Fault-Tolerant Facility Location: a randomized dependent LP-rounding algorithm*

Jaroslaw Byrka[1**], Aravind Srinivasan[2], and Chaitanya Swamy[3]

[1] Institute of Mathematics, Ecole Polytechnique Federale de Lausanne, CH-1015 Lausanne, SWITZERLAND. jaroslaw.byrka@epfl.ch
[2] Dept. of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742, USA. srin@cs.umd.edu
[3] Dept. of Combinatorics & Optimization, Faculty of Mathematics, University of Waterloo, Waterloo, ON N2L 3G1, CANADA. cswamy@math.uwaterloo.ca

Abstract. We give a new randomized LP-rounding 1.725-approximation algorithm for the metric Fault-Tolerant Uncapacitated Facility Location problem. This improves on the previously best known 2.076-approximation algorithm of Swamy & Shmoys. To the best of our knowledge, our work provides the first application of a dependent-rounding technique in the domain of facility location. The analysis of our algorithm benefits from, and extends, methods developed for Uncapacitated Facility Location; it also helps uncover new properties of the dependent-rounding approach. An important concept that we develop is a novel, hierarchical clustering scheme. Typically, LP-rounding approximation algorithms for facility location problems are based on partitioning facilities into disjoint clusters and opening at least one facility in each cluster. We extend this approach and construct a laminar family of clusters, which then guides the rounding procedure: this allows us to exploit properties of dependent rounding, and provides a quite tight analysis resulting in the improved approximation ratio.

1 Introduction

In Facility Location problems we are given a set of clients $\mathcal{C}$ that require a certain service. To provide such a service, we need to open a subset of a given set of facilities $\mathcal{F}$. Opening each facility $i \in \mathcal{F}$ costs f_i and serving a client j by facility i costs c_{ij}; the standard assumption is that the c_{ij} are symmetric and constitute a metric. (The non-metric case is much harder to approximate.) In this paper, we follow Swamy & Shmoys [11] and study the Fault-Tolerant Facility Location

* This work was partially supported by: (i) the Future and Emerging Technologies Unit of EC (IST priority - 6th FP), under contract no. FP6-021235-2 (project ARRIVAL), (ii) MNISW grant number N N206 1723 33, 2007-2010, (iii) NSF ITR Award CNS-0426683 and NSF Award CNS-0626636, and (iv) NSERC grant 327620-09 and an Ontario Early Researcher Award.

** Work of this author was partially conducted at CWI Amsterdam, TU Eindhoven, and while visiting the University of Maryland.

(FTFL) problem, where each client has a positive integer specified as its *coverage requirement* r_j. The task is to find a minimum-cost solution which opens some facilities from $\mathcal{F}$ and connects each client j to r_j different open facilities.

The FTFL problem was introduced by Jain & Vazirani [7]. Guha et al. [6] gave the first constant factor approximation algorithm with approximation ratio 2.408. This was later improved by Swamy & Shmoys [11] who gave a 2.076-approximation algorithm. FTFL generalizes the standard Uncapacitated Facility Location (UFL) problem wherein $r_j = 1$ for all j, for which Guha & Khuller [5] proved an approximation lower bound of ≈ 1.463. The current-best approximation ratio for UFL is achieved by the 1.5-approximation algorithm of Byrka [2].

In this paper we give a new LP-rounding 1.7245-approximation algorithm for the FTFL problem. It is the first application of the dependent rounding technique of [10] to a facility location problem.

Our algorithm uses a novel clustering method, which allows clusters not to be disjoint, but rather to form a laminar family of subsets of facilities. The hierarchical structure of the obtained clustering exploits properties of dependent rounding. By first rounding the "facility-opening" variables within smaller clusters, we are able to ensure that at least a certain number of facilities is open in each of the clusters. Intuitively, by allowing clusters to have different sizes we may, in a more efficient manner, guarantee the opening of sufficiently-many facilities around clients with different coverage requirements r_j. In addition, one of our main technical contributions is Theorem 2, which develops a new property of the dependent-rounding technique that appears likely to have further applications. Basically, suppose we apply dependent rounding to a sequence of reals and consider an arbitrary subset S of the rounded variables (each of which lies in $\{0, 1\}$) as well as an arbitrary integer $k > 0$. Then, a natural fault-tolerance-related objective is that if X denotes the number of variables rounded to 1 in S, then the random variable $Z = \min\{k, X\}$ be "large". (In other words, we want X to be "large", but X being more than k does not add any marginal utility.) We prove that if X_0 denotes the corresponding sum wherein the reals are rounded *independently* and if $Z_0 = \min\{k, X_0\}$, then $\mathbf{E}[Z] \geq \mathbf{E}[Z_0]$. Thus, for analysis purposes, we can work with Z_0, which is much more tractable due to the independence; at the same time, we derive all the benefits of dependent rounding (such as a given number of facilities becoming available in a cluster, with probability one). Given the growing number of applications of dependent-rounding methodologies, we view this as a useful addition to the toolkit.

2 Dependent rounding

Given a fractional vector $y = (y_1, y_2, \dots, y_N) \in [0, 1]^N$ we often seek to round it to an integral vector $\hat{y} \in \{0, 1\}^N$ that is in a problem-specific sense very "close to" y. The dependent-randomized-rounding technique of [10] is one such approach known for preserving the sum of the entries deterministically, along with concentration bounds for any linear combination of the entries; we will generalize a known property of this technique in order to apply it to the FTFL problem. The very useful *pipage rounding* technique of [1] was developed prior to [10], and can be viewed as a derandomization (deterministic analog) of [10] via the method of conditional probabilities. Indeed, the results of [1] were applied in the work of [11]; the probabilistic intuition, as well as our generalization of the analysis of [10], help obtain our results.

