal{D}$ is serviced by the facility built at location $i \in \mathcal{F}$. Inequalities (1.1) state that each demand point $j \in \mathcal{D}$ must be assigned to some facility, whereas inequalities (1.2) say that clients can only be assigned to open facilities. Thus the linear program P is indeed a relaxation of the problem. Given a feasible fractional solution $(\overline{x}, \overline{y})$, we will say that $\sum_{i \in \mathcal{F}} f_i \overline{y}_i$ and $\sum_{j \in \mathcal{D}} \sum_{i \in \mathcal{F}} c_{ij} \overline{x}_{ij}$ are, respectively, its fractional facility and service cost.

Given a feasible solution to the linear programming relaxation P, the algorithm of Shmoys, Tardos & Aardal first partitions the demand points into clusters and then for each cluster opens exactly one facility, which services all of the points in it. In their analysis, they show that the resulting solution has the property that the total facility cost is within a constant factor of the fractional facility cost and the total service cost is within a constant factor of the fractional service cost. The main drawback of this approach is that *most of the time* the solution is *unbalanced*, in the sense that the first constant is approximately three times smaller than the second.

One of the simplest ways to round an optimal solution (x^*, y^*) to the linear program P is to use the randomized rounding technique of Raghavan & Thompson [RT87] as proposed by Sviridenko [Svi97] for the special case in which all of the distances are 1 or 2. The 1.2785-approximation algorithm of [Svi97], essentially, opens a facility at location $i \in \mathcal{F}$ with probability y_i^*; and then assigns each demand point to its nearest facility. Guha & Khuller [GK99] also considered this special case, and proved a matching lower bound; that is, no better guarantee is possible unless $P = NP$. Ageev & Sviridenko [AS97] have recently shown that the randomized rounding analysis for the maximum satisfiability problem of Goemans & Williamson [GW94] can be adapted to obtain improved bounds for the maximization version of the problem studied by [CFN77].

The following simple ideas enable us to develop a rounding procedure for the linear programming relaxation P with an improved performance guarantee. We explicitly exploit optimality conditions of the linear program, and in particular, we use properties of the optimal dual solution and complementary slackness. A key element to our improvement is the use of randomized rounding in conjunction with the approach of Shmoys, Tardos & Aardal. To understand the essence of our approach, suppose that for each location $i \in \mathcal{F}$, independently, we open a facility at i with probability y_i^*. The difficulty arises when attempting to estimate the expected service cost: the distance from a given demand point to the closest open facility might be too large. However, we could always use the routing of the algorithm of Shmoys, Tardos & Aardal if we knew that each cluster has a facility open. Rather than opening each facility independently with probability y_i^*, we instead open *one* facility in each cluster with probability y_i^*. The precise algorithm is not much more complicated, but the most refined analysis of it is not quite so simple. Our algorithms are randomized, and can be easily derandomized using the method of conditional expectations. Our main result is the following.

THEOREM 1.1. *There is a polynomial-time algorithm that rounds an optimal solution to the linear programming relaxation* P *to a feasible integer solution whose value is within* $(1 + 2/e) \approx 1.736$ *of the optimal value of the linear programming relaxation* P.

Since the optimal LP value is a lower bound on the integer optimal value, the theorem yields a 1.736-approximation algorithm. The running time of algorithm is dominated by the time required to solve the linear programming relaxation P.

Since the appearance of the preliminary version of this paper [Chu98], there have been significant strides forward on research on approximation algorithms for this problem. Most notably, Jain & Vazirani [JV01] gave a primal-dual 3-approximation algorithm for this problem, which by virtue of no longer needing to solve the linear programming relaxation, yields a substantially more efficient algorithm. The starting point of our algorithm is the graph defined by positive fractional primal assignment variables, for which complementary slackness conditions are then invoked to yield tight dual constraints; the method of Jain & Vazirani first constructs a dual solution, and this serves to define an analogous graph in which the edges have the corresponding dual constraints hold with equality.

Subsequent algorithmic results have touched on several different paradigms. Sviridenko [Svi02] has provided a more sophisticated analysis of the approach used here, also incorporating a more clever use of the so-called pipeage rounding technique. Another significant contribution of this paper is a much simpler technique for proving one of the crucial probabilistic lemmas of our paper (as well as a generalization needed for this subsequent improvement).

A number of papers have given analyses that are primal-dual in flavor. Results of Jain, Mahdian, and Saberi [JMS02] and Mahdian, Markakis, Saberi, and Vazirani [MMSV01] gave a primal-dual based analysis of greedy-style algorithms. Mettu and Plaxton [MP00] gave an algorithm that, at first consideration, does not appear to be a primal-dual algorithm at all, but by defining "radii" for amortizing the fixed cost needed to open a facility at a particular location, is merely doing so implicitly. At this writing, the best known performance guarantee follows from an analysis of this type: Mahdian, Ye, and Zhang have given a 1.52-approximation algorithm [MYZ02].

Further work has also been done on local search algorithms; most notably, Charikar and Guha [CG99] have given a more sophisticated neighborhood structure that is amenable to analysis to yield stronger bounds than those obtained by Korupolu, Plaxton, and Rajaraman. Kolliopoulos and Rao [KR99] have also improved on the state of the art in constructing polynomial approximation schemes for these problems; most notably, they showed that a polynomial approximation scheme could be obtained in Euclidean metric spaces of constant dimension.

2. A Simple 4-approximation Algorithm. In this section we present a new simple 4-approximation algorithm. Even though the guarantees we will prove in the next section are substantially better, we will use most of the ideas presented here. After stating a few definitions and simple facts, we review the work of Shmoys, Tardos & Aardal [STA97], and introduce the dual of the linear programming relaxation P and properties of primal and dual solutions that will be useful throughout the paper. First we define the *neighborhood* of a demand point $k \in \mathcal{D}$.

DEFINITION 2.1. *If $(\overline{x}, \overline{y})$ is a feasible solution to the linear programming relaxation P, and $j \in \mathcal{D}$ is any demand point, the* neighborhood *of j, $\mathsf{N}(j)$, is the set of facilities that fractionally service j, that is, $\mathsf{N}(j) = \{i \in \mathcal{F} : \overline{x}_{ij} > 0\}$.*

The following fact is a simple consequence of the previous definition and inequality (1.1).

FACT 1. *For each demand point $j \in \mathcal{D}$, $\sum_{i \in \mathsf{N}(j)} \overline{x}_{ij} = 1$.*

The following definition was crucial for the algorithm of Shmoys, Tardos & Aardal [STA97].

DEFINITION 2.2. *Suppose that $(\overline{x}, \overline{y})$ is a feasible solution to the linear programming relaxation P and let $g_j \geq 0$, for each $j \in \mathcal{D}$. Then $(\overline{x}, \overline{y})$ is g-*close *if $\overline{x}_{ij} > 0$ implies that $c_{ij} \leq g_j \quad (j \in \mathcal{D},\ i \in \mathcal{F})$.*

Notice that if $(\overline{x}, \overline{y})$ is g-close and $j \in \mathcal{D}$ is any demand point, all the neighbors of j, that is, the facilities that fractionally service j, are inside the ball of radius g_j centered at j. The following lemma is from [STA97].

LEMMA 2.3. *Given a feasible g-close solution $(\overline{x}, \overline{y})$, we can find, in polynomial time, a feasible integer $3g$-close solution $(\widehat{x}, \widehat{y})$ such that*

$$\sum_{i \in \mathcal{F}} f_i \widehat{y}_i \leq \sum_{i \in \mathcal{F}} f_i \overline{y}_i \ .$$

We briefly sketch the proof below. The algorithm can be divided into two steps: a clustering step and a facility opening step. The clustering step works as follows (see Table 2.1). Let $\mathcal{S}$ be the set of demand points that have not yet been assigned to any cluster; initially, $\mathcal{S} = \mathcal{D}$. Find the unassigned demand point $j_\circ$ with smallest g_j-value and create a new cluster *centered* at $j_\circ$. Then all of the unassigned demand points that are fractionally serviced by facilities in the neighborhood of $j_\circ$ (that is, all of the demand points $k \in \mathcal{S}$ with $\mathsf{N}(k) \cap \mathsf{N}(j_\circ) \neq \varnothing$) are assigned to the cluster centered at $j_\circ$; the set $\mathcal{S}$ is updated accordingly. Repeat the procedure until all of the demand points are assigned to some cluster (i.e., $\mathcal{S} = \varnothing$). We will use $\mathcal{C}$ to denote the set of centers of the clusters. The following fact follows easily from the clustering construction

1.	$\mathcal{S} \leftarrow \mathcal{D}, \quad \mathcal{C} \leftarrow \varnothing$
2.	while $\mathcal{S} \neq \varnothing$
3.	choose $j_\circ \in \mathcal{S}$ with smallest g_j value $(j \in \mathcal{S})$
4.	create a new cluster $\mathcal{Q}$ centered at $j_\circ$, $\mathcal{C} \leftarrow \mathcal{C} \cup \{j_\circ\}$
5.	$\mathcal{Q} \leftarrow \{k \in \mathcal{S} : \mathsf{N}(k) \cap \mathsf{N}(j_\circ) \neq \varnothing\}$
6.	$\mathcal{S} \leftarrow \mathcal{S} - \mathcal{Q}$

TABLE 2.1
The clustering construction of Shmoys, Tardos & Aardal.

and the definition of neighborhood, and is essential for the success of the algorithm.

FACT 2. *Suppose that we run the clustering algorithm of Table 2.1, using any g-close solution $(\overline{x}, \overline{y})$. Then the neighborhoods of distinct centers are disjoint; that is, if j and k are centers, $j \neq k \in \mathcal{C}$, then $\mathsf{N}(j) \cap \mathsf{N}(k) = \varnothing$.*

After the clustering step, the algorithm of [STA97] opens exactly one facility per cluster. For each center $j \in \mathcal{C}$ we open the facility $i_\circ$ in the neighborhood of j, $\mathsf{N}(j)$, with smallest fixed cost f_i and assign all the demand points in the cluster of j to facility $i_\circ$. Observe that by inequalities (1.2) and Fact 1, $\sum_{i \in \mathsf{N}(j)} \overline{y}_i \geq 1$; thus $f_{i_\circ} \leq \sum_{i \in \mathsf{N}(j)} f_i \overline{y}_i$. Using Fact 2, the total facility cost incurred by the algorithm is never more than the total fractional facility cost $\sum_{i \in \mathcal{F}} f_i \overline{y}_i$.

Next consider any demand point $k \in \mathcal{D}$ and suppose it belongs to the cluster centered at $j_\circ$; let $\ell \in \mathsf{N}(k) \cap \mathsf{N}(j_\circ)$ be a common neighbor and let $i_\circ$ be the open facility in the neighborhood of $j_\circ$ (see Figure 2.1). Then, the distance from k to $i_\circ$ can be bounded by the distance from k to ℓ (which is at most g_k) plus the distance from ℓ to $j_\circ$ (which is at most $g_{j_\circ}$) plus the distance from $j_\circ$ to $i_\circ$ (which is at most $g_{j_\circ}$). Thus, the

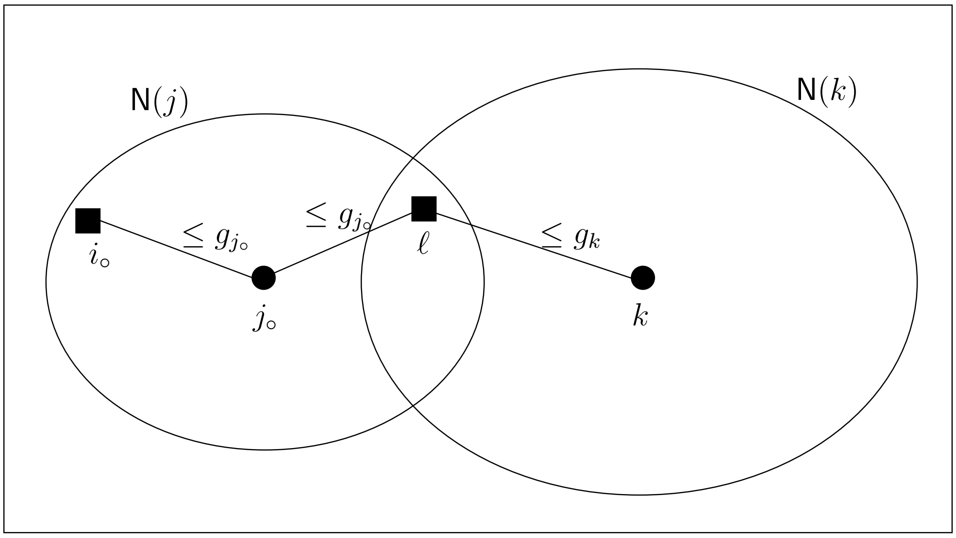

FIG. 2.1. *Bounding the service cost of k (4-approximation algorithm). The circles (●) are demand points, whereas the squares (■) are facility locations.*

distance from k to an opened facility is at most $2g_{j_\circ} + g_k$, which is at most $3g_k$, since $j_\circ$ was the remaining demand point with minimum g-value. Hence the total service cost can be bounded by $3\sum_{k\in\mathcal{D}} g_k$.

