CrossMark

Combinatorial approximation algorithms for the robust facility location problem with penalties

Fengmin Wang · Dachuan Xu · Chenchen Wu

Abstract In this paper, we consider the robust facility location problem with penalties, aiming to serve only a specified fraction of the clients. We formulate this problem as an integer linear program to identify which clients must be served. Based on the corresponding LP relaxation and dual program, we propose a primal–dual (combinatorial) 3-approximation algorithm. Combining the greedy augmentation procedure, we further improve the above approximation ratio to 2.

1 Introduction

The classical uncapacitated facility location problem (UFLP), first formulated in the early 60's, has received widespread attention in the operations research and computer science community [11,19]. It aims to open some facilities from the given location set to serve all the given clients, so that the sum of facility opening cost and serving cost is minimized. Since the UFLP is one of the classical NP-hard problems, recent works have mainly concentrated on designing approximation algorithms for it [5,9,14,16,17,21,26,30,32]. For a minimization combinatorial optimization problem (such as the UFLP), an ρ-approximation algorithm A is a polynomial-time algorithm which always outputs a feasible solution whose value is no more than ρ times the optimal value for each instance of the problem. Here ρ is called

F. Wang · D. Xu (✉)
Department of Information and Operations Research, Beijing University of Technology, 100 Pingleyuan, Chaoyang District, Beijing 100124, People's Republic of China
e-mail: xudc@bjut.edu.cn

C. Wu
College of Science, Tianjin University of Technology, Tianjin 300384, People's Republic of China

the approximation factor. We refer to [33,34] for detailed discussions about approximation algorithms. Among the existing approximation algorithms for the UFLP, the first constant factor is 3.16 proposed by Shmoys et al. [30]; and the currently best factor is 1.488 achieved by Li [21]. It is well-known that the lower bound for the UFLP is 1.463 [14].

Due to practical application, various variants of the UFLP are considered in the literatures [1–4,6–8,10,12,13,15,18,20,23–25,27–29,31,35–42]. These literature studies various variants of the UFLP using different techniques and they all design approximation algorithms for the problem considered. The contributions are improving the approximation ratio or putting forwards a new model of the problem. To model the case when there are a few very distant clients (named outliers) for the UFLP, Charikar et al. [6] proposed two variants of the UFLP, i.e., robust facility location problem (RFLP) and facility location problem with penalties (FLPWP). In the RFLP, given n clients and an integer parameter $q < n$, we need to make sure that at least $n - q$ clients are served while leaving out the rest which are called outliers. The objective is to minimize the sum of the opening cost and the connection cost. Charikar et al. [6] presented a primal–dual 3-approximation algorithm for the RFLP. In the FLPWP, each client has a penalty cost and we will provide service to some of the clients while penalizing the rest. The objective is to minimize the sum of the opening cost, the connection cost and the penalty cost. After the primal–dual 3-approximation algorithm given by Charikar et al. [6] for the FLPWP, Xu and Xu [37,38] presented an LP-rounding $(2 + 2/e)$-approximation algorithm; and then, combining the power of the primal–dual method and greedy augmentation techniques, they further provided an 1.8526-approximation algorithm. Li et al. [22] presented an LP-rounding 1.514-approximation algorithm which has the currently best ratio for the FLPWP.

Since the FLPWP does not consider the possibility of outliers, and the RFLP does not consider the possibility that not all the clients are required to be served and there are certain penalty cost for the clients, in this paper, we consider the robust facility location problem with penalties (RFLPWP) in which not all clients are required to be served. Given a parameter q, the RFLPWP aims to serve only a specified fraction of the clients, penalize some clients and ignore at most q outliers. The objective is to minimize the sum of the opening cost, the connection cost and the penalty cost. We extend the primal–dual method in [17] for the UFLP to a modified instance of the RFLPWP, similar to the one in [6], and obtain a 3-approximation algorithm for the RFLPWP. Combining the greedy augmentation procedure [5,14], we further improve the above approximation ratio to 2.