Define $[t] = \{1, 2, \ldots, t\}$. Given a fractional vector $y = (y_1, y_2, \ldots, y_N) \in [0,1]^N$, the rounding technique of [10] (henceforth just referred to as "dependent rounding") is a polynomial-time randomized algorithm to produce a random vector $\hat{y} \in \{0,1\}^N$ with the following three properties:

(P1): marginals. $\forall i,\ \Pr[\hat{y}_i = 1] = y_i$;
(P2): sum-preservation. With probability one, $\sum_{i=1}^N \hat{y}_i$ equals either $\lfloor \sum_{i=1}^N y_i \rfloor$ or $\lceil \sum_{i=1}^N y_i \rceil$; and
(P3): negative correlation. $\forall S \subseteq [N]$, $\Pr[\bigwedge_{i \in S}(\hat{y}_i = 0)] \leq \prod_{i \in S}(1 - y_i)$, and $\Pr[\bigwedge_{i \in S}(\hat{y}_i = 1)] \leq \prod_{i \in S} y_i$.

In this paper, we also exploit the order in which the entries of the given fractional vector y are rounded. We initially define a laminar family of subsets of indices $\mathcal{S} \subseteq 2^{[N]}$. When applying the dependent rounding procedure, we first round within the smaller sets, until at most one fractional entry in a set is left, then we proceed with bigger sets possibly containing the already rounded entries. It can easily be shown that it assures the following version of property (P2) for all subsets S from the laminar family $\mathcal{S}$:

(P2'): sum-preservation. With probability one, $\sum_{i \in S} \hat{y}_i = \sum_{i \in S} y_i$ and $|\{i \in S : \hat{y}_i = 1\}| = \lfloor \sum_{i \in S} y_i \rfloor$.

Now, let $S \subseteq [N]$ be any subset, not necessarily from $\mathcal{S}$. In order to present our results, we need two functions, Sum_S and $g_{\lambda,S}$. For any vector $x \in [0,1]^n$, let $\mathrm{Sum}_S(x) = \sum_{i \in S} x_i$ be the sum of the elements of x indexed by elements of S; in particular, if x is a (possibly random) vector with all entries either 0 or 1, then $\mathrm{Sum}_S(x)$ counts the number of entries in S that are 1. Next, given $s = |S|$ and a real vector $\lambda = (\lambda_0, \lambda_1, \lambda_2, \ldots, \lambda_s)$, we define, for any $x \in \{0,1\}^n$,

$$g_{\lambda,S}(x) = \sum_{i=0}^{s} \lambda_i \cdot \mathcal{I}(\mathrm{Sum}_S(x) = i),$$

where $\mathcal{I}(\cdot)$ denotes the indicator function. Thus, $g_{\lambda,S}(x) = \lambda_i$ if $\mathrm{Sum}_S(x) = i$.

Let $\mathcal{R}(y)$ be a random vector in $\{0,1\}^N$ obtained by *independently* rounding each y_i to 1 with probability y_i, and to 0 with the complementary probability of $1 - y_i$. Suppose, as above, that $\hat{y}$ is a random vector in $\{0,1\}^N$ obtained by applying the dependent rounding technique to y. We start with a general theorem and then specialize it to Theorem 2 that will be very useful for us:

Theorem 1. *Suppose we conduct dependent rounding on $y = (y_1, y_2, \ldots, y_N)$. Let $S \subseteq [N]$ be any subset with cardinality $s \geq 2$, and let $\lambda = (\lambda_0, \lambda_1, \lambda_2, \ldots, \lambda_s)$ be any vector, such that for all r with $0 \leq r \leq s-2$ we have $\lambda_r - 2\lambda_{r+1} + \lambda_{r+2} \leq 0$. Then, $\mathbf{E}[g_{\lambda,S}(\hat{y})] \geq \mathbf{E}[g_{\lambda,S}(\mathcal{R}(y))]$.*

Theorem 2. *For any $y \in [0,1]^N$, $S \subseteq [N]$, and $k = 1, 2, \ldots$, we have*

$$\mathbf{E}[\min\{k, Sum_S(\hat{y})\}] \geq \mathbf{E}[\min\{k, Sum_S(\mathcal{R}(y))\}].$$

Using the notation $\exp(t) = e^t$, our next key result is:

Theorem 3. *For any $y \in [0,1]^N$, $S \subseteq [N]$, and $k = 1, 2, \ldots$, we have*

$$\mathbf{E}[\min\{k, Sum_S(\mathcal{R}(y))\}] \geq k \cdot (1 - \exp(-Sum_S(y)/k)).$$

The above two theorems yield a key corollary that we will use:

Corollary 1.

$$\mathbf{E}[\min\{k, Sum_S(\hat{y})\}] \geq k \cdot (1 - \exp(-Sum_S(y)/k)).$$

Proofs will appear in the full version of the paper (see also [3]).

3 Algorithm

3.1 LP-relaxation

The FTFL problem is defined by the following Integer Program (IP).

$$\begin{array}{llll}
\text{minimize } \sum_{i\in\mathcal{F}} f_i y_i + \sum_{j\in\mathcal{C}} \sum_{i\in\mathcal{F}} c_{ij} x_{ij} & & & (1) \\
\text{subject to:} \quad \sum_i x_{ij} \geq r_j & & \forall j \in \mathcal{C} & (2) \\
\qquad x_{ij} \leq y_i & & \forall j \in \mathcal{C}\ \forall i \in \mathcal{F} & (3) \\
\qquad y_i \leq 1 & & \forall i \in \mathcal{F} & (4) \\
\qquad x_{ij}, y_i \in Z_{\geq 0} & & \forall j \in \mathcal{C}\ \forall i \in \mathcal{F}, & (5)
\end{array}$$

where $\mathcal{C}$ is the set of clients, $\mathcal{F}$ is the set of possible locations of facilities, f_i is a cost of opening a facility at location i, c_{ij} is a cost of serving client j from a facility at location i, and r_j is the amount of facilities client j needs to be connected to.

If we relax constraint (5) to $x_{ij}, y_i \geq 0$ we obtain the standard LP-relaxation of the problem. Let (x^*, y^*) be an optimal solution to this LP relaxation. We will give an algorithm that rounds this solution to an integral solution $(\tilde{x}, \tilde{y})$ with cost at most $\gamma \approx 1.7245$ times the cost of (x^*, y^*).

3.2 Scaling

We may assume, without loss of generality, that for any client $j \in \mathcal{C}$ there exists at most one facility $i \in \mathcal{F}$ such that $0 < x_{ij} < y_i$. Moreover, this facility can be assumed to have the highest distance to client j among the facilities that fractionally serve j in (x^*, y^*).