Shmoys, Tardos & Aardal used the filtering technique of Lin & Vitter [LV92b] to obtain g-close solutions (as we will describe in Section 5) and then applied Lemma 2.3 to obtain the first constant factor approximation algorithm for the problem. However, a simpler g-close solution is directly obtained by using the optimal solution to the dual linear program of P. More precisely, the dual of the linear program P is given by

$$
(\mathsf{D}) \qquad \text{subject to} \qquad
\begin{aligned}
\text{Maximize} \quad & \sum_{j\in\mathcal{D}} v_j \\
& \sum_{j\in\mathcal{D}} w_{ij} \leq f_i && \text{for each } i \in \mathcal{F} && (2.1)\\
& v_j - w_{ij} \leq c_{ij} && \text{for each } i \in \mathcal{F},\ j \in \mathcal{D} && (2.2)\\
& w_{ij} \geq 0 && \text{for each } i \in \mathcal{F},\ j \in \mathcal{D}\quad . && (2.3)
\end{aligned}
$$

Fix an optimal primal solution (x^*, y^*) and an optimal dual solution (v^*, w^*), and let LP^* be the optimal LP value. Complementary slackness gives that $x^*_{ij} > 0$ implies $v^*_j - w^*_{ij} = c_{ij}$; since $w^*_{ij} \geq 0$, we get the following lemma.

LEMMA 2.4. *If (x^*, y^*) is an optimal solution to the primal linear program P and (v^*, w^*) is an optimal solution to the dual linear program D, then (x^*, y^*) is v^*-close.*

By applying Lemma 2.3 to the optimal v^*-close solution (x^*, y^*), we obtain a feasible solution for the problem with total facility cost at most $\sum_{i\in\mathcal{F}} f_i y^*_i$ and with total service cost bounded by $3\sum_{j\in\mathcal{D}} v^*_j = 3\,\mathsf{LP}^*$. We can bound the sum of these by $4\,\mathsf{LP}^*$; thus we have a 4-approximation algorithm. Note the imbalance in bounding facility and service costs.

3. A Randomized Algorithm. After solving the linear program P, a very simple randomized algorithm is the following: open a facility at location $i \in \mathcal{F}$ with probability y^*_i *independently* for every $i \in \mathcal{F}$, and then assign each demand point to its closest open facility. Notice that the expected facility cost is just $\sum_{i\in\mathcal{F}} f_i y^*_i$, the same bound as in the algorithm of Section 2. Focus on a demand point $k \in \mathcal{D}$. If it happens that one of its neighbors has been opened, then the service cost of k would be bounded by the optimal dual variable v^*_k. However, if we are unlucky and this is not the case (an event that, as we will see, can easily be shown to occur with probability at most $1/e \approx 0.368$, where the bound is tight), the service cost of k could be very large. On the other hand, suppose that we knew, for instance, that, for the clustering computed in Section 2, k belongs to a cluster centered at j, and that one of the facilities in $\mathsf{N}(j)$ has been opened. Then in this unlucky case we could bound the service cost of k using the routing cost of the 4-approximation algorithm.

Our algorithm is also based on randomized rounding and the expected facility cost is $\sum_{i\in\mathcal{F}} f_i y^*$. However, we weaken the randomized rounding step and do *not* open facilities *independently* with probability y^*_i, but rather in a dependent way to ensure that each cluster center has *one* of its neighboring facilities opened.

Even though the algorithms presented in this section work for any g-close feasible solution, for sake of simplicity of the exposition, we will assume as in the end of Section 2 that we have a fixed optimal primal solution (x^*, y^*) and a fixed optimal dual solution (v^*, w^*), so that (x^*, y^*) is v^*-close. It is easy to see that we can assume that $y_i^* \le 1$ for each potential facility location $i \in \mathcal{F}$.

To motivate the following definition, fix a demand location $j \in \mathcal{D}$, and suppose without loss of generality that the neighborhood of j (that is, the facilities i for which $x_{ij}^* > 0$) is $\{1, \ldots, d\}$ with $c_{1j} \le c_{2j} \le \ldots \le c_{dj}$. Then it is clear that we can assume that j is assigned "as much as possible" to facility 1, then to facility 2 and so on; that is, $x_{1j}^* = y_1^*$, $x_{2j}^* = y_2^*$, …, $x_{d-1,j}^* = y_{d-1}^*$ (but maybe $x_{dj}^* < y_d^*$).

Definition 3.1. *A feasible solution $(\overline{x}, \overline{y})$ to the linear programming relaxation* P *is* complete *if $\overline{x}_{ij} > 0$ implies that $\overline{x}_{ij} = \overline{y}_i$, for every $i \in \mathcal{F}$, $j \in \mathcal{D}$.*

Thus the optimal solution (x^*, y^*) is "almost" complete, in the sense that for every $j \in \mathcal{D}$ there is at most one $i \in \mathcal{F}$ with $0 < x_{ij}^* < y_i^*$. We point out that the notion of completeness will be helpful to highlight the main ideas of the proofs and simplify the derandomization of the algorithm, although it is not essential. Next we show that any feasible solution to P can be made complete for an equivalent instance of the problem. Recall that for a feasible solution $(\overline{x}, \overline{y})$, its fractional facility and service costs are given by, respectively, $\sum_{i\in\mathcal{F}} f_i \overline{y}_i$ and $\sum_{j\in\mathcal{D}} \sum_{i\in\mathcal{F}} c_{ij}\overline{x}_{ij}$.

Lemma 3.2. *Suppose that $(\overline{x}, \overline{y})$ is a feasible solution to the linear program* P *for a given instance of the uncapacitated facility location problem $\mathcal{I}$. Then we can find, in polynomial time, an equivalent instance $\widetilde{\mathcal{I}}$ and a complete feasible solution $(\widetilde{x}, \widetilde{y})$ to its linear programming relaxation with the same fractional facility and service costs as $(\overline{x}, \overline{y})$. The new instance $\widetilde{\mathcal{I}}$ differs only by replacing each facility location by at most $|\mathcal{D}| + 1$ copies of the same location; furthermore if $(\overline{x}, \overline{y})$ is g-close, then so is $(\widetilde{x}, \widetilde{y})$.*

Proof. Pick any facility $i \in \mathcal{F}$ for which there is a demand point $j \in \mathcal{D}$ with $0 < \overline{x}_{ij} < \overline{y}_i$ (if there is no such a facility, the original solution $(\overline{x}, \overline{y})$ is complete and we are done). Among the demand points $j \in \mathcal{D}$ for which $\overline{x}_{ij} > 0$, let $j_\circ$ be the one with smallest $\overline{x}_{ij}$ value. Next create a new facility location i' which is an exact copy of i (i.e., the same fixed cost and in the same location), and set $\widetilde{y}_{i'} = \overline{y}_i - \overline{x}_{ij_\circ}$ and set $\widetilde{y}_i$ equal to $\overline{x}_{ij_\circ}$. Next for every $j \in \mathcal{D}$ with $\overline{x}_{ij} > 0$, set $\widetilde{x}_{ij} = \overline{x}_{ij_\circ} = \widetilde{y}_i$, and set $\widetilde{x}_{i'j} = \overline{x}_{ij} - \overline{x}_{ij_\circ}$ (which is nonnegative by the choice of $j_\circ$). All of the other components of $\overline{x}$ and $\overline{y}$ remain unchanged. Clearly $(\widetilde{x}, \widetilde{y})$ is a feasible solution to the linear programming relaxation of the new instance; if $(\overline{x}, \overline{y})$ is g-close so is $(\widetilde{x}, \widetilde{y})$. It is straightforward to verify that the new instance is equivalent to the old one and that the fractional facility and service costs of the solutions $(\overline{x}, \overline{y})$ and $(\widetilde{x}, \widetilde{y})$ are the same. Since the number of pairs (k, j) for which $0 < \overline{x}_{kj} < \overline{y}_k$ has decreased at least by one, and initially there can be at most $|\mathcal{D}||\mathcal{F}| \le n^2$ such pairs, n^2 iterations suffice to construct a new instance with the desired complete solution. ◻

By Lemma 3.2, we can assume that (x^*, y^*) is complete. To understand some of the crucial points of our improved algorithm we will first consider the following RANDOMIZED ROUNDING WITH CLUSTERING. Suppose that we run the clustering procedure exactly as in Table 2.1, and let $\mathcal{C}$ be the set of cluster centers. We partition the facility locations into two classes, according to whether they are in the neighborhood of a cluster center or not.

Definition 3.3. *The set of* central facility locations *$\mathcal{L}$ is the set of facility locations that are in the neighborhood of some cluster center, that is, $\mathcal{L} = \cup_{j\in\mathcal{C}} \mathsf{N}(j)$; the remaining set of facility locations $\mathcal{R} = \mathcal{F} - \mathcal{L}$ are* noncentral facility locations.

The algorithm opens facilities in a slightly more complicated way than the simplest randomized rounding algorithm described in the beginning of the section. First we open exactly one central facility per cluster as follows: independently for each center $j \in \mathcal{C}$, open neighboring facility $i \in \mathsf{N}(j)$ at random with probability x_{ij}^* (recall Fact 1). Next, we independently open each noncentral facility $i \in \mathcal{R}$ with probability y_i^*. The algorithm then simply assigns each demand point to its closest open facility.

Lemma 3.4. *For each facility location $i \in \mathcal{F}$, the probability that a facility at location i is open is y_i^*.*

Proof. If i is a noncentral facility ($i \in \mathcal{R}$), we open a facility at i with probability y_i^*. Suppose next that i is a central facility ($i \in \mathcal{L}$), and assume that $i \in \mathsf{N}(j)$, for a center $j \in \mathcal{C}$. A facility will be opened at location i only if the center j chooses it with probability x_{ij}^*; but $x_{ij}^* = y_i^*$, since (x^*, y^*) is complete. ◻

Corollary 3.5. *The expected total facility cost is* $\sum_{i\in\mathcal{F}} f_i y_i^*$.

For each demand point $k \in \mathcal{D}$, let $\overline{C}_k$ denote the fractional service cost of k, that is, $\overline{C}_k = \sum_{i\in\mathcal{F}} c_{ik} x_{ik}^*$. The expected service cost of $k \in \mathcal{D}$ is bounded in the following lemma whose proof is presented below.

Lemma 3.6. *For each demand point $k \in \mathcal{D}$, the expected service cost of k is at most $\overline{C}_k + (3/e)v_k^*$.*

Overall, since $\sum_{k\in\mathcal{D}} v_k^* = \mathsf{LP}^*$, the expected total service cost can be bounded as follows.

Corollary 3.7. *The expected total service cost is at most* $\sum_{k\in\mathcal{D}} \overline{C}_k + (3/e)\,\mathsf{LP}^*$.

By combining Corollaries 3.5 and 3.7, and noting that $\sum_{k\in\mathcal{D}} \overline{C}_k + \sum_{i\in\mathcal{F}} f_i y_i^* = \mathsf{LP}^*$, we obtain the following.

THEOREM 3.8. *The expected total cost incurred by* RANDOMIZED ROUNDING WITH CLUSTERING *is at most* $(1+3/e)\mathsf{LP}^*$.

