New Approximation Algorithms for the Unsplittable Capacitated Facility Location Problem *

Babak Behsaz[1], Mohammad R. Salavatipour[1], and Zoya Svitkina[2]

[1] Dept. of Computing Sci., Univ. of Alberta, Edmonton, Alberta T6G 2E8, Canada,
{behsaz, mrs}@ualberta.ca
[2] Google Inc., 1600 Amphitheatre Parkway, Mountain View, CA 94043, USA,
zoya@cs.cornell.edu

Abstract. In this paper, we consider the Unsplittable (hard) Capacitated Facility Location Problem (UCFLP) with uniform capacities and present some new approximation algorithms for it. This problem is a generalization of the classical facility location problem where each facility can serve at most u units of demand and each client must be served by *exactly one* facility. It is known that it is NP-hard to approximate this problem within any factor without violating the capacities. So we consider bicriteria (α, β)-approximations where the algorithm returns a solution whose cost is within factor α of the optimum and violates the capacity constraints within factor β. We present a framework for designing bicriteria approximation algorithms and show two new approximation algorithms with factors $(10.173, 3/2)$ and $(30.432, 4/3)$. These are the first algorithms with constant approximation in which the violation of capacities is below 2. The heart of our algorithms is a reduction from the UCFLP to a restricted version of the problem. One feature of this reduction is that any $(O(1), 1+\epsilon)$-approximation for the restricted version implies an $(O(1), 1+\epsilon)$-approximation for the UCFLP for any constant $\epsilon > 0$ and we believe our techniques might be useful towards finding such approximations or perhaps $(f(\epsilon), 1+\epsilon)$-approximation for the UCFLP for some function f. In addition, we present a quasi-polynomial time $(1+\epsilon, 1+\epsilon)$-approximation for the (uniform) UCFLP in Euclidean metrics, for any constant $\epsilon > 0$.

Keywords: approximation algorithms, unsplittable capacitated facility location problem, Euclidean metrics

1 Introduction

We consider the Unsplittable Capacitated Facility Location Problem (UCFLP) with uniform capacities. In this problem, we are given a set of clients C and

* Research supported by Alberta Innovates Future Technologies, Canada. The second author was additionally supported by NSERC.

facilities F where client j has demand d_j and each facility i has capacity u and opening cost f_i. We have a metric cost function c_{ij} which denotes the cost of serving one unit of demand of client j at facility i. The goal is to open a subset of facilities $I \subseteq F$ and assign each client j to *exactly one* open facility $\phi(j)$ to serve its entire demand d_j so that the total amount of demand assigned to each open facility is no more than u, while minimizing the total cost of opening facilities and connecting (serving) clients to them, *i.e.*, minimizing $\sum_{i \in I} f_i + \sum_{j \in C} d_j \cdot c_{\phi(j)j}$. This problem generalizes the bin packing, the minimum makespan, and some facility location problems. If the demands of clients can be served by multiple open facilities, then we have the *splittable* version of the problem (called splittable CFLP). If each facility can be opened multiple times then we have the so-called *soft-capacitated* version. Each of these relaxations (*i.e.*, allowing splitting the demands of clients and/or having multiple copies of each facility) makes the problem significantly easier as discussed below.

By a simple reduction from the partition problem, one can show that any approximation algorithm for the uniform UCFLP violates the capacities of $\Omega(n)$ facilities unless P=NP. Thus, research has focused on the design of bicriteria approximation algorithms. An (α, β)-*approximation* for the UCFLP returns a solution whose cost is within factor α of the optimum and violates the capacity constraints within factor β. It should be noted that if we violate capacity of a facility within factor β, we must pay β times its opening cost. In the context of approximation algorithms, Shmoys, Tardos, and Aardal [9] were the first to consider this problem and presented a $(9, 4)$-approximation algorithm. They used a filtering and rounding technique to get an approximation algorithm for the splittable version and used a rounding for the generalized assignment problem (GAP) [8] to obtain their algorithm for the unsplittable version. This technique of reducing the unsplittable version using the rounding for the GAP to the splittable version was a cornerstone of the subsequent approximation algorithms. Korupolu, Plaxton, and Rajaraman [5] gave the first constant factor approximation algorithm for the splittable hard capacitated version, and applied the GAP rounding technique of [9] to get a $(O(1), 2)$-approximation algorithm for the UCFLP. Applying the current best approximation algorithms for the splittable capacitated version with non-uniform capacities (*i.e.*, each facility has capacity u_i) [10] and uniform capacities [1], one can get factor $(11, 2)$ and $(5, 2)$ approximation algorithms for the UCFLP with non-uniform and uniform capacities, respectively.