We first set $\tilde{x}_{ij} = \tilde{y}_i = 0$ for all $i \in \mathcal{F}$, $j \in \mathcal{C}$. Then we scale up the fractional solution by the constant $\gamma \approx 1.7245$ to obtain a fractional solution $(\hat{x}, \hat{y})$. To be precise: we set $\hat{x}_{ij} = \min\{1, \gamma \cdot x^*_{ij}\}$, $\hat{y}_i = \min\{1, \gamma \cdot y^*_i\}$. We open each facility i with $\hat{y}_i = 1$ and connect each client-facility pair with $\hat{x}_{ij} = 1$. To be more precise, we modify $\hat{y}$, $\tilde{y}$, $\hat{x}$, $\tilde{x}$ and service requirements r as follows. For each facility i with $\hat{y}_i = 1$, set $\hat{y}_i = 0$ and $\tilde{y}_i = 1$. Then, for every pair (i, j) such that $\hat{x}_{ij} = 1$, set $\hat{x}_{ij} = 0$, $\tilde{x}_{ij} = 1$ and decrease r_j by one. When this process is finished we call the resulting r, $\hat{y}$ and $\hat{x}$ by $\overline{r}$, $\overline{y}$ and $\overline{x}$. Note that the connections that we made in this phase can be paid for by a difference in the connection cost between $\hat{x}$ and $\overline{x}$. We will show that the remaining connection cost of the solution of the algorithm is expected to be at most the cost of $\overline{x}$.

For the feasibility of the final solution, it is essential that if we connected client j to facility i in this initial phase, we will not connect it again to i in the rest of the algorithm. There will be two ways of connecting clients in the process of rounding $\overline{x}$. The first one connects client j to a subset of facilities serving j in $\overline{x}$. Recall that if j was connected to facility i in the initial phase, then $\overline{x}_{ij} = 0$, and no additional i-j connection will be created.

The connections of the second type will be created in a process of *clustering*. The clustering that we will use is a generalization of the one of Chudak & Shmoys for the UFL problem [4]. As a result of this clustering process, client j will be allowed to connect itself via a different client j' to a facility open around j'. j' will be called a *cluster center* for a subset of facilities, and it will make sure that at least some guaranteed number of these facilities will get opened.

To be certain that client j does not get again connected to facility i with a path via client j', facility i will never be a member of the set of facilities clustered by client j'. We call a facility i *special* for client j iff $\tilde{y}_i = 1$ and $0 < \overline{x}_{ij} < 1$. Note that, by our earlier assumption, there is at most one special facility for each client j, and that a special facility must be at maximal distance among facilities serving j in $\overline{x}$. When rounding the fractional solution in Section 3.5, we take care that special facilities are not members of the formed clusters.

3.3 Close and distant facilities

Before we describe how do we cluster facilities, we specify the facilities that are interesting for a particular client in the clustering process. The following can be fought of as a version of a *filtering* technique of Lin and Vitter [8], first applied to facility location by Shmoys et al. [9]. The analysis that we use here is a version of the argument of Byrka [2].

As a result of the scaling that was described in the previous section, the connection variables $\overline{x}$ amount for a total connectivity that exceeds the requirement $\overline{r}$. More precisely, we have $\sum_{i \in \mathcal{F}} \overline{x}_{ij} \geq \gamma \cdot \overline{r}_j$ for every client $j \in \mathcal{C}$. We will consider for each client j a subset of facilities that are just enough to provide it a fractional connection of $\overline{r}_j$. Such a subset is called a set of *close facilities* of client j and is defined as follows.

For every client j consider the following construction. Let $i_1, i_2, \dots, i_{|\mathcal{F}|}$ be the ordering of facilities in $\mathcal{F}$ in a nondecreasing order of distances c_{ij} to client j. Let i_k be the facility in this ordering, such that $\sum_{l=1}^{k-1} \overline{x}_{i_l j} < \overline{r}_j$ and $\sum_{l=1}^{k} \overline{x}_{i_l j} \geq \overline{r}_j$.

Define

$$\overline{x}^{(c)}_{i_l j} = \begin{cases} \overline{x}_{i_l j} & \text{for } l < k, \\ \overline{r}_j - \sum_{l=1}^{k-1} \overline{x}_{i_l j} & \text{for } l = k, \\ 0 & \text{for } l > k \end{cases}$$

Define $\overline{x}^{(d)}_{ij} = \overline{x}_{ij} - \overline{x}^{(c)}_{ij}$ for all $i \in \mathcal{F}, j \in \mathcal{C}$.

We will call the set of facilities $i \in \mathcal{F}$ such that $\overline{x}^{(c)}_{ij} > 0$ the set of *close facilities* of client j and we denote it by C_j. By analogy, we will call the set of facilities $i \in \mathcal{F}$ such that $\overline{x}^{(d)}_{ij} > 0$ the set of *distant facilities* of client j and denote it D_j. Observe that for a client j the intersection of C_j and D_j is either empty, or contains exactly one facility. In the latter case, we will say that this facility is both distant and close. Note that, unlike in the UFL problem, we cannot simply split this facility to the close and the distant part, because it is essential that we make at most one connection to this facility in the final integral solution. Let $d^{(max)}_j = c_{i_k j}$ be the distance from client j to the farthest of its close facilities.

3.4 Clustering

We will now construct a family of subsets of facilities $\mathcal{S} \in 2^{\mathcal{F}}$. These subsets $S \in \mathcal{S}$ will be called clusters and they will guide the rounding procedure described next. There will be a client related to each cluster, and each single client j will be related to at most one cluster, which we call S_j.

Not all the clients participate in the clustering process. Clients j with $\overline{r}_j = 1$ and a special facility $i' \in C_j$ (recall that a special facility is a facility that is fully open in $\hat{y}$ but only partially used by j in $\overline{x}$) will be called special and will not take part in the clustering process. Let $\mathcal{C}'$ denote the set of all other, non-special clients. Observe that, as a result of scaling, clients j with $\overline{r}_j \geq 2$ do not have any special facilities among their close facilities (since $\sum_i \overline{x}_{ij} \geq \gamma \overline{r}_j > \overline{r}_j + 1$). As a consequence, there are no special facilities among the close facilities of clients from $\mathcal{C}'$, the only clients actively involved in the clustering procedure.