Proof of Lemma 3.6. Fix a demand point $k \in \mathcal{D}$. For future reference, let $j_\circ$ be the center of the cluster to which k belongs; notice that $j_\circ$ always has a neighboring facility $i_\circ$ opened (i.e., $i_\circ \in \mathsf{N}(j_\circ)$), and hence its service cost is never greater than $v^*_{j_\circ}$. To gain some intuition behind the analysis, suppose first that each center in $\mathcal{C}$ shares at most one neighbor with k; that is, $|\mathsf{N}(j) \cap \mathsf{N}(k)| \le 1$, for each center $j \in \mathcal{C}$. Each neighbor $i \in \mathsf{N}(k)$ is opened with probability $y^*_i = x^*_{ik}$ *independently* in this special case. For notational simplicity suppose that $\mathsf{N}(k) = \{1, \dots, d\}$, with $c_{1k} \le \dots \le c_{dk}$. Let q be the probability that none of the facilities in $\mathsf{N}(k)$ is open. Note that $q = \prod_{i=1}^d (1-y^*_i) = \prod_{i=1}^d (1-x^*_{ik})$. One key observation is that q is "not too big": Fact 1 combined with $1-x \le e^{-x}$ $(x>0)$ implies that

$$q = \prod_{i=1}^d (1-x^*_{ik}) \le \prod_{i=1}^d e^{-x^*_{ik}} = e^{-\sum_{i=1}^d x^*_{ik}} = \frac{1}{e} \ .$$

We will bound the expected service cost of k by considering a provably worse algorithm: assign k to its closest open neighbor; if none of the neighbors of k is open, assign k to the open facility $i_\circ \in \mathsf{N}(j_\circ)$ (exactly as in Section 2). If facility 1 is open, an event which occurs with probability y^*_1, the service cost of k is c_{1k}. If, on the other hand, facility 1 is closed, but facility 2 is open, an event which occurs with probability $(1-y^*_1)y^*_2$, the service cost of k is c_{2k}, and so on. If all of the facilities in the neighborhood of k are closed, which occurs with probability q, then k is assigned to the open facility $i_\circ \in \mathsf{N}(j_\circ)$. But in this case, k is serviced by $i_\circ$, so the service cost of k is at most $2v^*_{j_\circ} + v^*_k \le 3v^*_k$ exactly as in Figure 2.1 (Section 2); in fact, this *backup routing* gives a deterministic bound: the service cost of k is *always* no more than $3v^*_k$. Thus the expected service cost of k is at most

$$\begin{aligned} & c_{1k}\, y^*_1 + c_{2k}\, y^*_2(1-y^*_1) + \cdots + c_{dk}\, y^*_d(1-y^*_1)\dots(1-y^*_{d-1}) + 3v^*_k\, q \\ & \le \sum_{i=1}^d c_{ik}x^*_{ik} + \frac{1}{e}\, 3v^*_k = \overline{C}_k + \frac{3}{e}\, v^*_k \ , \end{aligned}$$

which concludes the proof of the lemma in this special case.

Now we return to the more general case in which there are centers in $\mathcal{C}$ that can share more than one neighbor with k. We assumed that this was not the case in order to ensure that the events of opening facilities in $\mathsf{N}(k)$ were independent, but now this is no longer true for facilities $i, i' \in \mathsf{N}(k)$ that are neighbors of the same center. However, if one of i or i' is closed, the probability that the other is open increases; thus the dependencies are favorable for the analysis. The key idea of the proof is to group together those facilities that are neighbors of the same cluster center, so that the independence is retained and the proof of the special case above still works. A more rigorous analysis follows.

Let $\widehat{\mathcal{C}}$ be the subset of centers that share neighbors with k. For each center $j \in \mathcal{C}$, let $S_j = \mathsf{N}(j) \cap \mathsf{N}(k)$, and so $\widehat{\mathcal{C}} = \{j \in \mathcal{C} : S_j \ne \varnothing\}$. We have already proved the lemma when $|S_j| \le 1$, for each center $j \in \mathcal{C}$. For each center $j \in \widehat{\mathcal{C}}$, let E_j be the event that at least one common neighbor of j and k is open (see Figure 3.1). To follow the proof, for each $j \in \widehat{\mathcal{C}}$, it will be convenient to think of the event of choosing facility i in S_j as a sequence of two events: first j chooses to "open" S_j with probability $p_j = \sum_{i \in S_j} x^*_{ik}$ (i.e., event E_j occurs); and then if S_j is open, j chooses facility $i \in S_j$ with probability x^*_{ij}/p_j (which is the conditional probability of opening i given event E_j). Now let $\overline{c}_j = \sum_{i \in S_j} c_{ik}x^*_{ik}/p_j$; that is, $\overline{c}_j$ is the conditional expected distance from k to S_j, given the event E_j. For example, if $S_j = \{r, s, t\}$ are the common neighbors of j and k, the event E_j occurs when one of r, s or t is open, $p_j = x^*_{rk} + x^*_{sk} + x^*_{tk}$ and $\overline{c}_j = c_{rk}x^*_{rk}/p_j + c_{sk}x^*_{sk}/p_j + c_{tk}x^*_{tk}/p_j$. Notice that by Fact 2, the events E_j $(j \in \widehat{\mathcal{C}})$ are independent. This completes the facility central grouping. Consider the neighbors of k that are noncentral facility locations; that is, locations $i \in \mathsf{N}(k) \cap \mathcal{R}$. For each each noncentral neighbor $i \in \mathsf{N}(k) \cap \mathcal{R}$, let E_i be the event in which facility i is open, $\overline{c}_i$ be the distance c_{ik}, and $p_i = x^*_{ik}$. Next notice that *all* of the events E_ℓ are independent. It follows easily from the definitions that $\sum_\ell p_\ell = \sum_{i \in \mathcal{F}} \overline{x}_{ik} = 1$ and $\sum_\ell \overline{c}_\ell p_\ell = \overline{C}_k$.

Now we can argue essentially as in the simple case when $|S_j| \le 1$ for each center $j \in \mathcal{C}$. Assume that there are d events E_ℓ, and for notational simplicity, they are indexed by $\ell \in \{1, \dots, d\}$, with $\overline{c}_1 \le \dots \le \overline{c}_d$. Let D be the event that none of $E_1, \dots, E_d$ occurs; that is, D is precisely the event in which all the facilities in the neighborhood of k, $\mathsf{N}(k)$, are closed; let q be the probability of event D. Note that, as in the simple case, the service cost of k is never greater that its backup routing cost $3v^*_k$, in particular, this bound holds even conditioned on D. As before, we will analyze the expected service cost of a worse algorithm: k is assigned to the open neighboring facility with smallest $\overline{c}_\ell$; and if all the neighbors are closed, k is assigned

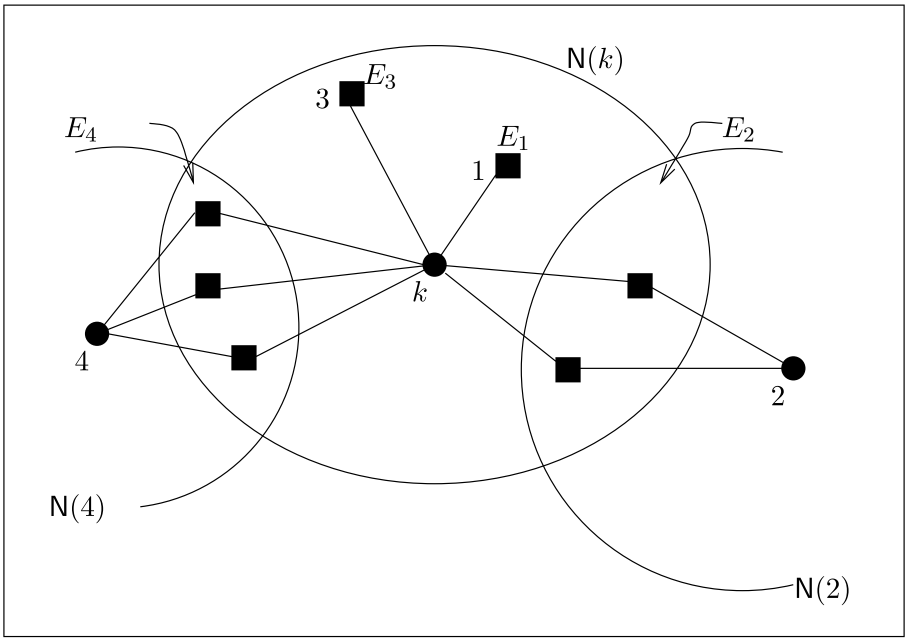

FIG. 3.1. *Estimating the expected service cost of k. Here the centers that share a neighbor with k are demand locations 2 and 4 ($\widehat{\mathcal{C}} = \{2, 4\}$). The neighbors of k that are noncentral locations are 1 and 3. Event E_2 (respectively E_4) occurs when a facility in $\mathsf{N}(k) \cap \mathsf{N}(2)$ (respectively $\mathsf{N}(k) \cap \mathsf{N}(4)$) is open, while event E_1 (respectively E_3) occurs when facility 1 (respectively 3) is open. Though there are dependencies among the neighbors of a fixed center, the events E_1, E_2, E_3 and E_4 are independent.*

through its backup routing to the open facility $i_\circ \in \mathsf{N}(j_\circ)$. If the event E_1 occurs (with probability p_1), the expected service cost of k is $\overline{c}_1$. If event E_1 does not occur, but event E_2 occurs (which happens with probability $(1-p_1)p_2$), the expected service cost of k is $\overline{c}_2$, and so on. If we are in the complementary space D, which occurs with probability $q = \prod_{\ell=1}^{d}(1-p_\ell)$, the service cost of k is never greater than its backup service cost $3v_k^*$. Thus the expected service cost of k can be bounded by

$$\overline{c}_1\, p_1 + \overline{c}_2\,(1-p_1)p_2 + \cdots + \overline{c}_d\,(1-p_1)\ldots(1-p_{d-1})p_d + 3v_k^*\, q \ . \tag{3.1}$$

To prove the lemma we bound the first d terms of (3.1) by $\overline{C}_k$, and q by $1/e$. □

Notice than even though the clustering construction is deterministic, the backup service cost of k (that is, the distance between k and the facility open in $\mathsf{N}(j_\circ)$) is a random variable B. In the proof above, we used the upper bound $B \le 3v_k^*$. In fact, the proof of Lemma 3.6 shows that the expected service cost of k is no more than $\overline{C}_k + q\,\mathsf{E}[B|D]$, where D is the event in which no neighbor k is open, as in the proof of the lemma. As can be easily seen, the upper bound used for equation (3.1) is not tight. In fact, we can get an upper bound of $(1-q)\,\overline{C}_k + q\,\mathsf{E}[B|D]$ as follows. First note the following simple probabilistic interpretation of the first d terms of (3.1). Let Z_ℓ ($\ell = 1, \ldots, d$) be independent 0-1 random variables with $\mathsf{Prob}\{Z_\ell = 1\} = p_\ell$. Consider the set of indices for which Z_ℓ is 1, and let Z be the minimum $\overline{c}_\ell$ value in this set of indices; if all of the Z_ℓ are 0, Z is defined to be 0. Then the expected value of Z is exactly equal to the first d terms of (3.1). Given a set of numbers S, we will use $\min_\circ(S)$ to denote the smallest element of S if S is nonempty, and 0 if S is empty, so that $Z = \min_\circ\{\overline{c}_\ell Z_\ell : \ell = 1, \ldots, d \text{ and } Z_\ell = 1\}$. The following intuitive probability lemma, whose proof is given in Section 6, provides a bound on the first d terms of (3.1). (A much simpler proof of this lemma, based on the Chebyshev Integral Inequality, was observed by Sviridenko [Svi02].)

LEMMA 3.9. *Suppose that $0 \le \overline{c}_1 \le \ldots \le \overline{c}_d$, $p_1, \ldots, p_d > 0$, with $\sum_{\ell=1}^{d} p_\ell = 1$. Let $Z_1, \ldots, Z_d$ be 0-1 independent random variables, with $\mathsf{Prob}\{Z_\ell = 1\} = p_\ell$; let $\overline{C} = \sum \overline{c}_\ell p_\ell$. Then*

$$\mathsf{E}\left[\min_{\{\ell : Z_\ell = 1\}}{}_\circ\ \overline{c}_\ell Z_\ell + \overline{C}\prod_{\ell=1}^{d}(1-Z_\ell)\right] \le \overline{C} \ .$$

Applying the lemma to the first d terms of equation (3.1), since $\mathsf{E}[\prod_{\ell=1}^{d}(1-Z_\ell)] = \prod_{\ell=1}^{d}(1-p_\ell) = q$, we have that

$$\overline{c}_1 p_1 + \overline{c}_2(1-p_1)p_2 + \cdots + \overline{c}_d(1-p_1)\ldots(1-p_{d-1})p_d \le \overline{C}_k(1-q) \ . \tag{3.2}$$

Thus we have proved the following.

LEMMA 3.10. *For each demand point $k \in \mathcal{D}$, the expected service cost of k is at most* $(1-q)\,\overline{C}_k + q\,\mathsf{E}[B|D]$.

Finally we introduce the last idea that leads to the $(1+2/e)$-approximation algorithm. In Figure 2.1, we have bounded the distance from the center $j_\circ$ to the open facility $i_\circ$, $c_{i_\circ j_\circ}$, by $v^*_{j_\circ}$. However, $i_\circ$ is selected (by the center $j_\circ$) with probability $x^*_{i_\circ j_\circ}$ and, thus, the expected length of this leg of the routing is $\sum_{i\in\mathcal{F}} c_{ij_\circ} x^*_{ij_\circ} = \overline{C}_{j_\circ}$, which in general is smaller than the estimate $v^*_{j_\circ}$ used in the proof of Lemma 3.6. Thus, to improve our bounds, we slightly modify the clustering procedure by changing line 3 of Table 2.1 to

3'.	choose $j_\circ \in \mathcal{S}$ with smallest $v^*_j + \overline{C}_j$ value $(j \in \mathcal{S})$

We will call the modified algorithm RANDOMIZED ROUNDING WITH IMPROVED CLUSTERING. Notice that Lemmas 3.4 and 3.10 are unaffected by this change. We will show that the modified rule 3' leads to the bound $\mathsf{E}[B|D] \le 2v^*_k + \overline{C}_k$, improving on the bound of $3v^*_k$ we used in the proof of Lemma 3.6.