Recently, Bateni and Hajiaghayi [3] modeled an assignment problem in content distribution networks by the UCFLP. In this application, it is crucial to keep the violation of capacities as small as possible. Motivated by this strict requirement on capacities, the authors of [3] designed a $(1+\epsilon, 1+\epsilon)$-approximation algorithm for *tree metrics* (for any constant $\epsilon > 0$) using a dynamic programming approach. They also presented a quasi-polynomial time $(1+\epsilon, 1+\epsilon)$-approximation algorithm (again for trees) for the non-uniform capacity case. Using Fakcharoenphol *et al.*'s improvement of Bartal's machinery this implies a polynomial time $(O(\log n), 1 + \epsilon)$-approximation algorithm for almost uniform capacities and a

quasi-polynomial time $(O(\log n), 1+\epsilon)$-approximation algorithm for non-uniform case for an arbitrary constant $\epsilon > 0$.

All the known constant-factor algorithms for the UCFLP violate the capacity constraints by a factor of at least 2 which is mainly due to using the rounding algorithm for GAP [8]. Also, the algorithm of [3] (although has $1 + \epsilon$ violation) is not a constant factor approximation. We present the first constant factor approximation algorithms with capacity violation factor less than 2. Particularly, we present two approximation algorithms with factors $(10.173, 3/2)$ and $(30.432, 4/3)$ for the UCFLP. We also consider the UCFLP restricted to Euclidean metrics and give a $(1+\epsilon, 1+\epsilon)$-approximation that runs in the quasi-polynomial time.

1.1 Related Works

Perhaps the most well-studied facility location problem is the *uncapacitated* facility location problem (UFLP). In this problem, we do not have the capacity constraints and we only need to decide which facilities to open; as each client will be assigned to its closest open facility. The first constant approximation for the UFLP was a 3.16-approximation algorithm by Shmoys, Tardos, and Aardal [9]. The ratio for the UFLP was improved in a series of papers down to 1.488 [6]. On the negative side, a result of Guha and Khuller [4], combined with an observation of Sviridenko implies 1.463-hardness for the UFLP.

Capacitated facility location problems have also received a lot of attention. The solutions of the soft capacitated version have a similar structure to the solution of uncapacitated version and this problem can be reduced to the UFLP. For example, see [7] for a reduction. This paper gives the current best ratio, 2, for the soft capacitated version to the best of our knowledge. Since Mahdian *et al.* [7] reduce the problem to the UFLP, they give a solution that sends each client to exactly one facility. As a result, this solution is also feasible for the unsplittable case and is a 2-approximation for this case too. This comes from the fact that the optimal value of splittable version is a lower-bound for the optimal value of the unsplittable version. In contrast, there is an important distinction between the splittable and unsplittable case in the presence of hard capacities, because even checking the feasibility of the latter becomes NP-hard and we need bicriteria algorithms for the latter (see discussions above). In a series of local search algorithms, the ratio for the splittable CFLP with non-uniform capacities decreased to $5.83 + \epsilon$ [10] and with uniform capacities decreased to 3 [1]. It should be noted that all the known LP relaxations for both the splittable and unsplittable versions have super-constant integrality gap in the general case of the problems.

1.2 The main results and techniques

Recall that given an instance (F, C) of the UCFLP with opening costs f_i, demands d_j, and connection costs c_{ij}, a solution is a subset I of facilities to open

along with assignment function $\phi : C \to I$. We use $c_f(\phi)$ to denote the facility cost and $c_s(\phi)$ to denote the service cost; thus $c(\phi) = c_f(\phi) + c_s(\phi)$ will be the total cost. Since all capacities are uniform, by a simple scaling, we can assume that all of them are 1 and all the client demands are at most 1. As we explained before, we are interested in $(O(1), \beta)$-approximation algorithms for some $\beta < 2$. We define a restricted version of the problem and show that finding a good approximation algorithm for this restricted version would imply a good approximation for the general version.

Definition 1. *An ϵ-restricted UCFLP, denoted by RUCFLP(ϵ), instance is an instance of the UCFLP in which each demand has size more than ϵ, i.e., $\epsilon < d_j \leq 1$ for all $j \in C$.*

The following theorem establishes the reduction from the general instances of the UCFLP to the restricted version. Here, the general idea is that if we assign the large clients oblivious to small clients, we can fractionally assign the small clients without paying too much. We use the maximum-flow minimum-cut theorem to show this. Then we can round this fractional assignment of small clients with the GAP rounding technique [8].

Theorem 1. *If $\mathcal{A}$ is an $(\alpha(\epsilon), \beta(\epsilon))$-approximation algorithm for the RUCFLP(ϵ) with running time $\tau(\mathcal{A})$ then there is a $(g(\epsilon, \alpha(\epsilon)), \max\{\beta(\epsilon), 1+\epsilon\})$-approximation algorithm for the UCFLP whose running time is polynomial in $\tau(\mathcal{A})$ and the instance size, where $g(\epsilon, \alpha(\epsilon))$ is a function of ϵ and $\alpha(\epsilon)$, and is linear in $\alpha(\epsilon)$.*

Corollary 1. *For any constant $\epsilon > 0$, an $(\alpha(\epsilon), 1+\epsilon)$-approximation algorithm for the RUCFLP(ϵ) yields an $(O(\alpha(\epsilon)), 1+\epsilon)$-approximation for the UCFLP. Particularly, when $\alpha(\epsilon)$ is a constant, we have a constant approximation for the UCFLP with a $(1+\epsilon)$ violation of capacities in polynomial time.*