For each client $j \in \mathcal{C}'$ we will keep two families A_j and B_j of disjoint subsets of facilities. Initially $A_j = \{\{i\} : i \in C_j\}$, i.e., A_j is initialized to contain a singleton set for each close facility of client j; B_j is initially empty. A_j will be used to store these initial singleton sets, but also clusters containing only close facilities of j; B_j will be used to store only clusters that contain at least one close facility of j. When adding a cluster to either A_j or B_j we will remove all the subsets it intersects from both A_j and B_j, therefore subsets in $A_j \cup B_j$ will always be pairwise disjoint.

The family of clusters that we will construct will be a laminar family of subsets of facilities, i.e., any two clusters are either disjoint or one entirely contains the other. One can imagine facilities being leaves and clusters being internal nodes of a forest that eventually becomes a tree, when all the clusters are added.

We will use $\overline{y}(S)$ as a shorthand for $\sum_{i \in S} \overline{y}_i$. Let us define $\underline{y}(S) = \lfloor \overline{y}(S) \rfloor$. As a consequence of using the family of clusters to guide the rounding process, by

Property (P2') of the dependent rounding procedure when applied to a cluster, th quantity $\underline{y}(S)$ lower bounds the number of facilities that will certainly be opened in cluster S. Additionally, let us define the residual requirement of client j to be $rr_j = \overline{r}_j - \sum_{S \in (A_j \cup B_j)} \underline{y}(S)$, that is $\overline{r}_j$ minus a lower bound on the number of facilities that will be opened in clusters from A_j and B_j.

We use the following procedure to compute clusters. While there exists a client $j \in \mathcal{C}'$, such that $rr_j > 0$, take such j with minimal $d_j^{(max)}$ and do the following:

1. Take X_j to be an inclusion-wise minimal subset of A_j, such that $\sum_{S \in X_j} (\overline{y}(S) - \underline{y}(S)) \geq rr_j$. Form the new cluster $S_j = \bigcup_{S \in X_j} S$.
2. Make S_j a new cluster by setting $\mathcal{S} \leftarrow \mathcal{S} \cup \{S_j\}$.
3. Update $A_j \leftarrow (A_j \setminus X_j) \cup \{S_j\}$.
4. For each client j' with $rr_{j'} > 0$ do
   - If $X_j \subseteq A_{j'}$, then set $A_{j'} \leftarrow (A_{j'} \setminus X_j) \cup \{S_j\}$.
   - If $X_j \cap A_{j'} \neq \emptyset$ and $X_j \setminus A_{j'} \neq \emptyset$, then set $A_{j'} \leftarrow A_{j'} \setminus X_j$ and $B_{j'} \leftarrow \{S \in B_{j'} : S \cap S_j = \emptyset\} \cup \{S_j\}$.

Eventually, add a cluster $S_r = \mathcal{F}$ containing all the facilities to the family $\mathcal{S}$.

We call a client j' active in a particular iteration, if before this iteration its residual requirement $rr_j = \overline{r}_j - \sum_{S \in (A_j \cup B_j)} \underline{y}(S)$ was positive. During the above procedure, all active clients j have in their sets A_j and B_j only maximal subsets of facilities, that means they are not subsets of any other clusters (i.e., they are roots of their trees in the current forest). Therefore, when a new cluster S_j is created, it contains all the other clusters with which it has nonempty intersections (i.e., the new cluster S_j becomes a root of a new tree).

We shall now argue that there is enough fractional opening in clusters in A_j to cover the residual requirement rr_j when cluster S_j is to be formed. Consider a fixed client $j \in \mathcal{C}'$. Recall that at the start of the clustering we have $A_j = \{\{i\} : i \in C_j\}$, and therefore $\sum_{S \in A_j} (\overline{y}(S) - \underline{y}(S)) = \sum_{i \in C_j} \overline{y}_i \geq \overline{r}_j = rr_j$. It remains to show, that $\sum_{S \in A_j} (\overline{y}(S) - \underline{y}(S)) - rr_j$ does not decrease over time until client j is considered. When a client j' with $d_{j'}^{(max)} \leq d_j^{(max)}$ is considered and cluster $S_{j'}$ is created, the following cases are possible:

1. $S_{j'} \cap (\bigcup_{S \in A_j} S) = \emptyset$: then A_j and rr_j do not change;
2. $S_{j'} \subseteq (\bigcup_{S \in A_j} S)$: then A_j changes its structure, but $\sum_{S \in A_j} \overline{y}(S)$ and $\sum_{S \in B_j} \underline{y}(S)$ do not change; hence $\sum_{S \in A_j} (\overline{y}(S) - \underline{y}(S)) - rr_j$ also does not change;
3. $S_{j'} \cap (\bigcup_{S \in A_j} S) \neq \emptyset$ and $S_{j'} \setminus (\bigcup_{S \in A_j} S) \neq \emptyset$: then, by inclusion-wise minimality of set $X_{j'}$, we have $\underline{y}(S_{j'}) - \sum_{S \in B_j, S \subseteq S_{j'}} \underline{y}(S) - \sum_{S \in A_j, S \subseteq S_{j'}} \overline{y}(S) \geq 0$; hence, $\sum_{S \in A_j} (\overline{y}(S) - \underline{y}(S)) - rr_j$ cannot decrease.

Let $A'_j = A_j \cap \mathcal{S}$ be the set of clusters in A_j. Recall that all facilities in clusters in A'_j are close facilities of j. Note also that each cluster $S_{j'} \in B_j$ was created from close facilities of a client j' with $d_{j'}^{(max)} \leq d_j^{(max)}$. We also have for each $S_{j'} \in B_j$ that $S_{j'} \cap C_j \neq \emptyset$, hence, by the triangle inequality, all facilities in $S_{j'}$ are at distance at most $3 \cdot d_j^{(max)}$ from j. We thus infer the following

Corollary 2. *The family of clusters $\mathcal{S}$ contains for each client $j \in \mathcal{C}'$ a collection of disjoint clusters $A'_j \cup B_j$ containing only facilities within distance $3 \cdot d_j^{(max)}$, and $\sum_{S \in A'_j \cup B_j} \lfloor \sum_{i \in S} \overline{y}_i \rfloor \geq \overline{r}_j$.*

Note that our clustering is related to, but more complex then the one of Chudak and Shmoys [4] for UFL and of Swamy and Shmoys [11] for FTFL, where clusters are pairwise disjoint and each contains facilities whose fractional opening sums up to or slightly exceeds the value of 1.