LEMMA 3.11. *If we run* RANDOMIZED ROUNDING WITH IMPROVED CLUSTERING, *the conditional expected backup service cost of k,* $\mathsf{E}[B|D]$, *is at most* $2v^*_k + \overline{C}_k$.

Proof. Suppose that the clustering partition assigned k to the cluster with center $j_\circ$. Deterministically, we divide the proof into two cases.

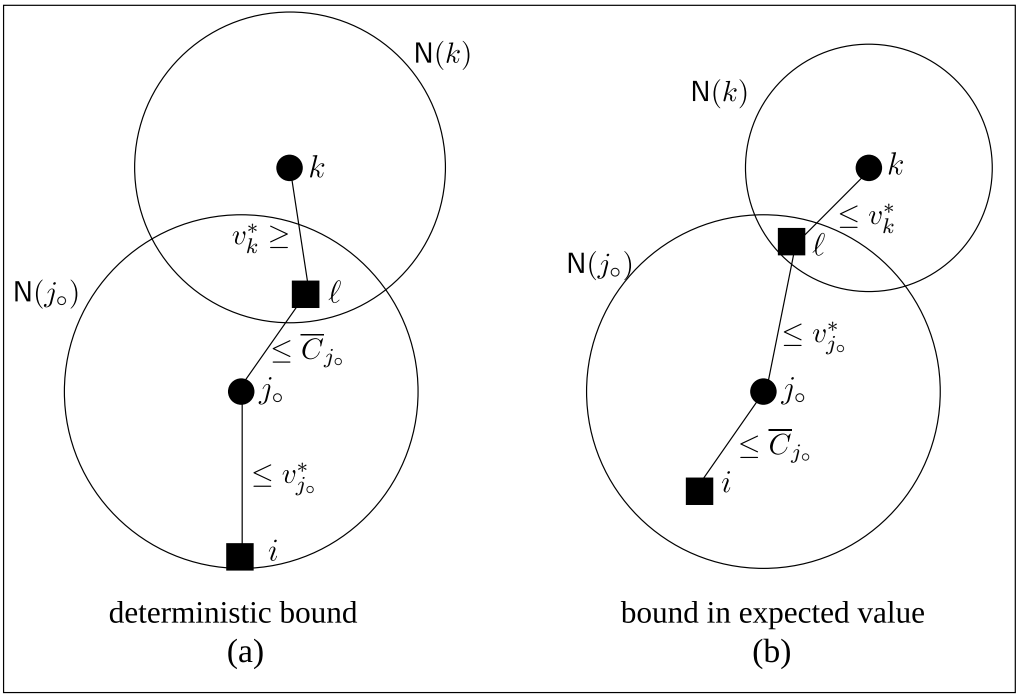

FIG. 3.2. *Bounding the backup service cost of k.*

Case 1. Suppose that there is a facility $\ell \in \mathsf{N}(k)\cap\mathsf{N}(j_\circ)$, such that $c_{\ell j_\circ} \le \overline{C}_{j_\circ}$ (see Figure 3.2(a)). Let i be the facility in $\mathsf{N}(j_\circ)$ that was opened by $j_\circ$; notice that $c_{ij_\circ} \le v^*_{j_\circ}$ (because $(\overline{x},\overline{y})$ is v^*-close). Then the service cost of k is at most $c_{ik} \le c_{k\ell} + c_{\ell j_\circ} + c_{j_\circ i}$, which using again that (x^*, y^*) is v^*-close, is at most $v^*_k + c_{\ell j_\circ} + v^*_{j_\circ} \le v^*_k + \overline{C}_{j_\circ} + v^*_{j_\circ} \le \overline{C}_k + 2v^*_k$, where the last inequality follows from the fact that the center has the minimum $(\overline{C}_j + v^*_j)$ value. In this case, we have a (deterministic) bound, $B \le \overline{C}_k + 2v^*_k$.

Case 2. Assume that $c_{\ell j_\circ} > \overline{C}_{j_\circ}$ for every $\ell \in \mathsf{N}(k)\cap\mathsf{N}(j_\circ)$ (see Figure 3.2(b)). First note that when we do not condition on D (i.e., that no facility in $\mathsf{N}(k)$ is open), then the expected length of the edge from $j_\circ$ to the facility that $j_\circ$ has selected is $\overline{C}_{j_\circ}$. However, we are given that all of the facilities in the neighborhood of k are closed, but in this case, all of these facilities that contribute to the expected service cost of $j_\circ$ (the facilities in $\mathsf{N}(k)\cap\mathsf{N}(j_\circ)$) are at distance greater than the average $\overline{C}_{j_\circ}$. Thus the conditional expected service cost of $j_\circ$ is at most the unconditional expected service cost of $j_\circ$, $\overline{C}_{j_\circ}$. It follows then that if $\ell \in \mathsf{N}(k)\cap\mathsf{N}(j_\circ)$, the conditional expected service cost of k is at most $\overline{C}_{j_\circ} + c_{j_\circ \ell} + c_{\ell k} \le \overline{C}_{j_\circ} + v^*_{j_\circ} + v^*_k \le \overline{C}_k + 2v^*_k$, where again the last inequality from the fact that $\overline{C}_{j_\circ} + v^*_{j_\circ} \le \overline{C}_k + v^*_k$. Hence, $\mathsf{E}[B|D] \le \overline{C}_k + 2v^*_k$ in this case, too. ∎

Thus, using Lemmas 3.10 and 3.11, the expected service cost of k can be bounded by

$$\overline{C}_k(1-q)+q\,(2v_k^*+\overline{C}_k)=\overline{C}_k+2q\,v_k^*\leq\overline{C}_k+\frac{2}{e}v_k^*\ ,$$

where once again we bound q by $1/e$.

Corollary 3.12. *The expected total service cost of* randomized rounding with improved clustering *is at most* $\sum_{k\in\mathcal{D}}\overline{C}_k+(2/e)\sum_{k\in\mathcal{D}}v_k^*$.

Combining Corollaries 3.5 and 3.12, randomized rounding with improved clustering produces a feasible solution with expected cost no greater than

$$\sum_{i\in\mathcal{F}}f_iy_i^*+\sum_{k\in\mathcal{D}}\overline{C}_k+\frac{2}{e}\sum_{k\in\mathcal{D}}v_k^*=\left(1+\frac{2}{e}\right)\mathsf{LP}^*\approx 1.736\,\mathsf{LP}^*.$$

Thus we have proved the following theorem.

Theorem 3.13. *There is a polynomial-time randomized algorithm that finds a feasible solution to the uncapacitated facility location problem with expected cost at most* $(1\ +\ 2/e)\ \mathsf{LP}^*$.

As a consequence of the theorem, we obtain the following corollary on the quality of the value of the linear programming relaxation of [Bal65].

Corollary 3.14. *The optimal value of the linear programming relaxation* P *is within a factor of 1.736 of the optimal cost.*

This improves on the previously best known factor of 3.16 presented in [STA97].

To finish the proof of Theorem 1.1 we will show in the next section that the algorithm of Theorem 3.13 can be derandomized using standard methods.

4. Derandomization. In this section we show how to derandomize the algorithm of Theorem 3.13. To this end, we will use the method of conditional expectations due to Erdös & Selfridge [ES73] (see also Spencer [Spe87]) that works as follows. Suppose that we have m 0-1 random variables $U_1,\ldots,U_m$, and know that $\mathsf{E}[F]=G$, where $F=g(U_1,\ldots,U_m)$ for a nonnegative real-valued function g of m variables. Our task is to determine a set of values $\overline{u}_1,\ldots,\overline{u}_m$ for the random variables $U_1,\ldots,U_m$ such that $g(\overline{u}_1,\ldots,\overline{u}_m)\leq G$. We wish to decide whether to set U_1 to 0 or 1. The expected value $\mathsf{E}[F]$ is a convex combination of the conditional expectations $\mathsf{E}[F|U_1=0]$ and $\mathsf{E}[F|U_1=1]$, more precisely,

$$\mathsf{E}[F]=\mathsf{Prob}\{U_1=0\}\,\mathsf{E}[F|U_1=0]+\mathsf{Prob}\{U_1=1\}\,\mathsf{E}[F|U_1=1].$$

We would have certainly made the right decision if the conditional expectation decreases. Thus we set $\overline{u}_1$ to 1 if $\mathsf{E}[F|U_1=1]\leq\mathsf{E}[F|U_1=0]$, and 0 otherwise. Note that $\mathsf{E}[F|U_1=\overline{u}_1]\leq\mathsf{E}[F]=G$. The process continues inductively. Suppose that we have already decided on the values $\overline{u}_1,\ldots,\overline{u}_t\in\{0,1\}$ so that the conditional expected value $\mathsf{E}[F|U_1=\overline{u}_1,\ldots,U_t=\overline{u}_t]$ is at most G. Now $\mathsf{E}[F|U_1=\overline{u}_1,\ldots,U_t=\overline{u}_t]$ is a convex combination of $\mathsf{E}[F|U_1=\overline{u}_1,\ldots,U_t=\overline{u}_t,U_{t+1}=0]$ and $\mathsf{E}[F|U_1=\overline{u}_1,\ldots,U_t=\overline{u}_t,U_{t+1}=1]$. Again we choose to set $\overline{u}_{t+1}$ to 1 if the expected value decreases, that is, if $\mathsf{E}[F|U_1=\overline{u}_1,\ldots,U_t=\overline{u}_t,U_{t+1}=1]\leq\mathsf{E}[F|U_1=\overline{u}_1,\ldots,U_t=\overline{u}_t,U_{t+1}=0]$, and 0 otherwise. Clearly, at termination, we obtain a set of 0-1 values $\overline{u}_1,\ldots,\overline{u}_m$ such that $g(\overline{u}_1,\ldots,\overline{u}_m)\leq G$ as desired. Notice that we only need to compute $2m$ conditional expected values. However for this process to work we have to be able to compute all the intermediary conditional expectations efficiently (i.e., in polynomial time).

We first show that the analysis of randomized rounding with improved clustering also provides a random variable W, a *pessimistic estimator* [Rag88], such that the cost of the solution delivered by the algorithm is at most W, and the bounds we obtained can be read off as a bound on the expected value of W, that is, $\mathsf{E}[W]\leq(1+2/e)\mathsf{LP}^*$. Furthermore the expected value of W can be computed exactly. The upper bound W will let us use the method of conditional expectations to derandomize the algorithm.

The random variable W is precisely the cost of the worse algorithm we used to prove Theorem 3.13; this algorithm either assigns each demand k to the closest open neighbor (in the $\overline{c}$ sense as in the proof of Lemma 3.6) or, if none of its neighbors is open, it assigns k to the backup facility of the cluster to which k belongs. More concretely, for each potential facility location $i\in\mathcal{F}$ let U_i be the 0-1 random variable that is 1 exactly when a facility at location i is open. Notice that the random variables U_i are not independent in general, and that, since we assumed that the feasible solution $(\overline{x},\overline{y})$ was complete, the expected value of U_i is $\overline{y}_i$. It follows immediately that the facility cost of the randomized algorithm is $\sum_{i\in\mathcal{F}}f_iU_i$.

Next we need to find an upper bound for the service costs expressed in terms of the U_i's. We fix a demand location $k\in\mathcal{D}$ and carefully revise the bounds on the expected service cost of k given in the previous section. Recall the notation used in the proof of Lemma 3.6. For each $\ell\in\{1,\ldots,d\}$, we extend the definition of S_ℓ when ℓ is a neighbor of k by just setting S_ℓ equal to $\{\ell\}$. Note that in any case

$\overline{c}_\ell p_\ell = \sum_{i \in S_\ell} c_{ik}\overline{y}_i = \mathsf{E}[\sum_{i \in S_\ell} c_{ik}U_i]$. Now let $P_\ell = \sum_{i \in S_\ell} U_i$ and $T_\ell = \sum_{i\in S_\ell} c_{ik}U_i$, for $\ell = 1,\ldots,d$; and let $Q = \prod_{\ell=1}^{d}(1-P_\ell)$. Observe that because of the dependencies, the random variables P_ℓ are 0-1 with $p_\ell = \mathsf{E}[P_\ell] = \sum_{i\in S_\ell} x^*_{ik}$. Now suppose that according to Lemma 3.11 we are in Case 1. Then the arguments given to bound the expected service cost of k in effect say that the service cost of k is at most

$$T_1 + T_2(1-P_1) + \cdots + T_d(1-P_1)\ldots(1-P_{d-1}) + Q(v^*_k + v^*_{j_\circ} + \overline{C}_{j_\circ}) \ ; \tag{4.1}$$

let Z_k denote this upper bound. Since the factors of the products in each term are independent, it is straightforward to verify that the bound on the service cost of k of Theorem 3.13 implies $\mathsf{E}[Z_k] \le \overline{C}_k + (2/e)v^*_k$.