This reduction shows it is sufficient to consider large clients only, which may open the possibility of designing algorithms using some of the techniques used in the bin packing type problems. We believe that one can find an $(O(1), 1+\epsilon)$-approximation algorithm for the RUCFLP(ϵ). If one finds such an algorithm, the above corollary shows that we have an $(O(1), (1+\epsilon))$-approximation for the UCFLP. As an evidence for this, we find approximation algorithms for the RUCFLP(1/2) and the RUCFLP(1/3). For the RUCFLP(1/2), we present an exact algorithm and for the RUCFLP(1/3), we present a $(21, 1)$-approximation algorithm. These, together with Theorem 1 imply:

Theorem 2. *There is a polynomial time $(10.173, 3/2)$-approximation algorithm for the UCFLP.*

Theorem 3. *There is a polynomial time $(30.432, 4/3)$-approximation algorithm for the UCFLP.*

Finally, we give a QPTAS for Euclidean metrics. Here, we employ a dynamic programming technique and combine the shifted quad-tree dissection of Arora [2], some ideas from [3], and some new ideas to design a dynamic programming.

Theorem 4. *There exists a $(1+\epsilon, 1+\epsilon)$-approximation algorithm for the Euclidean UCFLP in $\mathbb{R}^2$ with running time in quasi-polynomial for any constant $\epsilon > 0$.*

Although this theorem is presented for $\mathbb{R}^2$, it can be generalized to $\mathbb{R}^d$ for constant $d > 2$. Due to lack of space, we defer the proof of Theorem 4 to the full version.

The rest of this paper is organized as follows. In Section 2, we prove Theorem 1. Next, we present approximation algorithms for the RUCFLP(1/2) and RUCFLP(1/3), which also prove weaker versions of Theorems 2 and 3 (see the full version for improved ratios). Finally, in Section 4, we conclude the paper.

2 Reduction to the Restricted UCFLP

In this section, we prove Theorem 1. Let $L = \{j \in C : d_j > \epsilon\}$ be the set of large clients and $S = C \backslash L$ be the set of small clients[3]. We call two assignment $\phi_1 : C_1 \to F_1$ and $\phi_2 : C_2 \to F_2$ consistent if $\phi_1(j) = \phi_2(j)$ for all $j \in C_1 \cap C_2$. The high level idea of the algorithm (Algorithm 1) is as follows. We first ignore the small clients and solve the problem restricted to only the large clients by running algorithm $\mathcal{A}$ of Theorem 1. We can show that given a good assignment of large clients, there exists a good assignment of all the clients (large and small) which is consistent with this assignment of large clients, i.e. a solution which assigns the large clients the same way that $\mathcal{A}$ does, whose cost is not far from the optimum cost. More specifically, we show there is a *fractional* (i.e. splittable) assignment of small clients that together with the assignment of large clients obtained from $\mathcal{A}$ gives an approximately good solution. Having this property, we try to find a fractional assignment of small clients. To assign the small clients, we update the capacities and the opening costs of facilities with respect to the assignment of large clients (according to the solution of $\mathcal{A}$). Then, we fractionally assign small clients and round this fractional assignment at the cost of violating the capacities within factor $1 + \epsilon$.

First, we formally prove the property that given assignment of large clients, there is a feasible *fractional* assignment of small clients with an acceptable cost. Note that we do not open facilities fractionally and a fractional assignment of demands of (small) clients is essentially equivalent to splitting their demands between multiple open facilities. We should point out that the proof of this property is only an existential result and we do not actually find the assignment in the proof. We only use this lemma to bound the cost of our solution. Let OPT be an optimum solution which opens set I^* of facilities and with assignment of clients $\phi^* : C \to I^*$. We use $\phi^*_L : L \to I^*$ and $\phi^*_S : S \to I^*$ to denote the restriction of ϕ^* to large and small clients, respectively. Here, $\phi^{-1}(i)$ is the

[3] We should point out that the definitions of L and S are with respect to a given parameter ϵ. Since throughout the following sections, this parameter is the same for all statements, in the interest of brevity, we use this notation instead of $L(\epsilon)$ and $S(\epsilon)$.

Algorithm 1 Algorithm for the UCFLP by reduction to the RUCFLP(ϵ)

Require: An instance of UCFLP, an $\epsilon > 0$, and the algorithm $\mathcal{A}$ for the RUCFLP(ϵ)
Ensure: A subset $I \subseteq F$ of facilities to open and an assignment of clients $\phi : C \to I$