3.5 Opening of facilities by dependent rounding

Given the family of subsets $\mathcal{S} \in 2^{\mathcal{F}}$ computed by the clustering procedure from Section 3.4, we can proceed with rounding the fractional opening vector $\overline{y}$ into an integral vector y^R. We do it by applying the rounding technique of Section 2, guided by the family $\mathcal{S}$, which is done as follows.

While there is more than one fractional entry, select a minimal subset of $S \in \mathcal{S}$ which contains more than one fractional entry and apply the rounding procedure to entries of $\overline{y}$ indexed by elements of S until at most one entry in S remains fractional. Eventually, if there remains a fractional entry, round it independently and let y^R be the resulting vector.

Observe that the above process is one of the possible implementations of dependent rounding applied to $\overline{y}$. As a result, the random integral vector y^R satisfies properties (P1),(P2), and (P3). Additionally, property (P2') holds for each cluster $S \in \mathcal{S}$. Hence, at least $\lfloor \sum_{i \in S} \overline{y}_i \rfloor$ entries in each $S \in \mathcal{S}$ are rounded to 1. Therefore, by Corollary 2, we get

Corollary 3. *For each client $j \in \mathcal{C}'$.*

$$|\{i \in \mathcal{F} | y_i^R = 1 \text{ and } c_{ij} \leq 3 \cdot d_j^{(max)}\}| \geq \overline{r}_j.$$

Next, we combine the facilities opened by rounding y^R with facilities opened already when scaling which are recorded in $\tilde{y}$, i.e., we update $\tilde{y} \leftarrow \tilde{y} + y^R$.

Eventually, we connect each client $j \in \mathcal{C}$ to r_j closest opened facilities and code it in $\tilde{x}$.

4 Analysis

We will now estimate the expected cost of the solution $(\tilde{x}, \tilde{y})$. The tricky part is to bound the connection cost, which we do as follows. We argue that a certain fraction of the demand of client j can be satisfied from its close facilities, then some part of the remaining demand can be satisfied from its distant facilities. Eventually, the remaining (not too large in expectation) part of the demand is satisfied via clusters.

4.1 Average distances

Let us consider weighted average distances from a client j to sets of facilities fractionally serving it. Let d_j be the average connection cost in $\overline{x}_{ij}$ defined as

$$d_j = \frac{\sum_{i \in \mathcal{F}} c_{ij} \cdot \overline{x}_{ij}}{\sum_{i \in \mathcal{F}} \overline{x}_{ij}}.$$

Let $d_j^{(c)}$, $d_j^{(d)}$ be the average connection costs in $\overline{x}_{ij}^{(c)}$ and $\overline{x}_{ij}^{(d)}$ defined as

$$d_j^{(c)} = \frac{\sum_{i \in \mathcal{F}} c_{ij} \cdot \overline{x}_{ij}^{(c)}}{\sum_{i \in \mathcal{F}} \overline{x}_{ij}^{(c)}},$$

$$d_j^{(d)} = \frac{\sum_{i \in \mathcal{F}} c_{ij} \cdot \overline{x}_{ij}^{(d)}}{\sum_{i \in \mathcal{F}} \overline{x}_{ij}^{(d)}}.$$

Let R_j be a parameter defined as

$$R_j = \frac{d_j - d_j^{(c)}}{d_j}$$

if $d_j > 0$ and $R_j = 0$ otherwise. Observe that R_j takes value between 0 and 1. $R_j = 0$ implies $d_j^{(c)} = d_j = d_j^{(d)}$, and $R_j = 1$ occurs only when $d_j^{(c)} = 0$. The role played by R_j is that it measures a certain parameter of the instance, big values are good for one part of the analysis, small values are good for the other.

Lemma 1. $d_j^{(d)} \leq d_j(1 + \frac{R_j}{\gamma - 1})$.

Proof. Recall that $\sum_{i \in \mathcal{F}} \overline{x}_{ij}^{(c)} = \overline{r}_j$ and $\sum_{i \in \mathcal{F}} \overline{x}_{ij}^{(d)} \geq (\gamma - 1) \cdot \overline{r}_j$. Therefore, we have $(d_j^{(d)} - d_j) \cdot (\gamma - 1) \leq (d_j - d_j^{(c)}) \cdot 1 = R_j \cdot d_j$, which can be rearranged to get $d_j^{(d)} \leq d_j(1 + \frac{R_j}{\gamma - 1})$. $\square$

Finally, observe that the average distance from j to the distant facilities of j gives an upper bound on the maximal distance to any of the close facilities of j. Namely, $d_j^{(max)} \leq d_j^{(d)}$.

4.2 Amount of service from close and distant facilities

We now argue that in the solution $(\tilde{x}, \tilde{y})$, a certain portion of the demand is expected to be served by the close and distant facilities of each client. Recall that for a client j it is possible, that there is a facility that is both its close and its distant facility. Once we have a solution that opens such a facility, we would like to say what fraction of the demand is served from the close facilities. To make our analysis simpler we will toss a properly biased coin to decide if using this facility counts as using a close facility. With this trick we, in a sense,

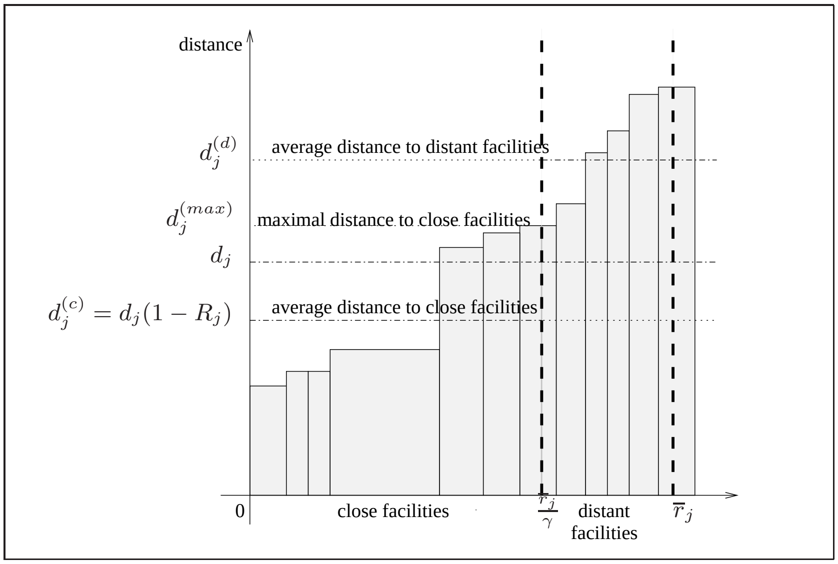

Fig. 1. Distances to facilities serving client j in $\overline{x}$. The width of a rectangle corresponding to facility i is equal to $\overline{x}_{ij}$. Figure helps to understand the meaning of R_j.