For Case 2, the service cost of k is at most

$$T_1 + T_2(1-P_1) + \cdots + T_d(1-P_1)\ldots(1-P_{d-1}) + Q(v^*_k + v^*_{j_\circ} + \sum_{i\in \mathsf{N}(j_\circ)-\mathsf{N}(k)} c_{ij}U_i) \ ; \tag{4.2}$$

which as before will be denoted by Z_k. This case is slightly more complicated than Case 1, since before we used the fact that our upper bound was deterministic. Now dependencies have to be taken into account. As in Case 1, the expected value of the first d terms can be upper bounded, exactly as in the proof of Theorem 3.13, by $(1-q)\overline{C}_k$. For the last term, notice first that the dependencies imply that $P_{j_\circ}U_i = 0$ (and thus $(1-P_{j_\circ})U_i = U_i$) for each $i \in \mathsf{N}(j_\circ) - \mathsf{N}(k)$, so that if $Q' = \prod_{j\ne j_\circ}(1-P_j)$,

$$Q \sum_{i\in \mathsf{N}(j_\circ)-\mathsf{N}(k)} c_{ij}U_i = Q'\,(1-P_{j_\circ}) \sum_{i\in \mathsf{N}(j_\circ)-\mathsf{N}(k)} c_{ij}U_i = Q' \sum_{i\in \mathsf{N}(j_\circ)-\mathsf{N}(k)} c_{ij}U_i \ .$$

Now the two factors on the right-most expression are independent, and

$$\mathsf{E}\left[\sum_{i\in \mathsf{N}(j_\circ)-\mathsf{N}(k)} c_{ij}U_i\right] \le (1-p_{j_\circ})\overline{C}_{j_\circ} \ ,$$

since by Case 2, $c_{ij_\circ} > \overline{C}_{j_\circ}$ for all $i \in S_{j_\circ}$. Thus the expected value of the last term of (4.2) can be bounded by $q(v^*_k + v^*_{j_\circ} + \overline{C}_{j_\circ})$. Putting the pieces together, we have again that $\mathsf{E}[Z_k] \le \overline{C}_k + (2/e)v^*_k$ as needed. Notice also that, as in Case 1, Z_k can be written as a sum in which each term is the product of independent random variables.

Clearly if $W = \sum_{i\in\mathcal{F}} f_iU_i + \sum_{k\in\mathcal{D}} Z_k$, the cost of the randomized algorithm can be bounded by W, and $\mathsf{E}[W] \le (1+2/e)\mathsf{LP}^*$. Since W is the cost of a worse algorithm, if we were able to find 0-1 values for the U_i's so that the corresponding value of W is at most its expected value $\mathsf{E}[W]$, we would have a feasible solution to the problem whose cost is at most $(1+2/e)\mathsf{LP}^*$, thus proving Theorem 1.1. As mentioned earlier, to find such values for the U_i's we apply the method of conditional expectations.

We need to explain how to compute the conditional expected values of W. To understand how to apply the method of conditional expectations suppose first that *all* of the U_i's are independent random variables. Now the conditional expected value $\mathsf{E}[W|U_i = 0]$ (respectively $\mathsf{E}[W|U_i = 1]$) is easy to compute: simply replace U_i by 0 (respectively by 1) in the expression of W, and compute the expected value directly. It is also easy to see that we can substitute each U_i by $u_i \in \{0,1\}$ for $i \in \mathcal{E}$, for any subset $\mathcal{E} \subseteq \mathcal{F}$ and compute $\mathsf{E}[W|U_i = u_i \ (i \in \mathcal{E})]$. Thus, if we did not have dependencies, the derandomization of the algorithm is quite simple. However, if for instance U_s and U_t $(s,t \in \mathcal{F},\ s \ne t)$ are dependent, when conditioning on the event $\{U_s = 1\}$, we cannot just replace U_s by 1 in the expression of W, since the value of U_t might also be affected. This apparent difficulty can be easily overcome in our case, since the dependencies imply that $U_t = 0$, whenever $U_s = 1$. Next we describe the derandomization process with more detail.

We first consider the noncentral facilities. If i is a noncentral facility location, $i \in \mathcal{R}$, it is easy to compute the conditional expected value of W given that $U_i = u$ $(u = 0,1)$: simply substitute U_i by u in the expression of W and take expected values. In the same way, we can also compute $\mathsf{E}[W|U_i = u_i \ (i \in \mathcal{E})]$, for any subset $\mathcal{E} \subseteq \mathcal{R}$, where $u_i \in \{0,1\}$ $(i \in \mathcal{E})$. Now we can determine the values of $\overline{u}_i$ for $i \in \mathcal{R}$ applying the method of conditional expectations as described earlier. For notational simplicity, we will assume that $\mathcal{R} = \{1,...,r\}$. In the first step, we set $\overline{u}_1$ equal to 1 (or equivalently open facility 1) only if $\mathsf{E}[W|U_1 = 1] \le \mathsf{E}[W|U_1 = 0]$; otherwise we set $\overline{u}_1$ equal to 0. Inductively, if we already know $\overline{u}_1,\ldots,\overline{u}_{h-1}$, in step h we set U_h to 1 if $\mathsf{E}[W|U_1 = \overline{u}_1,\ldots,U_{h-1} = \overline{u}_{h-1}, U_h = 1] \le \mathsf{E}[W|U_1 = \overline{u}_1,\ldots,U_{h-1} = \overline{u}_{h-1}, U_h = 0]$, and to 0 otherwise. When $h = r$, we have values $\overline{u}_1,\ldots,\overline{u}_r$ such that if $\overline{W}$ is the conditional random variable $W|U_1 = \overline{u}_1,\ldots,U_r = \overline{u}_r$, then $\mathsf{E}[\overline{W}] \le \mathsf{E}[W]$.

In what follows we find values for the remaining variables U_i, that is, decide which central facilities to open, in such a way that at termination the cost of the solution is at most $\mathsf{E}[\overline{W}]$. Fix a center $j \in \mathcal{C}$. We wish to decide which neighboring facility to open. For each neighbor $i \in \mathsf{N}(j)$ we can compute the conditional expectation of $\overline{W}$ given that U_i is 1 as follows. Since $U_i = 1$, for all the other neighbors i' of j it must be that $U_{i'} = 0$. Hence we just replace these values in the formula of $\overline{W}$ and compute the corresponding expectation. The key observation is that, since we open exactly one facility in the neighborhood of j, $\mathsf{E}[\overline{W}]$ is a convex combination of the conditional expected values $\mathsf{E}[\overline{W}|U_i = 1]$ for $i \in \mathsf{N}(j)$, that is,

$$\mathsf{E}[\overline{W}] = \sum_{i \in \mathsf{N}(j)} \mathsf{Prob}\{U_i = 1\}\, \mathsf{E}[\overline{W}|U_i = 1] \ .$$

Thus we open the neighboring facility $i_\circ \in \mathsf{N}(j)$ for which the conditional expected value is smallest, more precisely, we set $\overline{u}_{i_\circ}$ equal to 1, and $\overline{u}_i$ equal to 0 for $i \in \mathsf{N}(j), i \neq i_\circ$. Using now that the neighborhoods of distinct centers are disjoint, we can essentially argue as in the simple case when all the variables were independent. We repeat the process inductively; at each step we treat a new center and decide which neighboring facility to open, so that the conditional expected value never increases.

Finally notice that there are $|\mathcal{R}| + |\mathcal{C}|$ steps, and overall we need only to compute $1 + 2|\mathcal{R}| + |\mathcal{L}|$ expected values. The most expensive computation that dominates the whole derandomization process is to find, initially, the expected value of W, which takes $O(|\mathcal{D}||\mathcal{F}| \log|\mathcal{F}|)$ arithmetic operations.

5. Extensions. To motivate the results of this section, suppose that the contribution of the fractional facility cost to the optimal fractional cost is very small. If we run RANDOMIZED ROUNDING WITH IMPROVED CLUSTERING and focus on the analysis, we would have an imbalance between the facility and service cost upper bounds. Hence, it is intuitively clear that we could afford to open facilities with higher probability to balance out the bounds. In this section, we will show how to do this in general, providing a more refined performance guarantee that depends on information related to the cost distribution of the optimal fractional solution.

A standard technique to improve the performance of randomized rounding consists of using a nontrivial mapping of the optimal fractional solution into probabilities. One of the most common approaches boosts all the probabilities by a factor of γ, for a fixed parameter $\gamma > 0$. For instance, the simplest randomized rounding algorithm would open facility $i \in \mathcal{F}$ with probability $\min\{\gamma y_i^*, 1\}$. These ideas can also be applied to our randomized algorithm in a simple fashion that we describe below.

First of all, as mentioned in Section 3, the proof of Theorem 3.13 is valid for any g-close solution, that is, if $(\overline{x}, \overline{y})$ is g-close, RANDOMIZED ROUNDING WITH IMPROVED CLUSTERING produces a feasible solution with expected cost at most

$$\sum_{i\in\mathcal{F}} f_i \overline{y}_i + \sum_{j\in\mathcal{D}} \sum_{i\in\mathcal{F}} c_{ij}\overline{x}_{ij} + \frac{2}{e}\sum_{j\in\mathcal{D}} g_j.$$

In fact, it is also easy to see that the derandomization of Section 4 also carries over to this more general setting.

Now suppose again that $(\overline{x}, \overline{y})$ is g-close and complete, and let $\gamma \geq 1$. We next describe the algorithm γ-RANDOMIZED ROUNDING WITH IMPROVED CLUSTERING, which is a variant of RANDOMIZED ROUNDING WITH IMPROVED CLUSTERING. The only difference is that now we open facilities with higher probabilities. Each noncentral facility $i \in \mathcal{R}$ is opened independently with probability $\min\{\gamma\overline{y}_i, 1\}$. As in RANDOMIZED ROUNDING WITH IMPROVED CLUSTERING, each center $j \in \mathcal{C}$ opens one facility in its neighborhood $i \in \mathsf{N}(j)$ with probability $\overline{x}_{ij}$, in a *first phase.* Additionally, if a facility $i \in \mathsf{N}(j)$ has not been opened, it is now opened in a *second phase* with probability $\min\{\gamma\overline{y}_i - \overline{x}_{ij}, 1\} = \min\{\gamma\overline{y}_i - \overline{y}_i, 1\}$. The new algorithm has a guarantee given by the following theorem, whose proof is a minor variation of the proof of Theorem 3.13.

THEOREM 5.1. *For each $\gamma \geq 1$, the expected cost of the solution produced by* γ-RANDOMIZED ROUNDING WITH IMPROVED CLUSTERING *is at most*

$$\gamma\sum_{i\in\mathcal{F}} f_i \overline{y}_i + \sum_{j\in\mathcal{D}} \sum_{i\in\mathcal{F}} c_{ij}\overline{x}_{ij} + \frac{2}{e^\gamma}\sum_{j\in\mathcal{D}} g_j \ .$$

Proof. For simplicity, we will analyze a worse algorithm in which we are allowed to open more than one facility at each location $i \in \mathcal{F}$. More precisely, for each facility $i \in \mathcal{L}$, if i is in the neighborhood of center j, we open a facility at location i in the second phase with probability $\min\{\gamma\overline{y}_i - \overline{y}_i, 1\}$ independently on whether there is already an open facility at i (from the first phase); in this case, we of course take into account the extra facility cost.

First we estimate the expected total facility cost as in Lemma 3.4. If $i \in \mathcal{R}$ is a noncentral facility, i is open with probability $\min\{\gamma\overline{y}_i, 1\}$, giving a contribution to the expected total facility cost of $f_i \min\{\gamma\overline{y}_i, 1\} \leq$

$\gamma\, f_i\overline{y}_i$. Now if $i \in \mathcal{L}$ is a central facility location, $i \in \mathsf{N}(j)$ for $j \in \mathcal{C}$, a facility is open at i first with probability $\overline{y}_i = \overline{x}_{ij}$, contributing $f_i\overline{y}_i$. In addition, another facility is open at i with probability $\min\{\gamma\overline{y}_i - \overline{y}_i, 1\}$, contributing an additional $f_i \min\{\gamma\overline{y}_i - \overline{y}_i, 1\}$ to the expected total facility cost. Overall, the contribution of i is $f_i\overline{y}_i + f_i \min\{\gamma\overline{y}_i - \overline{y}_i, 1\} \le \gamma\, f_i\overline{y}_i$, once again. Thus, the expected total facility cost is at most $\gamma \sum_{i\in\mathcal{F}} f_i\overline{y}_i$.