1: Let $L = \{j \in C : d_j > \epsilon\}$ and $S = C \backslash L$. Assign the clients in L by running $\mathcal{A}$. Let I_L be the opened facilities and $\phi_L : L \to I_L$ be the assignment found by $\mathcal{A}$.
2: For $i \in I_L$, set $f_i = 0$, and set $u'_i = \max\{0, 1 - \sum_{j \in \phi_L^{-1}(i)} d_j\}$ be the new capacity of facility i. Assign the clients in S with respect to updated opening costs and capacities by an approximation algorithm for the splittable CFLP with non-uniform capacities. Let I_S be the new set of opened facilities and $\phi'_S : S \to I'_S$ be the assignment function, where $I'_S \subseteq I_S \cup I_L$.
3: Round the splittable assignment ϕ'_S using algorithm of [8] to find an unsplittable assignment $\phi_S : S \to I'_S$.
4: Let $I = I'_S \cup I_L$ and define $\phi : C \to I$ as $\phi(j) = \phi_S(j)$ if $j \in S$ and otherwise $\phi(j) = \phi_L(j)$. Return ϕ and I.

set of clients assigned to facility i by the assignment ϕ and for a $F' \subseteq F$, $\phi^{-1}(F') = \cup_{i \in F'} \phi^{-1}(i)$.

Lemma 1. *Suppose I_L is a set of open facilities and $\phi_L : L \to I_L$ is an arbitrary (not necessarily capacity respecting) assignment of large clients. Given the assignment ϕ_L, there exists a feasible fractional assignment of small clients, $\phi''_S : S \to I''_S$ such that $c_s(\phi''_S) \leq c_s(\phi^*) + c_s(\phi_L)$ and $c_f(\phi''_S) \leq c_f(\phi^*)$.*

Proof. Let $u'_i = \max\{0, 1 - \sum_{j \in \phi_L^{-1}(i)} d_j\}$, i.e., the amount of capacity left for facility i after the assignment of large clients based on ϕ_L. We assume we open all the open facilities in the optimum solution, I^* (if not already open in I_L). Let $I''_S = I_L \cup I^*$. To show the existence of ϕ''_S, first we move the demands of small clients to the facilities in I^* based on ϕ^*_S and we pay $c_s(\phi^*_S)$ for this. So now the demands of small clients are located at facilities in I^*. However, a facility $i \in I''_S$ has only u'_i residual capacity left (after commiting parts of its capacity for the large clients assigned to it by ϕ_L) and this capacity may not be enough to serve the demands of small clients moved to that location. In order to rectify this, we will fractionally redistribute the demands of these small clients between facilities (in I''_S) in such a way that we do not violate capacities u'_i. In this redistribution, we only use the edges used in ϕ_L or ϕ^*_L and if an edge is used to assign large client j to facility i (in ϕ_L or ϕ^*_L), we move at most d_j units of demands of small clients along this edge. Therefore, we pay at most $c_s(\phi_L) + c_s(\phi^*_L)$ in this redistribution. Thus, by the Triangle Inequality, the connection cost of the fractional assignment of small clients obtained at the end is bounded by $c_s(\phi^*_S) + c_s(\phi^*_L) + c_s(\phi_L) = c_s(\phi^*) + c_s(\phi_L)$. Since we only open facilities in the optimum solution (on top of what is already open in I_L) the extra facility cost (for assignment ϕ''_S) is bounded by the facility cost of the optimum.

This process of moving the small client demands can be alternatively thought in the following way. We start from the optimum assignment ϕ^* and change the assignment of large clients to get an assignment identical to ϕ_L for those in L.

Specifically, we change the assignment of a large client j from $i' = \phi^*(j)$ to $i = \phi_L(j)$. This switch increases the amount of demands served at i by d_j and decreases the amount of demand served at i' by d_j. After doing all these switches we might have more demand at some facilities than their capacities, while the total demands assigned to some facilities might be less than 1. To resolve this problem, we try to redistribute (fractionally) the demands of small clients so that there is no capacity violation and we use the max-flow min-cut theorem to show that this redistribution is possible. The details of this part appear in the full version. □

Proof of Theorem 1. Since the cost of the optimum solution for the instance consisting of only the large clients is clearly no more than that of the original instance, after Step 1 of Algorithm 1, we have an assignment ϕ_L such that $c(\phi_L) \leq \alpha(\epsilon)c(\phi_L^*)$ and it violates the capacities by a factor of at most $\beta(\epsilon)$. By Lemma 1, given ϕ_L, there is a feasible fractional assignment ϕ_S'' for small clients such that $c_s(\phi_S'') \leq c_s(\phi^*) + c_s(\phi_L)$ and $c_f(\phi_S'') \leq c_f(\phi^*)$.

In Step 2, consider the instance of the splittable CFLP consisting of the small clients and the residual facility opening costs and capacities as defined. We use an approximation algorithm for the splittable CFLP to find an approximate splittable (i.e. fractional) assignment ϕ_S' for small clients. Suppose that the approximation algorithm used for the splittable CFLP has separate factors λ_{ss}, λ_{sf}, λ_{fs}, λ_{ff} such that it returns an assignment with service cost at most $\lambda_{ss}c_s(\tilde{\phi}_S) + \lambda_{sf}c_f(\tilde{\phi}_S)$ and with opening cost $\lambda_{fs}c_s(\tilde{\phi}_S) + \lambda_{ff}c_f(\tilde{\phi}_S)$ for any feasible solution $\tilde{\phi}_S$. Therefore, using Lemma 1:

$$c_s(\phi_S') \leq \lambda_{ss}c_s(\phi_S'') + \lambda_{sf}c_f(\phi_S''), \tag{1}$$

and

$$c_f(\phi_S') \leq \lambda_{fs}c_s(\phi_S'') + \lambda_{ff}c_f(\phi_S''). \tag{2}$$

The current best approximation for the splittable CFLP is due to Zhang *et al.* [10] with parameters $\lambda_{ss} = 1$, $\lambda_{sf} = 1$, $\lambda_{fs} = 4$, and $\lambda_{ff} = 5$.