split such a facility into a close and a distant part. Note that we can only do it for this part of the analysis, but not for the actual rounding algorithm from Section 3.5. Applying the above-described split of the undecided facility, we get that the total fractional opening of close facilities of client j is exactly $\overline{r}_j$, and the total fractional opening of both close and distant facilities is at least $\gamma \cdot \overline{r}_j$. Therefore, Corollary 1 yields the following:

Corollary 4. *The amount of close facilities used by client j in a solution described in Section 3.5 is expected to be at least $(1 - \frac{1}{e}) \cdot \overline{r}_j$.*

Corollary 5. *The amount of close and distant facilities used by client j in a solution described in Section 3.5 is expected to be at least $(1 - \frac{1}{e^\gamma}) \cdot \overline{r}_j$.*

Motivated by the above bounds we design a selection method to choose a (large-enough in expectation) subset of facilities opened around client j:

Lemma 2. *For $j \in \mathcal{C}'$ we can select a subset F_j of open facilities from $C_j \cup D_j$ such that:*

$$|F_j| \leq \overline{r}_j \text{ (with probability 1)},$$
$$E[F_j] = (1 - \frac{1}{e^\gamma}) \cdot \overline{r}_j,$$
$$E[\sum_{i \in F_j} c_{ij}] \leq ((1 - 1/e) \cdot \overline{r}_j) \cdot d_j^{(c)} + (((1 - \frac{1}{e^\gamma}) - (1 - 1/e)) \cdot \overline{r}_j) \cdot d_j^{(d)}.$$

A technical but not difficult proof we sketch in the Appendix.

4.3 Calculation

We can now combine the pieces into the algorithm ALG:

1. solve the LP-relaxation of (1)-(5);
2. scale the fractional solution as described in Section 3.2;
3. create a family of clusters as described in Section 3.4;
4. round the fractional openings as described in Section 3.5;
5. connect each client j to r_j closest open facilities;
6. output the solution as $(\tilde{x}, \tilde{y})$.

Theorem 4. *ALG is an* 1.7245*-approximation algorithm for FTFL.*

Proof. First observe that the solution produced by ALG is trivially feasible to the original problem (1)-(5), as we simply choose different r_j facilities for client j in step 5. What is less trivial is that all the r_j facilities used by j are within a certain small distance. Let us now bound the expected connection cost of the obtained solution.

For each client $j \in \mathcal{C}$ we get $r_j - \overline{r}_j$ facilities opened in Step 2. As we already argued in Section 3.2, we can afford to connect j to these facilities and pay the connection cost from the difference between $\sum_i c_{ij}\hat{x}_{ij}$ and $\sum_i c_{ij}\overline{x}_{ij}$. We will now argue, that client j can connect to the remaining $\overline{r}_j$ with the expected connection cost bounded by $\sum_i c_{ij}\overline{x}_{ij}$.

For a special client $j \in (\mathcal{C} \setminus \mathcal{C}')$ we have $\overline{r}_j = 1$ and already in Step 2 one special facility at distance $d_j^{(max)}$ from j is opened. We cannot always just connect j to this facility, since $d_j^{(max)}$ may potentially be bigger then $\gamma \cdot d_j$. What we do instead is that we first look at close facilities of j that, as a result of the rounding in Step 4, with a certain probability, give one open facility at a small distance. By Corollary 4 this probability is at least $1 - 1/e$. It is easy to observe that the expected connection cost to this open facility is at most $d_j^{(c)}$. Only if no close facility is open, we use the special facility, which results in the expected connection cost of client j being at most

$$(1-1/e)d_j^{(c)}+(1/e)d_j^{(d)} \leq (1-1/e)d_j^{(c)}+(1/e)d_j(1+\frac{R_j}{\gamma-1}) \leq d_j(1+1/(e\cdot(\gamma-1)) \leq \gamma\cdot d_j,$$

where the first inequality is a consequence of Lemma 1, and the last one is a consequence of the choice of $\gamma \approx 1.7245$.

In the remaining, we only look at non-special clients $j \in \mathcal{C}'$. By Lemma 2, client j can select to connect itself to the subset of open facilities F_j, and pay for this connection at most $((1-1/e)\cdot\overline{r}_j)\cdot d_j^{(c)} + (((1-\frac{1}{e^\gamma}) - (1-1/e))\cdot\overline{r}_j)\cdot d_j^{(d)}$ in expectation. The expected number of facilities needed on top of those from F_j is $\overline{r}_j - E[|F_j|] = (\frac{1}{e^\gamma}\cdot\overline{r}_j)$. These remaining facilities client j gets deterministically within the distance of at most $3 \cdot d_j^{(max)}$, which is possible by the properties of the rounding procedure described in Section 3.5, see Corollary 3. Therefore, the expected connection cost to facilities not in F_j is at most $(\frac{1}{e^\gamma}\cdot\overline{r}_j)\cdot(3\cdot d_j^{(max)})$.