Next, focus on a demand point $k \in \mathcal{D}$. As in the proof of Lemma 3.6, for each center $j \in \mathcal{C}$, $S_j = \mathsf{N}(j) \cap \mathsf{N}(k)$ and $\widehat{\mathcal{C}} = \{j \in \mathcal{C} : S_j \ne \varnothing\}$. Now for each $j \in \widehat{\mathcal{C}}$, the event E_j occurs when a facility in S_j is open in the first phase; the probability of event E_j is then $p_j = \sum_{i\in S_j} \overline{x}_{ik}$. In addition, for each central neighbor $i \in \mathsf{N}(k) \cap \mathcal{L}$, define the event E_i in which a facility is open at location i in the second phase; hence the probability of event E_i is $p_i = \min\{\gamma\overline{y}_i - \overline{y}_i, 1\}$. For each noncentral neighbor $i \in \mathsf{N}(k) \cap \mathcal{R}$, the event E_i occurs when a facility at location i is open, and the probability of event E_i is $p_i = \min\{\gamma\overline{y}_i, 1\}$. As in the proof of Lemma 3.6 and by the assumption at the beginning of the proof, all the events E_ℓ are *independent.* For each $j \in \widehat{\mathcal{C}}$, $\overline{c}_j$ is the expected distance from k to S_j given the event E_j, and considering only the first phase. For each neighbor $i \in \mathsf{N}(k)$, $\overline{c}_i$ is simply the distance c_{ik}. Without loss of generality, we assume that there are d events E_ℓ, and that they are indexed by $\ell \in \{1, \dots, d\}$ with $\overline{c}_1 \le \dots \le \overline{c}_d$. Let D be the event in which none of the events $E_1, \dots, E_d$ occurs, and let $q = \prod_{\ell=1}^{d}(1 - p_\ell)$ be the probability of event D. If B is the backup service cost of k, that is, the distance from k to the facility open during the first phase in the cluster to which k belongs (as in the proof of Lemma 3.10), following the proofs of Lemmas 3.6 and 3.10, the expected service cost of k is at most

$$\overline{c}_1\, p_1 + \overline{c}_2\,(1 - p_1)p_2 + \cdots + \overline{c}_d\,(1 - p_1)\dots(1 - p_{d-1})p_d + \mathsf{E}[B|D]\, q \ . \tag{5.1}$$

It is easy to see that the proof of Lemma 3.11 remains valid, so that $\mathsf{E}[B|D] \le 2g_k + \sum_{i\in\mathcal{F}} c_{ik}\overline{x}_{ik}$. Following the proof of Theorem 3.13, we only need to show that the first d terms of (5.1) are at most $(1-q)\sum_{i\in\mathcal{F}} c_{ik}\overline{x}_{ik}$ and that $q \le 1/e^\gamma$. We will use the following generalization of Lemma 3.9, whose proof is postponed to Section 6. (A much simpler proof of this lemma, based on the Chebyshev Integral Inequality, was observed by Sviridenko [Svi02].)

LEMMA 5.2. *Let $0 \le \overline{c}_1 \le \dots \le \overline{c}_d$, and $z_1, \dots, z_d > 0$, with $\sum_{\ell=1}^{d} z_\ell = 1$, and let $\gamma \ge 0$. Suppose that $Z_1, \dots, Z_d$ are 0-1 independent random variables, with $\mathsf{Prob}\{Z_\ell = 1\} = \min\{\gamma z_\ell, 1\}$; let $\overline{C} = \sum \overline{c}_\ell z_\ell$. Then*

$$\mathsf{E}\left[\min_{Z_\ell=1}{}_{\circ}\, \overline{c}_\ell Z_\ell + \prod_{\ell=1}^{d}(1 - Z_\ell)\overline{C}\right] \le \overline{C} \ .$$

If $j \in \widehat{\mathcal{C}}$, let $z_j = p_j/\gamma$, so that $p_j = \gamma z_j = \min\{\gamma z_j, 1\}$. If $i \in \mathsf{N}(k) \cap \mathcal{R}$, let $z_i = \overline{y}_i = \overline{x}_{ik}$, so that $p_i = \min\{\gamma z_i, 1\}$. Finally, if $i \in \mathsf{N}(k) \cap \mathcal{L}$, with $i \in \mathsf{N}(j)$, $j \in \widehat{\mathcal{C}}$, let $z_i = \overline{x}_{ik} - \overline{x}_{ik}/\gamma$, so that $p_i = \min\{\gamma z_i, 1\}$. Now we have that $\sum_\ell z_\ell = 1$, and $\sum_\ell \overline{c}_\ell z_\ell = \sum_{i\in\mathcal{F}} c_{ik}\overline{x}_{ik}$. As in the proof of Lemma 3.10, using now Lemma 5.2, the first d terms of (5.1) can be bounded by $(1 - q)\sum_{i\in\mathcal{F}} c_{ik}\overline{x}_{ik}$.

To finish the proof of the theorem notice that if p_ℓ is 1 for some ℓ, q is 0; otherwise,

$$q = \prod_{\ell=1}^{d}(1 - p_\ell) = \prod_{\ell=1}^{d}(1 - \gamma z_\ell) \le \prod_{\ell=1}^{d} e^{-\gamma z_\ell} = e^{-\sum_{\ell=1}^{d}\gamma z_\ell} = \frac{1}{e^\gamma}.\Box$$

Using similar arguments to those given in Section 4, the algorithm of the theorem can be derandomized.

For the rest of the section let $\rho \in [0, 1]$ be defined by $\rho\, \mathsf{LP}^* = \sum_{i\in\mathcal{F}} f_i y_i^*$. If ρ is either 0 or 1, there is an optimal solution to P which is integral; that is, all the y_i's are 0 or 1. When $\rho = 0$, the optimal solution sets y_i to 1 exactly for those facilities $i \in \mathcal{F}$ for which $f_i = 0$. The case when $\rho = 1$ is slightly more complicated and requires that the distance function be symmetric and satisfy the triangle inequality. First for each facility location $i \in \mathcal{F}$, define D_i as the set of demand points that are at distance 0 from i. If $i, \ell \in \mathcal{F}$ and $j, k \in \mathcal{D}$, the inequality $c_{ij} \le c_{ik} + c_{\ell k} + c_{\ell j}$ implies that for $i \ne i'$, either $D_i = D_{i'}$ or $D_i \cap D_{i'} = \varnothing$. Hence, we can partition the set of facilities into classes such that if i and i' belong to the same class, $D_i = D_{i'}$, and $D_i \cap D_{i'} = \varnothing$ otherwise. Now since $\rho = 1$, the sets D_i $(i \in \mathcal{F})$ cover all the demand points. Finally, the optimal solution simply opens the cheapest facility in each class (notice that the fact that $D_i \cap D_{i'} = \varnothing$ for i and i' in different classes is crucial to argue optimality). Thus we will assume that $0 < \rho < 1$. For each fixed ρ, we want to apply Theorem 5.1 to a g-close feasible solution to P with a conveniently chosen γ so as to improve the performance guarantee of γ-RANDOMIZED ROUNDING WITH IMPROVED CLUSTERING.

First we consider the optimal solution (x^*, y^*), which is v^*-close. By Theorem 5.1, the expected cost of the solution produced by γ-RANDOMIZED ROUNDING WITH IMPROVED CLUSTERING is at most

$$\gamma \sum_{i\in\mathcal{F}} f_i y_i^* + \sum_{j\in\mathcal{D}}\sum_{i\in\mathcal{F}} c_{ij}x_{ij}^* + \frac{2}{e^\gamma}\sum_{j\in\mathcal{D}} v_j^* \ .$$

Since $\sum_{j\in\mathcal{D}} v_j^* = \mathsf{LP}^*$, and $\sum_{j\in\mathcal{D}}\sum_{i\in\mathcal{F}} c_{ij}x_{ij}^* = (1-\rho)\,\mathsf{LP}^*$, the expected performance guarantee is at most

$$\gamma\,\rho + (1-\rho) + \frac{2}{e^\gamma} \tag{5.2}$$

Now we choose $\gamma \geq 1$ so as to minimize (5.2), which gives

$$\gamma = \begin{cases} \ln(2/\rho) & \text{if } \rho \leq 2/e, \\ 1 & \text{if } \rho > 2/e. \end{cases}$$

In this case, we obtain a performance guarantee of

$$\begin{cases} 1+\rho\,\ln(2/\rho) & \text{if } \rho \leq 2/e, \\ 1+2/e & \text{if } \rho > 2/e. \end{cases}$$

Figure 5.1 shows the performance of the new algorithm for each value of ρ. Notice that the performance improves when ρ gets closer to 0, but then there is no improvement, for instance, when ρ is close to 1. To improve the guarantees for this range of values of ρ, we will use a different g-close solution, the one proposed by Shmoys, Tardos & Aardal [STA97], which provides better guarantees when ρ is close to 1.

In what follows, we apply Theorem 5.1 to the g-close solutions given in [STA97], which were obtained using the filtering technique of Lin & Vitter [LV92b] applied to the optimal solution (x^*, y^*). Fix $\alpha \in (0,1]$. For a demand point $j \in \mathcal{D}$, suppose that $\mathsf{N}(j) = \{1,\ldots,d\}$, with $c_{1j} \leq \ldots \leq c_{dj}$. Let $\ell^* = \min\{\ell : 1 \leq \ell \leq d, \sum_{i=1}^{\ell} x_{ij}^* \geq \alpha\}$; then the *$\alpha$-point* of j, $c_j(\alpha)$, is $c_{\ell^* j}$. For each demand point $j \in \mathcal{D}$, let

$$\beta_j^\alpha = \sum_{i:c_{ij}\leq c_j(\alpha)} x_{ij}^* \ .$$

The filtered solution (x^α, y^α) is defined in a simple way to ensure g-closeness, for $g_j = c_j(\alpha)$ $(j \in \mathcal{D})$, as follows: if $j \in \mathcal{D}$, $i \in \mathcal{F}$,

$$x_{ij}^\alpha = \begin{cases} \dfrac{x_{ij}^*}{\beta_j^\alpha} & \text{if } c_{ij} \leq c_j(\alpha), \\ 0 & \text{otherwise,} \end{cases}$$

and $y_i^\alpha = \min\{1, y_i^*/\alpha\}$ for $i \in \mathcal{F}$. It is easy to verify that (x^α, y^α) is a feasible primal solution (because $\beta_j^\alpha \geq \alpha$) and that (x^α, y^α) is $(c_j(\alpha))$-close. The total fractional facility cost of the filtered solution is $\sum_{i\in F} f_i y_i^\alpha$, and it is clearly bounded by $\sum_{i\in F} f_i y_i^*/\alpha$. Next define $\tau(\alpha) = \sum_{j\in\mathcal{D}} c_j(\alpha)/\sum_{j\in\mathcal{D}}\sum_{i\in\mathcal{F}} c_{ij}x_{ij}^*$. The following lemma follows from Lemma 10 of [STA97] and was observed in [GK99].