In Step 3, we round the splittable assignment ϕ_S' using the algorithm of Shmoys and Tardos [8] for the Generalized Assignment Problem (GAP) to find an integer assignment ϕ_S. The GAP is a scheduling problem which has similarities to the UCFLP. In the GAP, we have a collection of jobs J and a set M of machines. Each job must be assigned to exactly one machine in M. If job $j \in J$ is assigned to machine $i \in M$, then it requires p_{ij} units of processing and incurs a cost r_{ij}. Each machine $i \in M$ can be assigned jobs of total processing time at most P_i. We want to find an assignment of jobs to machines to minimize the total assignment cost. We should point out that r_{ij} values do not necessarily satisfy the triangle inequality. Shmoys and Tardos [8] show that a feasible fractional solution of the GAP can be rounded, in polynomial time, to an integer solution with the same cost that violates processing time limit P_i within additive factor $\max_{j \in J} p_{ij}$; in worst case this can be a factor 2. We can model (view) the unsplittable capacitated facility location problem as an instance of the GAP in the following sense: jobs are clients, machines are facilities, $p_{ij} = d_j$

for all i, $r_{ij} = d_j \cdot c_{ij}$ for all i and j, and $P_i = 1$, and all facilities are already open (machines are available). Therefore, if we have a fractional assignment of clients to facilities (i.e. a splittable assignment), ϕ'_S, then using the rounding algorithm of [8], we can round ϕ'_S to ϕ_S without increasing the connection cost, i.e. $c_s(\phi_S) = c_s(\phi'_S)$, such that the capacity constraints are violated by at most an additive factor of $\max_{j \in S} d_j$. Since all the jobs in S have demand at most ϵ, the capacity constraints are violated by at most a factor of $1 + \epsilon$.

After combining ϕ_S and ϕ_L in Step 4, the violation of capacities is within a factor of at most $\max\{\beta(\epsilon), (1+\epsilon)\}$, because the facilities with violated capacities in Step 1 will be removed in Step 2 and will not be used in Step 3. So it only remains to bound the cost of this assignment:

$$\begin{aligned}
c_s(\phi_S) &= c_s(\phi'_S) && \text{by rounding of [8]}\\
&\le \lambda_{ss} c_s(\phi''_S) + \lambda_{sf} c_f(\phi''_s) && \text{by Equation (1)}\\
&\le \lambda_{ss}(c_s(\phi^*) + c_s(\phi_L)) + \lambda_{sf} c_f(\phi^*), && \text{by Lemma 1}\\
c_f(\phi_S) &\le (1+\epsilon) c_f(\phi'_S) && \text{by rounding of [8]}\\
&\le (1+\epsilon)\lambda_{fs} c_s(\phi''_S) + (1+\epsilon)\lambda_{ff} c_f(\phi''_S) && \text{by Equation (2)}\\
&\le (1+\epsilon)\lambda_{fs}(c_s(\phi^*) + c_s(\phi_L)) + (1+\epsilon)\lambda_{ff} c_f(\phi^*). && \text{by Lemma 1}
\end{aligned}$$

Therefore:

$$\begin{aligned}
c(\phi) &= c(\phi_S) + c(\phi_L)\\
&= c_s(\phi_S) + c_f(\phi_S) + c_s(\phi_L) + c_f(\phi_L)\\
&\le h_1(\epsilon) c_s(\phi^*) + h_2(\epsilon) c_f(\phi^*) + (h_1(\epsilon)+1) c_s(\phi_L) + c_f(\phi_L), \qquad (3)
\end{aligned}$$

where $h_1(\epsilon) = \lambda_{ss} + (1+\epsilon)\lambda_{fs}$ and $h_2(\epsilon) = \lambda_{sf} + (1+\epsilon)\lambda_{ff}$. Since $h_1(\epsilon) \ge 0$ for any $\epsilon > 0$: $(h_1(\epsilon)+1)c_s(\phi_L)+c_f(\phi_L) \le (h_1(\epsilon)+1)c(\phi_L) \le \alpha(\epsilon)(h_1(\epsilon)+1)c(\phi^*_L) \le \alpha(\epsilon)(h_1(\epsilon)+1)c(\phi^*)$. Combining this with Inequality (3), we obtain that the cost of ϕ is within factor:

$$g(\epsilon, \alpha(\epsilon)) = \max(h_1(\epsilon), h_2(\epsilon)) + \alpha(\epsilon)(h_1(\epsilon) + 1) \qquad (4)$$

of the optimum. □

3 The RUCFLP($\frac{1}{2}$) and RUCFLP($\frac{1}{3}$)

In this section, we give two approximation algorithms for the RUCFLP($\frac{1}{2}$) and RUCFLP($\frac{1}{3}$). Combined with Theorem 1 (and using Algorithm 1) these imply two approximation algorithms for the UCFLP. We start with the simpler of the two, namely the RUCFLP($\frac{1}{2}$).