Concluding, the total expected connection cost of j may be bounded by

$$((1-1/e)\cdot\overline{r}_j)\cdot d_j^{(c)}+(((1-\frac{1}{e^\gamma})-(1-1/e))\cdot\overline{r}_j)\cdot d_j^{(d)}+(\frac{1}{e^\gamma}\cdot\overline{r}_j)\cdot(3\cdot d_j^{(max)})$$
$$\le \overline{r}_j\cdot\left((1-1/e)\cdot d_j^{(c)}+((1-\frac{1}{e^\gamma})-(1-1/e))\cdot d_j^{(d)}+\frac{1}{e^\gamma}\cdot(3d_j^{(d)})\right)$$
$$=\overline{r}_j\cdot\left((1-1/e)\cdot d_j^{(c)}+((1+\frac{2}{e^\gamma})-(1-1/e))\cdot d_j^{(d)}\right)$$
$$\le \overline{r}_j\cdot\left((1-1/e)\cdot(1-R_j)\cdot d_j+((1+\frac{2}{e^\gamma})-(1-1/e))\cdot(1+\frac{R_j}{\gamma-1})\cdot d_j\right)$$
$$=\overline{r}_j\cdot d_j\cdot\left((1-1/e)\cdot(1-R_j)+(\frac{2}{e^\gamma}+1/e)\cdot(1+\frac{R_j}{\gamma-1})\right)$$
$$=\overline{r}_j\cdot d_j\cdot\left((1-1/e)+(\frac{2}{e^\gamma}+1/e)+R_j\cdot((\frac{2}{e^\gamma}+1/e)\cdot\frac{1}{\gamma-1}-(1-1/e))\right)$$
$$=\overline{r}_j\cdot d_j\cdot\left(1+\frac{2}{e^\gamma}+R_j\cdot\left(\frac{(\frac{2}{e^\gamma}+1/e)}{\gamma-1}-(1-1/e)\right)\right),$$

where the second inequality follows from Lemma 1 and the definition of R_j.

Observe that for $1<\gamma<2$, we have $\frac{(\frac{2}{e^\gamma}+1/e)}{\gamma-1}-(1-1/e)>0$. Recall that by definition, $R_j\le 1$; so, $R_j=1$ is the worst case for our estimate, and therefore

$$\overline{r}_j\cdot d_j\cdot\left(1+\frac{2}{e^\gamma}+R_j\cdot\left(\frac{(\frac{2}{e^\gamma}+1/e)}{\gamma-1}-(1-1/e)\right)\right)\le\overline{r}_j\cdot d_j\cdot(1/e+\frac{2}{e^\gamma})(1+\frac{1}{\gamma-1}).$$

Recall that $\overline{x}$ incurs, for each client j, a fractional connection cost $\sum_{i\in\mathcal{F}}c_{ij}\overline{x}_{ij}\ge\gamma\cdot\overline{r}_j\cdot d_j$. We fix $\gamma=\gamma_0$, such that $\gamma_0=(1/e+\frac{2}{e^{\gamma_0}})(1+\frac{1}{\gamma_0-1})\le 1.7245$.

To conclude, the expected connection cost of j to facilities opened during the rounding procedure is at most the fractional connection cost of $\overline{x}$. The total connection cost is, therefore, at most the connection cost of $\hat{x}$, which is at most γ times the connection cost of x^*.

By property (P1) of dependent rounding, every single facility i is opened with the probability $\hat{y}_i$, which is at most γ times y_i^*. Therefore, the total expected cost of the solution produced by ALG is at most $\gamma\approx 1.7245$ times the cost of the fractional optimal solution (x^*,y^*). □

5 Concluding remarks

We have presented improved approximation algorithms for the metric Fault-Tolerant Uncapacitated Facility Location problem. The main technical innovation is the usage and analysis of dependent rounding in this context. We believe that variants of dependent rounding will also be fruitful in other location problems. Finally, we conjecture that the approximation threshold for both UFL and FTFL is the value $1.46\cdots$ suggested by [5]; it would be very interesting to prove or refute this.

Acknowledgment. We thank the IPCO referees for their helpful comments.

References

1. A. Ageev and M. Sviridenko. Pipage rounding: a new method of constructing algorithms with proven performance guarantee. *Journal of Combinatorial Optimization*, 8(3):307–328, 2004.
2. J. Byrka. An optimal bifactor approximation algorithm for the metric uncapacitated facility location problem. In *APPROX-RANDOM*, pages 29–43, 2007.
3. J. Byrka, A. Srinivasan, and C. Swamy. Fault-tolerant facility location: a randomized dependent LP-rounding algorithm. arXiv:1003.1295v1.
4. F. A. Chudak and D. B. Shmoys. Improved approximation algorithms for the uncapacitated facility location problem. *SIAM J. Comput.*, 33(1):1–25, 2003.
5. S. Guha and S. Khuller. Greedy strikes back: Improved facility location algorithms. *J. Algorithms*, 31(1):228–248, 1999.
6. S. Guha, A. Meyerson, and K. Munagala. A constant factor approximation algorithm for the fault-tolerant facility location problem. *J. Algorithms*, 48(2):429–440, 2003.
7. K. Jain and V. V. Vazirani. An approximation algorithm for the fault tolerant metric facility location problem. *Algorithmica*, 38(3):433–439, 2003.
8. J.-H. Lin and J. S. Vitter. epsilon-approximations with minimum packing constraint violation (extended abstract). In *STOC*, pages 771–782, 1992.
9. D. B. Shmoys, É. Tardos, and K. Aardal. Approximation algorithms for facility location problems (extended abstract). In *STOC*, pages 265–274, 1997.
10. A. Srinivasan. Distributions on level-sets with applications to approximation algorithms. In *FOCS*, pages 588–597, 2001.
11. C. Swamy and D. B. Shmoys. Fault-tolerant facility location. *ACM Transactions on Algorithms*, 4(4), 2008.

Appendix

Proof. **(for Lemma 2, a sketch)** Given client j, fractional facility opening vector $\overline{y}$, distances c_{ij}, requirement $\overline{r}_j$, and facility subsets C_j and D_j, we will describe how to randomly choose a subset of at most $k = \overline{r}_j$ open facilities from $C_j \cup D_j$ with the desired properties.

For this argument we assume that all the numbers are rational. Recall that the opening of facilities is decided in a dependent rounding routine, that in a single step couples two fractional entries to leave at most one of them fractional.

Observe that, for the purpose of this argument, we can split a single facility into many identical copies with smaller fractional opening. One can think that the input facilities and their original openings were obtained along the process of dependent rounding applied to the multiple "small" copies that we prefer to consider here. Therefore, without loss of generality, we can assume that all the facilities have fractional opening equal ϵ, i.e., $\overline{y}_i = \epsilon$ for all $i \in C_j \cup D_j$. Moreover, we can assume that sets C_j and D_j are disjoint.