LEMMA 5.3. *The function $\tau(\alpha)$ satisfies the following expression*

$$\int_0^1 \tau(\alpha)\ \mathrm{d}\alpha = 1.$$

Proof. It was shown in [STA97] that $\int_0^1 c_j(\alpha)\ \mathrm{d}\alpha = \sum_{i\in\mathcal{F}} c_{ij}x_{ij}^*$, from which the lemma follows at once according to our definitions. ◻

Also note that $\tau(\alpha)$ is a left continuous step function that has at most $O(|\mathcal{D}||\mathcal{F}|)$ break points. Since for each demand point $k \in \mathcal{D}$, the fractional service cost of the filtered solution is bounded by the fractional transportation cost, that is,

$$\sum_{i\in\mathcal{F}} c_{ik}x_{ik}^\alpha = \sum_{\{i:c_{ik}\leq c_k(\alpha)\}} c_{ik}\frac{x_{ik}^*}{\beta_j^\alpha} \leq \sum_{i\in\mathcal{F}} c_{ik}x_{ik}^*,$$

the previous theorem implies that the expected total cost of the solution of γ-RANDOMIZED ROUNDING WITH IMPROVED CLUSTERING when applied to the $(c_j(\alpha))$-close solution (x^α, y^α) is at most

$$\gamma\sum_{i\in F} f_i\frac{y_i^*}{\alpha} + \sum_{j\in\mathcal{D}}\sum_{i\in\mathcal{F}} c_{ij}x_{ij}^* + \frac{2}{e^\gamma}\sum_{j\in D} c_j(\alpha),$$

or an expected performance guarantee of

$$\gamma\frac{\rho}{\alpha} + 1 - \rho + \frac{2}{e^\gamma}(1-\rho)\tau(\alpha). \tag{5.3}$$

To find the best possible guarantee we will use binary search on a target guarantee and Lemma 5.3 in a way similar to that used in [GK99] as follows. Fix $c > 1$ and suppose that γ-RANDOMIZED ROUNDING WITH IMPROVED CLUSTERING does not produce a c-approximation algorithm for every $\alpha \in (0, 1)$, $\gamma \geq 1$. Then

$$\tau(\alpha) > \frac{e^{\gamma}}{2} \frac{(c-1+\rho)\alpha - \gamma\rho}{(1-\rho)\alpha}. \tag{5.4}$$

We are interested only in values of α for which the expression on the right is nonnegative; hence we will assume that $\alpha \geq \gamma\rho/(c-1+\rho)$. For fixed α, the value of $\gamma \geq 1$ that makes the right hand side of (5.4) greatest is given by

$$\gamma(\alpha) = \begin{cases} \dfrac{(c-1+\rho)\alpha - \rho}{\rho} & \text{if } 2\rho/(c-1+\rho) \leq \alpha \leq 1 \\ 1 & \text{if } \rho/(c-1+\rho) \leq \alpha \leq 2\rho/(c-1+\rho). \end{cases}$$

Thus, from (5.4), we obtain a lower bound for $\tau(\alpha)$, $LB_c(\alpha)$, given by

$$LB_c(\alpha) = \begin{cases} \dfrac{\rho}{2} \dfrac{\exp\{[(c-1+\rho)\alpha - \rho]/\rho\}}{(1-\rho)\alpha} & \text{if } 2\rho/(c-1+\rho) \leq \alpha \leq 1 \\ \dfrac{e}{2} \dfrac{(c-1+\rho)\alpha - \rho}{(1-\rho)\alpha} & \text{if } \rho/(c-1+\rho) \leq \alpha \leq 2\rho/(c-1+\rho) \\ 0 & \text{if } 0 \leq \alpha \leq \rho/(c-1+\rho). \end{cases}$$

Note that $LB_c(\alpha)$ is a continuous and increasing function of α (since c and ρ are fixed). Now since $\int_0^1 \tau(\alpha)\ \mathrm{d}\alpha = 1$, and $\tau(\alpha) > LB_c(\alpha)$, if we can show that $\int_0^1 LB_c(\alpha)\ \mathrm{d}\alpha > 1$, we have a contradiction and, thus, the performance guarantee of the algorithm is no greater than c. Using bisection search we can find the smallest c, say $c_\circ$, within a small error tolerance (say 0.0001) for which $\int_0^1 LB_{c_\circ}(\alpha)\ \mathrm{d}\alpha > 1$, and hence show that the algorithm has a performance guarantee $c_\circ$. It is clear that the bisection search takes constant time. We finally address the issue of how to find the parameters α and γ to obtain a $c_\circ$-approximation algorithm. We know that

$$\int_0^1 \tau(\alpha)\ \mathrm{d}\alpha = 1 < \int_0^1 LB_{c_\circ}(\alpha)\ \mathrm{d}\alpha, \tag{5.5}$$

hence there must be a value of α, say $\alpha_\circ$, such that

$$\tau(\alpha_\circ) \leq LB_{c_\circ}(\alpha_\circ). \tag{5.6}$$

This implies that when $\gamma = \gamma(\alpha_\circ)$, γ-RANDOMIZED ROUNDING WITH IMPROVED CLUSTERING produces a solution with expected cost within a factor of $c_\circ$ of optimum. Thus we only need to argue how to find such an $\alpha_\circ$ in polynomial time.

Now, we know that $\tau(\alpha)$ is a left continuous step function with at most $O(|\mathcal{D}|^2)$ break points. Hence we can find a partition of the interval $(0, 1] = (0, s_1] \cup (s_1, s_2] \cup \ldots \cup (s_k, 1]$, with $k = O(|\mathcal{D}|^2)$, such that $\tau(\alpha)$ is constant in each subinterval. Suppose that $\tau(\alpha) = z$ in the subinterval $(s_i, s_{i+1}]$. If $z > LB_{c_\circ}(s_{i+1})$, since $LB_{c_\circ}$ is increasing, the inequality must hold for the whole subinterval. Thus if for all the right end-points of the subintervals inequality (5.6) does not hold, $\tau(\alpha) > LB_{c_\circ}(\alpha)$ for each $\alpha \in (0, 1]$, contradicting (5.5). This gives a simple way to find a $c_\circ$-approximation algorithm. Figure 5.1 shows the performance guarantees obtained using this algorithm.

We conclude the section by pointing out that there is a minor improvement on the best guarantee we can achieve using the algorithms γ-RANDOMIZED ROUNDING WITH IMPROVED CLUSTERING. Indeed a careful computation gives a slightly improved overall performance guarantee of $1.73352 < 1.73576 \approx 1 + 2/e$.

6. Proof of Lemmas 3.9 and 5.2. Notice that Lemma 3.9 follows from Lemma 5.2 by simply taking $\gamma = 1$. Thus we only need to prove Lemma 5.2. As noted above, Sviridenko [Svi02] (in the proof and discussion around his Lemmas 4 and 5) observed that a much simpler version of this proof can be derived from the Chebyshev Integral Inequality, and cited particular variants in the text of Hardy, Littlewood, and Polya [HLP52].

The key idea of the proof is to observe that the expected value decreases as a function of γ; thus, since when γ is 0 the expected value is exactly $\overline{C}$ we are done.

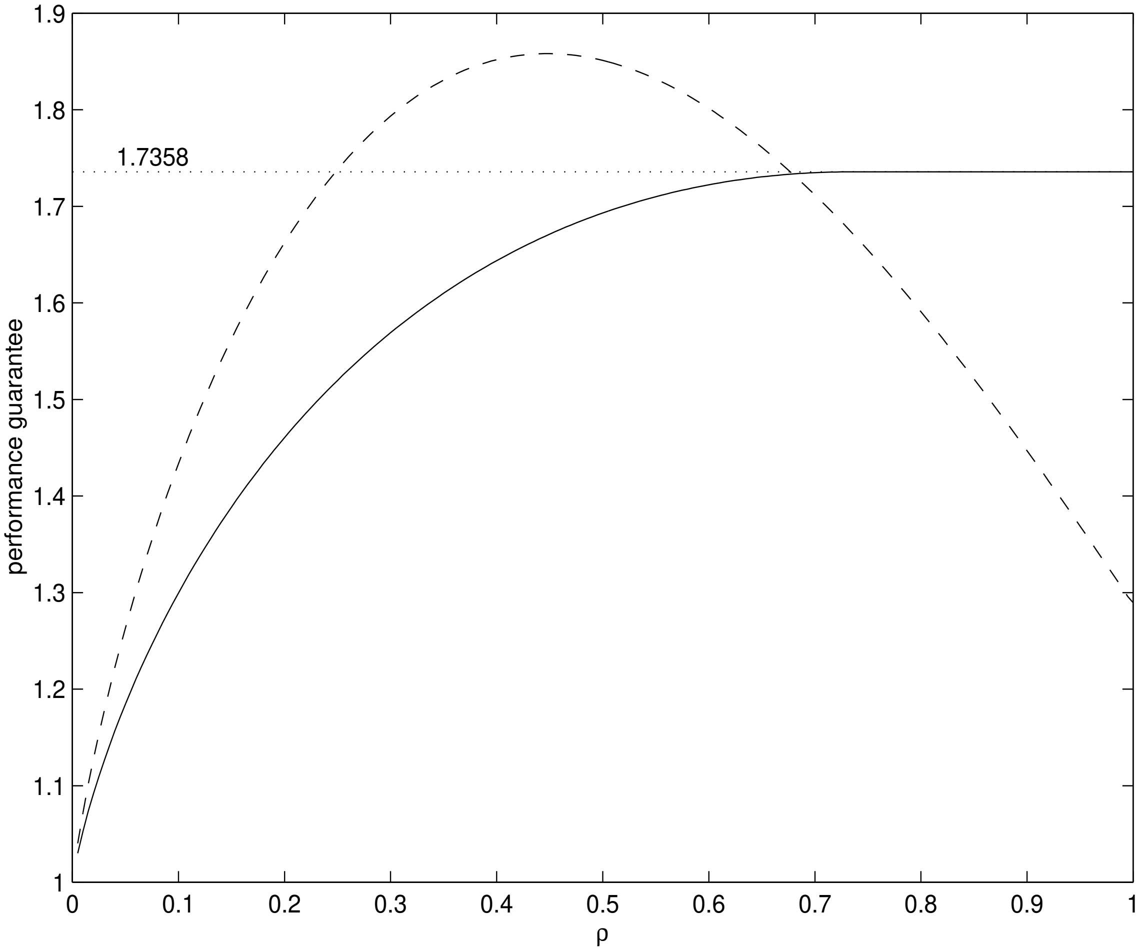

FIG. 5.1. *Performance guarantees of* γ-RANDOMIZED ROUNDING WITH IMPROVED CLUSTERING *as a function of* ρ. *The solid line corresponds to the algorithm that uses the optimal* v^**-close solution* (x^*, y^*)*, while the dashed line corresponds to the algorithm run with the filtered solution of [STA97].*

For $\gamma \geq 0$, let $p(\gamma) := \mathsf{E}\left[\min_{\circ\{i:Z_i=1\}} \overline{c}_i Z_i + \prod_{i=1}^{d}(1-Z_i)\overline{C}\right]$. Let $\overline{\gamma} = \min_i(1/z_i)$. We first prove the lemma for $\gamma \leq \overline{\gamma}$, so that $\gamma z_i < 1$, if $\gamma < \overline{\gamma}$. In this case, $p(\gamma)$ can be computed as

$$\overline{c}_1\gamma z_1 + \overline{c}_2\gamma z_2(1-\gamma z_1) + \cdots + \overline{c}_d\gamma z_d(1-\gamma z_1)\ldots(1-\gamma z_{d-1}) + \overline{C}\prod_{\ell=1}^{d}(1-\gamma z_\ell).$$

Notice that $p(\gamma)$ is a polynomial on γ, and that $p(0) = \overline{C}$. Hence, to prove the lemma in this case it is enough to show that $p'(\gamma) \leq 0$, for $\gamma \in (0, \overline{\gamma})$.

Next define for each $\ell = 0, 1, \ldots, d$

$$F_\ell(\gamma) = \begin{cases} 1 & \text{if } \ell = 0, \\ (1-\gamma z_\ell)F_{\ell-1}(\gamma) & \text{if } \ell \geq 1. \end{cases}$$

Thus we can write

$$\begin{aligned} p(\gamma) &= \overline{c}_1\gamma z_1 F_0(\gamma) + \overline{c}_2\gamma z_2 F_1(\gamma) + \cdots + \overline{c}_d\gamma z_d F_{d-1}(\gamma) + \overline{C}F_d(\gamma) \\ &= \sum_{\ell=1}^{d} \overline{c}_\ell \gamma z_\ell F_{\ell-1}(\gamma) + F_d(\gamma)\sum_{\ell=1}^{d}\overline{c}_\ell z_\ell \\ &= \sum_{\ell=1}^{d} \overline{c}_\ell z_\ell \left[\gamma F_{\ell-1}(\gamma) + F_d(\gamma)\right] \quad . \end{aligned}$$

Then

$$p'(\gamma) = \sum_{\ell=1}^{d} \overline{c}_\ell z_\ell \left[F_{\ell-1}(\gamma) + \gamma F'_{\ell-1}(\gamma) + F'_d(\gamma)\right].$$

For $\ell = 1, \ldots, d$, let $\lambda_\ell = F_{\ell-1}(\gamma) + \gamma F'_{\ell-1}(\gamma) + F'_d(\gamma)$, so that $p'(\gamma) = \sum_{\ell=1}^{d} \overline{c}_\ell z_\ell \lambda_\ell$. Next note that

$$F'_\ell(\gamma) = \begin{cases} 0 & \text{if } \ell = 0 \\ -z_\ell F_{\ell-1}(\gamma) + (1-\gamma z_\ell)F'_{\ell-1}(\gamma) & \text{if } \ell \geq 1. \end{cases}$$

It is easy to check by induction that $F'_\ell(\gamma) \leq 0$, $\ell = 0, \ldots, d$. To prove the lemma in this case, we will use the following two claims.

Claim 1. *The following equality holds:* $\sum_{\ell=1}^{d} z_\ell \lambda_\ell = 0$.