Theorem 5. *There is a polynomial time exact algorithm for the RUCFLP($\frac{1}{2}$).*

Proof. Consider an optimal solution for a given instance of this problem with value OPT_L. Because $d_j > \frac{1}{2}$ for all $j \in C$, each facility can serve at most one client in the optimal solution. Therefore, the optimal assignment function,

ϕ_L^*, induces a matching $M = \{j\phi_L^*(j) : j \in C\}$. Let $w_{ij} = c_{ij}.d_j + f_i$ and let $w(H) = \sum_{e \in H} w_e$ for any subset of edges $H \subseteq E$. It follows that $w(M) = \mathrm{OPT}_L$.

Let M^* be a minimum weight perfect matching with respect to weights w_{ij}. Clearly, $w(M^*) \leq w(M) = \mathrm{OPT}_L$. In addition, M^* induces a feasible assignment of clients to facilities with cost $w(M^*)$. Thus, M^* induces an optimal solution for the RUCFLP($\frac{1}{2}$). Since we can find a minimum weight perfect matching in polynomial time, there is an exact algorithm for the RUCFLP($\frac{1}{2}$). □

Corollary 2. *There is a polynomial time* $(16.5, 3/2)$*-approximation algorithm for the UCFL problem.*

Proof. We run Algorithm 1, where we use the algorithm of Theorem 5 in the first step. Substituting $\alpha(\epsilon) = 1$ and $\epsilon = 1/2$, we have $h_1(\frac{1}{2}) = 7$, $h_2(\frac{1}{2}) = 8.5$, and $g(\epsilon, \alpha(\epsilon)) = 16.5$. Since $\beta(\epsilon) = 1$, the overall ratio is $(16.5, 3/2)$. □

The algorithm for the RUCFLP($\frac{1}{3}$) is more involved. First, we show how finding an approximation algorithm for the RUCFLP(ϵ) with zero facility opening costs leads to an approximation algorithm for the general RUCFLP(ϵ). Then, we give an approximation algorithm for the RUCFLP($\frac{1}{3}$) with zero opening costs.

Lemma 2. *Given an algorithm* $\mathcal{A}'$ *for the RUCFLP(*ϵ*) with zero facility opening costs having approximation factor* $(\alpha'(\epsilon), \beta(\epsilon))$*, we can find a* $(\alpha'(\epsilon)\frac{1}{\epsilon}, \beta(\epsilon))$*-approximation algorithm* $\mathcal{A}$ *for the general RUCFLP(*ϵ*).*

Proof. Define a new connection cost $c'_{ij} = c_{ij} + f_i$ and opening cost $f'_i = 0$ for all $i \in F$ and $j \in C$. Note that the new cost function is still metric. Then, we run $\mathcal{A}'$ on this new modified instance and let the solution returned by it be assignment ϕ_L. We use ϕ_L to assign the clients for the original instance and we claim this is a $(\alpha'(\epsilon)\frac{1}{\epsilon}, \beta(\epsilon))$-approximation. The proof of this appears in the full version. □

Now, we present a $(7, 1)$-approximation algorithm for the RUCFLP($\frac{1}{3}$) with zero opening costs (see Algorithm 2), which coupled with Lemma 2 yields a $(21, 1)$-approximation algorithm for the RUCFLP($\frac{1}{3}$).

Theorem 6. *There is a* $(7, 1)$*-approximation algorithm for the RUCFLP(*$\frac{1}{3}$*) with zero opening costs.*

Proof. Note that all the clients in the given instance have size $> \frac{1}{3}$. We break them into two groups: $L' = \{j \in C : d_j > \frac{1}{2}\}$ and $L'' = C \backslash L'$ are those which have size in $(\frac{1}{3}, \frac{1}{2}]$. In this proof (and of Lemma 3), we call clients in L', *huge* clients and those in L'', *moderately-large* clients. The algorithm assigns the huge clients by running a minimum weight perfect matching algorithm with edge weights $w_{ij} = d_j c_{ij}$. Let $I_{L'}$ be the opened facilities and $\phi_{L'} : L' \to I_{L'}$ be the assignment function. For moderately-large clients (i.e. those in L''), we define a flow-network H and show that minimum cost maximum flows in H correspond to minimum cost feasible assignment of clients in L'' to facilities (given the assignment $\phi_{L'}$).