The rest of this paper is organized as follows. In Sect. 2, we present some preliminaries including the integer program, the linear programming relaxation and the dual program for the RFLPWP. In Sect. 3, we offer a primal–dual (combinatorial) 3-approximation algorithm. The improved algorithm and its analysis are given in Sect. 4. Finally, some discussions are given in Sect. 5.

2 Preliminaries

In the RFLPWP, given a facility set $\mathcal{F}$ and a client set $\mathcal{C}$, each client j has a penalty cost p_j. The opening cost of facility $i \in \mathcal{F}$ is f_i. The metric connection cost between client $j \in \mathcal{C}$ and facility $i \in \mathcal{F}$ is c_{ij}. We are also given q, the number of the outliers. Our objective is to determine an opening facility set $\hat{\mathcal{F}} \subseteq \mathcal{F}$, while selecting a penalized client set $\hat{\mathcal{P}} \subseteq \mathcal{C}$, an outlier set $\hat{\mathcal{O}} \subseteq \mathcal{C}(\hat{\mathcal{O}} = q)$, and then connect the clients in $\mathcal{C}\setminus(\hat{\mathcal{P}} \bigcup \hat{\mathcal{O}})$ to the opening
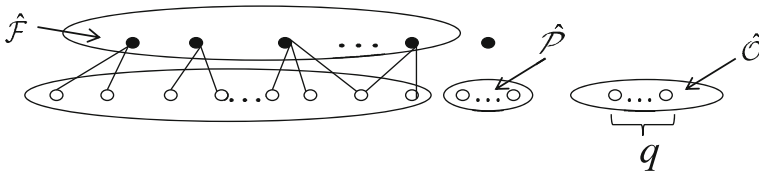
Fig. 1 The RFLPWP

facilities in $\hat{\mathcal{F}}$, such that the sum of the opening cost, the connection cost and the penalty cost is minimized. We illustrate the RFLPWP through Fig. 1.

We introduce four types of binary variables: y_i indicating whether facility i is opened; x_{ij} indicating whether client j is connected to facility i; z_j indicating whether client j is penalized; and r_j indicating whether client j is an extra outlier. The RFLPWP is formulated as

$$\begin{aligned}
\min \quad & \sum_{i \in \mathcal{F}} f_i y_i + \sum_{i \in \mathcal{F}} \sum_{j \in \mathcal{C}} c_{ij} x_{ij} + \sum_{j \in \mathcal{C}} p_j z_j \\
\text{s. t.} \quad & \sum_{i \in \mathcal{F}} x_{ij} + z_j + r_j \geq 1, \quad \forall j \in \mathcal{C}, \\
& x_{ij} \leq y_i, \quad \forall i \in \mathcal{F}, j \in \mathcal{C}, \\
& \sum_{j \in \mathcal{C}} r_j \leq q, \\
& x_{ij}, y_i, z_j, r_j \in \{0, 1\}, \quad \forall i \in \mathcal{F}, j \in \mathcal{C}.
\end{aligned} \qquad \text{(IP)}$$

In the above program, the first constraints denote that each client $j \in \mathcal{C}$ is connected to a facility or penalized or ignored as an outlier; the second constraints ensure that if client j is connected to facility i, then this facility must be opened; the third constraints indicate that there are at most q outliers.

Relaxing the last constraints, we obtain the LP relaxation.

$$\begin{aligned}
\min \quad & \sum_{i \in \mathcal{F}} f_i y_i + \sum_{i \in \mathcal{F}} \sum_{j \in \mathcal{C}} c_{ij} x_{ij} + \sum_{j \in \mathcal{C}} p_j z_j \\
\text{s. t.} \quad & \sum_{i \in \mathcal{F}} x_{ij} + z_j + r_j \geq 1, \quad \forall j \in \mathcal{C}, \\
& x_{ij} \leq y_i, \quad \forall i \in \mathcal{F}, j \in \mathcal{C}, \\
& \sum_{j \in \mathcal{C}} r_j \leq q, \\
& x_{ij}, y_i, z_j, r_j \geq 0, \quad \forall i \in \mathcal{F}, j \in \mathcal{C}.
\end{aligned} \qquad \text{(LP)}$$