Claim 2. *There is an index $\ell_\circ$, $0 \leq \ell_\circ \leq d$, such that $\lambda_1, \ldots, \lambda_{\ell_\circ - 1} \geq 0$, and $\lambda_{\ell_\circ}, \ldots, \lambda_d \leq 0$.*

To prove that $p'(\gamma) \leq 0$, take $\ell_\circ$ as in Claim 2, then use Claim 1 and the order of the $\overline{c}_\ell$'s:

$$\begin{aligned} p'(\gamma) &= \sum_{\ell=1}^{\ell_\circ-1} \overline{c}_\ell \lambda_\ell z_\ell + \sum_{\ell=\ell_\circ}^{d} \overline{c}_\ell \lambda_\ell z_\ell \\ &\leq \overline{c}_{\ell_\circ} \left(\sum_{\ell=1}^{\ell_\circ-1} \lambda_\ell z_\ell\right) + \overline{c}_{\ell_\circ} \left(\sum_{\ell=\ell_\circ}^{d} \lambda_\ell z_\ell\right) \\ &= \overline{c}_{\ell_\circ} \sum_{\ell=1}^{d} \lambda_\ell z_\ell \\ &= 0 \ . \end{aligned}$$

We complete the proof of the lemma for $\gamma \in [0, \overline{\gamma}]$ by proving the claims.

Proof of Claim 1. Note first that for $\ell = 1, \ldots, d$

$$-z_\ell F_{\ell-1}(\gamma) - \gamma z_\ell F'_{\ell-1}(\gamma) = F'_\ell(\gamma) - F'_{\ell-1}(\gamma).$$

Thus

$$\sum_{\ell=1}^{d} [-z_\ell F_{\ell-1}(\gamma) - \gamma z_\ell F'_{\ell-1}(\gamma)] = F'_d(\gamma) - F'_0(\gamma) = F'_d(\gamma),$$

which implies that $\sum_\ell z_\ell \lambda_\ell = 0$, since $\sum_\ell z_\ell = 1$. □

Proof of Claim 2. If $\lambda_\ell \geq 0$ (or $\lambda_\ell \leq 0$) for all ℓ, using Claim 1, $\lambda_\ell = 0$ for all ℓ, and any index works. Hence we can assume that at least one $\lambda_\ell < 0$ and at least one $\lambda_\ell > 0$. Suppose the claim does not hold. Then there must exist an index ℓ, $1 \leq \ell \leq d$, such that $\lambda_\ell \leq 0$, but $\lambda_{\ell+1} > 0$. In particular,

$$\lambda_{\ell+1} = F_\ell(\gamma) + \gamma F'_\ell(\gamma) + F'_d(\gamma) > 0.$$

Thus,

$$\begin{aligned} \lambda_{\ell+1} - F'_d(\gamma) &= F_\ell(\gamma) + \gamma F'_\ell(\gamma) \\ &= (1-\gamma z_\ell)F_{\ell-1}(\gamma) + (1-\gamma z_\ell)\gamma F'_{\ell-1}(\gamma) - \gamma z_\ell F_{\ell-1}(\gamma) \\ &= (1-\gamma z_\ell)[F_{\ell-1}(\gamma) + \gamma F'_{\ell-1}(\gamma)] - \gamma z_\ell F_{\ell-1}(\gamma) \\ &> -F'_d(\gamma) \\ &\geq 0. \end{aligned}$$

Let $t := F_{\ell-1}(\gamma) + \gamma F'_{\ell-1}(\gamma)$. Since $(1-\gamma z_\ell) > 0$ and $-\gamma z_\ell F_{\ell-1}(\gamma) < 0$, it must be the case that $t > 0$. Since $\lambda_\ell \leq 0$, $t \leq -F'_d(\gamma)$. Thus,

$$0 < t \leq -F'_d(\gamma) < \lambda_{\ell+1} - F'_d(\gamma) = (1-\gamma z_\ell)t - \gamma z_\ell F_{\ell-1}(\gamma) < (1-\gamma z_\ell)t \ ,$$

which is impossible since $(1-\gamma z_\ell) < 1$. □

Next suppose that $\gamma \geq \overline{\gamma}$, and let $\ell_\circ$ be the smallest index for which $\min\{\gamma z_\ell, 1\}$ is 1. Now we have that

$$p(\gamma) = \overline{c}_1 \gamma z_1 + \overline{c}_2 \gamma z_2 (1-\gamma z_1) + \cdots + \overline{c}_{\ell_\circ}(1-\gamma z_1)\ldots(1-\gamma z_{\ell_\circ-1}).$$

We reduce this case to the previous one as follows. Let $\alpha = \sum_{\ell=1}^{\ell_\circ - 1} z_\ell$, $\gamma' = \gamma\alpha$, and let $z'_\ell = z_\ell/\alpha$ for $\ell = 1, \ldots, \ell_\circ - 1$; note that $\sum_{\ell=1}^{\ell_\circ - 1} z'_\ell = 1$. We have then that

$$p(\gamma) = \overline{c}_1 \gamma' z'_1 + \overline{c}_2 \gamma' z'_2 (1 - \gamma' z'_1) + \cdots + \overline{c}_{\ell_\circ} (1 - \gamma' z'_1) \ldots (1 - \gamma' z'_{\ell_\circ - 1}) \ . \tag{6.1}$$

Notice that $\gamma' z'_\ell < 1$ for each $\ell = 1, \ldots, \ell_\circ - 1$. If $\overline{C}' = \sum_{\ell=1}^{\ell_\circ - 1} \overline{c}_\ell z'_\ell$, and $q = \prod_{\ell=1}^{\ell_\circ - 1} (1 - \gamma' z'_\ell)$, we can apply the previous case to conclude that the first $\ell_\circ - 1$ terms of (6.1) can be bounded by $(1 - q)\overline{C}'$. Thus,

$$p(\gamma) \le (1 - q)\overline{C}' + \overline{c}_{\ell_\circ} q \ . \tag{6.2}$$

Suppose that $q \le 1 - \alpha$. Then, since $\overline{C}' \le \overline{c}_{\ell_\circ}$ by the ordering of the $\overline{c}$'s, from (6.2)

$$p(\gamma) \le \alpha \overline{C}' + (1 - \alpha)\overline{c}_{\ell_\circ} = \sum_{\ell=1}^{\ell_\circ - 1} \overline{c}_\ell z_\ell + (1 - \alpha)\overline{c}_{\ell_\circ} \le \sum_{\ell=1}^{d} \overline{c}_\ell z_\ell = \overline{C} \ ,$$

and the lemma follows. Hence to conclude the proof, we only need to argue that $q \le 1 - \alpha$. First note that since $1 - x \le e^{-x}$ $(x \ge 0)$,

$$q = \prod_{\ell=1}^{\ell_\circ - 1} (1 - \gamma' z'_\ell) \le \prod_{\ell=1}^{\ell_\circ - 1} e^{-\gamma' z'_\ell} = e^{-\gamma' \sum_{\ell=1}^{\ell_\circ - 1} z'_\ell} = e^{-\gamma'} = e^{-\gamma\alpha}.$$

Now since $\gamma z_{\ell_\circ} \ge 1$, it must be that $\gamma \ge 1/(1 - \alpha)$. Finally

$$q \le e^{-\gamma\alpha} \le e^{-\alpha/(1-\alpha)} \le 1 - \alpha,$$

where the last inequality follows from $1 - \ln(y) \le 1/y$ for $y \in (0, 1)$ by setting $y = 1 - \alpha$.

REFERENCES

[Aro98] S. Arora. Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems. *J. Assoc. Comput. Mach 45*, 753–782, 1998.

[ARR98] S. Arora, P. Raghavan, and S. Rao. Approximation schemes for Euclidean k-medians and related problems. In *Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, pages 106–113, 1998.

[AS97] A.A. Ageev and M.I. Sviridenko. An approximation algorithm for the uncapacitated facility location problem. Manuscript, 1997.

[Bal65] M.L. Balinski. Integer programming: methods, uses, computation. *Management Science 12*, 253–313, 1965.

[CFN77] 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 Science 23*, 789–810, 1977.

[CG99] M. Charikar and S. Guha. Improved combinatorial algorithms for the facility location and k-median problems. In *Proceedings of the 40th Annual IEEE Symposium on Foundations of Computer Science*, pages 378–388, 1999.

[Chu98] F.A. Chudak. Improved approximation algorithms for uncapacitated facility location. In R.E. Bixby, E.A. Boyd, R.Z. Rios-Mercado (eds.). *Integer Programming and Combinatorial Optimization, 6th International IPCO Conference, Lecture Notes in Computer Science 1412*, pages 180–194. Springer, Berlin, 1998.

[CNW90] 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.

[ES73] P. Erdös and J.L. Selfridge. On a combinatorial game. *Journal of Combinatorial Theory, Series A 14*, 298–301, 1973.

[Fei98] U. Feige. A threshold of $\ln n$ for approximating set cover. *J. Assoc. Comput. Mach. 45*, 634–652, 1998.

[GK99] S. Guha and S. Khuller. Greedy strikes back: improved facility location algorithms. *J. Algorithms 31*, 228–248, 1999.

[GW94] M.X. Goemans and D.P. Williamson. New 3/4-approximation algorithms for max-sat. *SIAM Journal on Discrete Mathematics 7*, 656–666, 1994.

[HLP52] G.H. Hardy, J.E. Littlewood, and G. Pólya. *Inequalities.* Cambridge University Press, Cambridge, 1952.

[Hoc82] D.S. Hochbaum. Heuristics for the fixed cost median problem. *Math. Programming 22*, 148–162, 1982.

[JMS02] K. Jain, M. Mahdian, and A. Saberi. A new greedy approach for facility location problems. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, pages 731–740, 2002.

[JV01] K. Jain and V.V. Vazirani. Approximation algorithms for metric facility location and k-median problems using primal-dual schema and Lagrangian relaxation. *J. Assoc. Comput. Mach. 48*, 274–296, 2001.

[KR99] S.G. Kolliopoulos and Satish Rao. A nearly linear-time approximation scheme for the Euclidean κ-median problem. In J. Nesetril (ed.). *Algorithms – ESA '99, 7th Annual European Symposium, Lecture Notes in Computer Science 1643*, pages 378–389. Springer, Berlin, 1999.

[KPR00] M.R. Korupolu, C.G. Plaxton, and R. Rajaraman. Analysis of a local search heuristic for facility location problems. *J. Algorithms 37*, 146–188, 2000.

[LV92a] J.H. Lin and J.S. Vitter. Approximation algorithms for geometric median problems. *Information Processing Letters 44*, 245–249, 1992.

[LV92b] J.H. Lin and J.S. Vitter. ϵ-approximation with minimum packing constraint violation. In *Proceedings of the 24th Annual ACM Symposium on Theory of Computing*, pages 771–782, 1992.

[MMSV01] M. Mahdian, E. Markakis, A. Saberi, and V. Vazirani. A greedy facility location algorithm analyzed using dual fitting. In M.X. Goemans, K. Jansen, J.D.P. Rolim, and L. Trevisan (eds.). *Approximation, Randomization, and Combinatorial Optimization: Algorithms and Techniques, 4th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, APPROX 2001, and 5th International Workshop on Randomization and Approximation Techniques in Computer Science, RANDOM 2001, Lecture Notes in Computer Science 2129*, pages 127–137. Springer, Berlin, 2001.

[MYZ02] M. Mahdian, Y. Ye, and J. Zhang. Improved approximation algorithms for metric facility location problems. In K. Jansen, S. Leonardi, V. Vazirani (eds.). *Approximation Algorithms for Combinatorial Optimization, 4th International Workshop, APPROX 2002, Lecture Notes in Computer Science 2462*, pages 229–242. Springer, Berlin, 2002.

[MF90] P. Mirchandani and R. Francis, eds. *Discrete Location Theory*. John Wiley and Sons, Inc., New York, 1990.

[MP00] R.R. Mettu and C.G. Plaxton. The online median problem. In *Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science*, pages 339–348, 2000.

[Rag88] P. Raghavan. Probabilistic construction of deterministic algorithms: approximating packing integer programs. *Journal of Computer and System Sciences 37*, 130–143, 1988.

[RT87] P. Raghavan and C.D. Thompson. Randomized rounding. *Combinatorica 7*, 365–374, 1987.

[Spe87] J. Spencer. *Ten Lectures on the Probabilistic Method*. SIAM, Philadelphia, 1987.

[STA97] D.B. Shmoys, É. Tardos, and K. Aardal. Approximation algorithms for facility location problems. In *Proceedings of the 29th ACM Symposium on Theory of Computing*, pages 265–274, 1997.

[Svi97] M.I. Sviridenko, July, 1997. Personal communication.

[Svi98] M.I. Sviridenko, July, 1998. Personal communication.

[Svi02] M. Sviridenko. An improved approximation algorithm for the metric uncapacitated facility location problem. In W. Cook and A.S. Schulz (eds.). *Integer Programming and Combinatorial Optimization, 9th International IPCO Converence, Lecture Notes in Computer Science 2337*, pages 240–257. Springer, Berlin, 2002.