By renaming facilities we obtain that $C_j = \{1, 2, \ldots, |C_j|\}$, $D_j = \{|C_j| + 1, \ldots, |C_j| + |D_j|\}$, and $c_{ij} \leq c_{i'j}$ for all $1 \leq i < i' \leq |C_j| + |D_j|$.

Consider random set $S_0 \subseteq C_j \cup D_j$ created as follows. Let $\hat{y}$ be the outcome of rounding the fractional opening vector $\overline{y}$ with the dependent rounding procedure, and define $S_0 = \{i : \hat{y}_i = 1, (\sum_{j<i} \hat{y}) < k\}$. By Corollary 1, we have that

$\mathbf{E}[|S_0|] \geq k \cdot (1 - \exp(-\mathrm{Sum}_{C_j \cup D_j}(\overline{y})/k))$. Define random set S_α for $\alpha \in (0, |C_j| + |D_j|]$ as follows. For $i = 1, 2, \ldots \lfloor |C_j| + |D_j| - \alpha \rfloor$ we have $i \in S_\alpha$ if and only if $i \in S_0$. For $i = \lceil |C_j| + |D_j| - \alpha \rceil$, in case $i \in S_0$ we toss a (suitably biased) coin and include i in S_α with probability $\alpha - \lfloor \alpha \rfloor$. For $i > \lceil |C_j| + |D_j| - \alpha \rceil$ we deterministically have $i \notin S_\alpha$.

Observe that $\mathbf{E}[|S_\alpha|]$ is a continuous monotone non-increasing function of α, hence there is α_0 such that $\mathbf{E}[|S_{\alpha_0}|] = k \cdot (1 - \exp(-\mathrm{Sum}_{C_j \cup D_j}(\overline{y})/k))$. We fix $F_j = S_{\alpha_0}$ and claim that it has the desired properties. By definition, we have $\mathbf{E}[|F_j|] = k \cdot (1 - \exp(-\mathrm{Sum}_{C_j \cup D_j}(\overline{y})/k)) = (1 - \frac{1}{e^\gamma}) \cdot \overline{r}_j$. We next show that the expected total connection cost between j and facilities in F_j is not too large.

Let $p_i^\alpha = Pr[i \in S_\alpha]$ and $p'_i = p_i^{\alpha_0} = Pr[i \in F_j]$. Consider the cumulative probability defined as $cp_i^\alpha = \sum_{j \leq i} p_j^\alpha$. Observe that application of Corollary 1 to subsets of first i elements of $C_j \cup D_j$ yields $cp_i^0 \geq k \cdot (1 - \exp(-\epsilon i/k))$ for $i = 1, \ldots, |C_j| + |D_j|$. Since $(1 - \exp(-\epsilon i/k))$ is a monotone increasing function of i one easily gets that also $cp_i^\alpha \geq k \cdot (1 - \exp(-\epsilon i/k))$ for $\alpha \leq \alpha_0$ and $i = 1, \ldots, |C_j| + |D_j|$. In particular, we get $cp_{|C_j|}^{\alpha_0} \geq k \cdot (1 - \exp(-\epsilon |C_j|/k))$.

Since $(1 - \exp(-\epsilon i/k))$ is a concave function of i, we also have $cp_i^{\alpha_0} \geq k \cdot (1 - \exp(-\epsilon i/k)) \geq (i/|C_j|) \cdot k \cdot (1 - \exp(-\epsilon |C_j|/k)) = (i/|C_j|) \cdot (1 - \frac{1}{e}) \cdot \overline{r}_j$ for all $1 \leq i \leq |C_j|$. Analogously, we get

$$\begin{aligned}
cp_i^{\alpha_0} &\geq (k \cdot (1 - \exp(-\epsilon |C_j|/k))) \\
&\quad + ((i - |C_j|)/|D_j|) \cdot k \cdot \left((1 - \exp(\frac{-\epsilon(|C_j| + |D_j|)}{k})) - (1 - \exp(-\epsilon |C_j|/k)) \right) \\
&= \overline{r}_j \cdot (1 - \frac{1}{e}) + \overline{r}_j \cdot \left(((i - |C_j|)/|D_j|)((1 - \frac{1}{e^\gamma}) - (1 - \frac{1}{e})) \right)
\end{aligned}$$

for all $|C_j| < i \leq |C_j| + |D_j|$.

Recall that we want to bound $\mathbf{E}[\sum_{i \in F_j} c_{ij}] = \sum_{i \in C_j \cup D_j} p'_i c_{ij}$. From the above bounds on the cumulative probability, we get that, by shifting the probability from earlier facilities to later ones, one can obtain a probability vector p'' with $p''_i = 1/|C_j| \cdot ((1 - \frac{1}{e}) \cdot \overline{r}_j)$ for all $1 \leq i \leq |C_j|$, and $p''_i = 1/|D_j| \cdot ((1 - \frac{1}{e^\gamma}) - (1 - \frac{1}{e})) \cdot \overline{r}_j$ for all $|C_j| < i \leq |C_j| + |D_j|$. As connection costs c_{ij} are monotone non-decreasing in i, shifting the probability never decreases the weighted sum; hence,

$$\begin{aligned}
\mathbf{E}[\sum_{i \in F_j} c_{ij}] &= \sum_{i \in F_j} p'_i c_{ij} \\
&\leq \sum_{i \in F_j} p''_i c_{ij} \\
&= \sum_{1 \leq i \leq |C_j|} 1/|C_j| \cdot ((1 - \frac{1}{e}) \cdot \overline{r}_j) c_{ij} \\
&\quad + \sum_{|C_j| < i \leq |C_j| + |D_j|} 1/|D_j| \cdot (((1 - \frac{1}{e^\gamma}) - (1 - \frac{1}{e})) \cdot \overline{r}_j) c_{ij} \\
&= ((1 - 1/e) \cdot \overline{r}_j) \cdot d_j^{(c)} + (((1 - \frac{1}{e^\gamma}) - (1 - 1/e)) \cdot \overline{r}_j) \cdot d_j^{(d)}. \qquad \square
\end{aligned}$$