Directed network H has node set $X \cup Y \cup \{s, t\}$ where there is a node $x_j \in X$ for every client $j \in L''$ and a node $y_i \in Y$ for every facility $i \in F$; s is the

Algorithm 2 Algorithm for solving the RUCFLP($\frac{1}{3}$) with zero opening costs

Require: An instance of the RUCFLP($\frac{1}{3}$) with zero opening costs
Ensure: A subset $I \subseteq F$ and a function $\phi : C \to I$

1: Let $L' = \{j \in C : d_j > \frac{1}{2}\}$ and $L'' = C \backslash L'$. Assign the clients in L' by running a minimum weight maximum matching algorithm that saturates L' with edge weights $w_{ij} = d_j c_{ij}$. Let $I_{L'}$ be the opened facilities and $\phi_{L'} : L' \to I_{L'}$ be the assignment function.
2: Build the flow network H as described in Theorem 6.
3: Find a minimum cost maximum flow in H. If the value of the flow is smaller than $|L''|$ then return "Infeasible". Else, let $I_{L''}$ be the subset of facilities in $F \backslash I_{L'}$ whose corresponding nodes in Y (in H) have non-zero flow through them and $\phi_{L''}$ be the assignment function defined as: if there is a unit flow from x_j to y_i in H then $\phi_{L''}(j) = i$.
4: Let $I = I_{L''} \cup I_{L'}$. Combine $\phi_{L''}$ and $\phi_{L'}$ to form assignment function $\phi_L : C \to I$ where $\phi(j) = \phi_{L''}(j)$ if $j \in L''$, otherwise $\phi(j) = \phi_{L'}(j)$. Return ϕ and I.

source and t is the sink. The source is connected to each node $x_j \in X$, and all $y_i \in Y$ are connected to the sink. Each $x_j \in X$ is connected to a node $y_i \in Y$ if either: the corresponding facility i is in $F \backslash I_{L'}$, i.e. unopened yet, or i is in $I_{L'}$ and the remaining capacity of i is enough to serve the demand of client j. Set the capacity of the edges between the source and the nodes in X to 1, set the capacity of the edges between X and Y to 1, set the capacity of the edges between the nodes $y_i \in Y$ whose corresponding facility i is unopened (i.e. not in $I_{L'}$) and the sink to 2, and set the capacity of the edges between the nodes $y_i \in Y$ whose corresponding facility is in $I_{L'}$ and the sink to 1. The cost of an edge connecting $x_j y_i$ is $d_j \cdot c_{ij}$ and all the other costs are 0. Algorithm 2 summarizes the algorithm for the RUCFLP($\frac{1}{3}$) with zero opening costs.

Let ϕ_L^* be an optimal assignment for the given instance of the RUCFLP($\frac{1}{3}$) with cost OPT_L. We use the following lemma (whose proof appears in the full version):

Lemma 3. *There exists an assignment ϕ' of clients consistent with $\phi_{L'}$ with cost at most $7\mathrm{OPT}_L$ where OPT_L is the cost of an optimum assignment ϕ_L^* for the given instance of the RUCFLP($\frac{1}{3}$).*

Below we prove that in Steps 2 and 3 the algorithm finds the best possible feasible assignment of clients in L'' (given $\phi_{L'}$). Therefore, the cost of ϕ formed in Step 4 is at most $c(\phi')$ and hence, is at most $7\mathrm{OPT}_L$.

Since for any $j \in L''$: $\frac{1}{3} < d_j \leq \frac{1}{2}$, each unopened facility after Step 1 can serve any two clients of L'' (and no more than two). This fact is reflected in that we connect all the nodes in X (corresponding to moderately-large clients) to the nodes in Y corresponding to unopened facilities $F \backslash I_{L'}$ and we set the capacity of the edges between those nodes in Y and the sink to 2. In addition, each facility in $I_{L'}$ can serve at most one moderately-large client, because more than $\frac{1}{2}$ of its capacity is already used by a huge client; accordingly we set the capacity of the edges from those nodes in Y to the sink to 1 and we only connect to them

the nodes of X whose corresponding client can be served by them. Considering these two simple facts (proof in the full version):

Lemma 4. *The maximum flow in H has value $|L''|$ if and only if the given instance is feasible and there is a one to one correspondence between maximum flows in H and feasible assignment of moderately-large clients (i.e. in L'') given $\phi_{L'}$. Furthermore, a pair of corresponding maximum flow in H and assignment of clients of L'' to F have the same cost.*

Therefore, the assignment $\phi_{L''}$ obtained from a minimum cost maximum flow in H has the minimum cost among the assignments consistent with $\phi_{L'}$. This together with Lemma 3 implies that ϕ_B as defined has cost at most $7\mathrm{OPT}_L$. □

Combining Lemma 2 and Theorem 6:

Corollary 3. *There is a polynomial time $(21, 1)$-approximation algorithm for the RUCFLP($\frac{1}{3}$).*

Corollary 4. *There is a (161.667,4/3)-approximation algorithm for the UCFL problem.*

Proof. We run Algorithm 1, where we use the algorithm of Corollary 3 for $\mathcal{A}$. That is, we first run the $(7, 1)$-approximation algorithm of Theorem 6 as algorithm $\mathcal{A}'$ in Lemma 2 to obtain $\mathcal{A}$ with $\alpha(\epsilon) = 21$ and $\epsilon = 1/3$. Thus $h_1(\frac{1}{3}) = 19/3$, $h_2(\frac{1}{3}) = 23/3$, and $g(\epsilon, \alpha(\epsilon)) = 23/3 + 21(22/3) < 161.667$. Since $\beta(\epsilon) = 1$, the overall ratio is $(161.667, 4/3)$. □

With a more careful analysis and a simple scaling to balance the bi-factors of connection and facility costs, we can bring down the factors of our algorithms (see the full version). This will imply the improved ratios in Theorem 2 and 3.

Notice that we solved the RUCFLP($\frac{1}{3}$) and the the RUCFLP($\frac{1}{2}$) without violation of capacities, but this is not possible for smaller values of ϵ as shown below (see the full version for the proof).

Theorem 7. *The RUCFLP(ϵ) does not admit any $(\alpha(\epsilon), 1)$-approximation algorithm for $\epsilon < \frac{1}{3}$ unless $P = NP$.*

It should be noted that to find an algorithm for the UCFLP that violates capacities within factor $1 + \epsilon$, we do not need to find an algorithm that does not violate capacities in the RUCFLP(ϵ). Even if we violate the capacities within factor $1 + \epsilon$ in the RUCFLP(ϵ), by Theorem 1 we can get an algorithm for the UCFLP that violates the capacities within factor $1 + \epsilon$. We think it is possible to find an $(\alpha(\epsilon), 1 + \epsilon)$-approximation for the RUCFLP(ϵ) for any constant $\epsilon > 0$. This, together with Theorem 1 would imply an $(f(\epsilon), 1 + \epsilon)$-approximation for the UCFLP, for any constant $\epsilon > 0$.

4 Discussion

We presented a reduction from the UCFLP to a restricted version in which all demand values are large (i.e. $> \epsilon$) and presented two algorithms for the case of $\epsilon = \frac{1}{2}$ and $\frac{1}{3}$. These implied two constant factor approximation algorithms for the UCFLP with capacity bounds within factor $3/2$ and $4/3$. We believe similar results can be found with capacity violations bounded within factor $1+\epsilon$ for any $\epsilon > 0$. We also showed that at a loss of factor $1/\epsilon$, we can ignore the opening cost of facilities, and that if there is an $(\alpha(\epsilon), 1+\epsilon)$-approximation for these instances then there is an $(\alpha'(\epsilon), 1+\epsilon)$-approximation for the general case. We believe that it should be possible to design constant factor (perhaps depending on ϵ) approximation for RUCFLP(ϵ) with zero opening costs with a violation of at most $1+\epsilon$ on capacities.

Acknowledgements: Part of this work was done while the second author was on sabbatical at Toyota Tech. Inst. at Chicago (TTIC). He would like to thank TTIC for hosting him.

References

1. Aggarwal, A., Anand, L., Bansal, M., Garg, N., Gupta, N., Gupta, S., Jain, S.: A 3-approximation for facility location with uniform capacities. In: Integer Programming and Combinatorial Optimization. pp. 149–162. Lecture Notes in Computer Science (2010)
2. Arora, S.: Polynomial time approximation schemes for euclidean tsp and other geometric problems. In: Proceedings of the 37th Annual Symposium on Foundations of Computer Science. pp. 2–12 (1996)
3. Bateni, M., Hajiaghayi, M.: Assignment problem in content distribution networks: unsplittable hard-capacitated facility location. In: Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms. pp. 805–814 (2009)
4. Guha, S., Khuller, S.: Greedy strikes back: improved facility location algorithms. In: SODA '98: Proceedings of the ninth annual ACM-SIAM symposium on Discrete algorithms. pp. 649–657 (1998)
5. Korupolu, M.R., Plaxton, C.G., Rajaraman, R.: Analysis of a local search heuristic for facility location problems. In: Proceedings of the ninth annual ACM-SIAM symposium on Discrete algorithms. pp. 1–10. SODA '98 (1998)
6. Li, S.: A 1.488 approximation algorithm for the uncapacitated facility location problem. In: Proceedings of the 38th international conference on Automata, languages and programming - Volume Part II. pp. 77–88. ICALP'11 (2011)
7. Mahdian, M., Ye, Y., Zhang, J.: A 2-approximation algorithm for the soft-capacitated facility location problem. In: RANDOM-APPROX. pp. 129–140 (2003)
8. Shmoys, D., Tardos, E.: An approximation algorithm for the generalized assignment problem. Mathematical Programming 62(3), 461–474 (1993)
9. Shmoys, D., Tardos, E., Aardal, K.: Approximation algorithms for facility location problems. In: Proceedings of the twenty-ninth annual ACM symposium on theory of computing. pp. 265–274 (1997)
10. Zhang, J., Chen, B., Ye, Y.: A multi-exchange local search algorithm for the capacitated facility location problem. In: Integer Programming and Combinatorial Optimization. pp. 1–4 (2004)