Using the duality theory for linear programming, we can easily obtain the following dual of the program (LP)

$$\begin{aligned}
\max \quad & \sum_{j \in \mathcal{C}} \alpha_j - q\theta \\
\text{s. t.} \quad & \alpha_j \leq \beta_{ij} + c_{ij}, \quad \forall i \in \mathcal{F}, j \in \mathcal{C}, \\
& \sum_{j \in \mathcal{C}} \beta_{ij} \leq f_i, \quad \forall i \in \mathcal{F}, \\
& \alpha_j \leq p_j, \quad \forall j \in \mathcal{C}, \\
& \alpha_j \leq \theta, \quad \forall j \in \mathcal{C}, \\
& \alpha_j, \beta_{ij}, \theta \geq 0, \quad \forall i \in \mathcal{F}, j \in \mathcal{C},
\end{aligned} \qquad \text{(DP)}$$

where α_j can be viewed as the budget of client j, and β_{ij} as the contribution of client j to facility i.

3 A primal–dual 3-approximation algorithm

In this section, we will first propose a primal–dual algorithm for the RFLPWP, then analyze the algorithm to obtain the approximation ratio of 3. The main idea of primal–dual algorithm is to solve the problem combinatorially by first constructing a dual feasible solution, and then constructing a primal feasible integer solution.

3.1 The primal–dual algorithm

Algorithm 1 *(The primal–dual algorithm)*

Step 0. Constructing a new instance.
  Since there is an unbounded integrality gap for (LP), we guess the most expensive facility cost in the optimal solution, say $f_{\max}$. We set $f_{\max} := 0$ and the facility cost greater than $f_{\max}$ (the nonzero value in the original instance) to ∞. Let us denote this new instance as $\mathcal{I}^{(1)}$. For the instance $\mathcal{I}^{(1)}$, we run the following steps.
Step 1. *Let us introduce time t. The algorithm starts at time $t = 0$. Initially all the dual variables are zero, all the facilities are closed, and all the clients are unfrozen. In the process of the algorithm, client j becomes frozen when the dual variable α_j stops increasing. Let $\tilde{\mathcal{F}}$ denote the temporarily open facility set, U denote the unfrozen client set, $\tilde{\mathcal{P}}$ denote the temporarily penalized client set, and $\tilde{\mathcal{O}}$ denote the outlier set. For each $i \in \mathcal{F}$, denote N_i^{wit} to be the set of the clients whose connecting witness is facility i (we will explain the connecting witness at Step 2.2). At the beginning of the algorithm, set $\tilde{\mathcal{F}} := \emptyset$, $U := \mathcal{C}$, $\tilde{\mathcal{P}} := \emptyset$, $\tilde{\mathcal{O}} := \emptyset$, $N_i^{wit} := \emptyset$ for all $i \in \mathcal{F}$.*
Step 2. *Constructing a dual feasible solution (α, β, θ).*
  For the unfrozen client $j \in U$, we increase α_j at the same rate with time t.

Step 2.1 *If $|U| > q$ go to Step 2.2. Otherwise, freeze $j \in U$, let $\tilde{\mathcal{O}} := U$ and $U := \emptyset$. We denote this time by t_q. Let $\theta := t_q$. Go to Step 3.*
Step 2.2 *As time goes on, some of the constraints of (DP) will become tight, hence the following events will happen. If several events happen simultaneously, we execute the algorithm in arbitrary order.*
  Event 1. There is a client $j \in U$ and a facility $i \in \mathcal{F}$, such that $\alpha_j = c_{ij}$.
    Event 1.1 If the facility $i \in \tilde{\mathcal{F}}$, we say client j touches the facility $i \in \tilde{\mathcal{F}}$. Set $i(j) := i$ and call $i(j)$ the connecting witness of client j. Freeze j, and update $N_i^{wit} := N_i^{wit} \cup \{j\}$, $U := U \setminus \{j\}$.
    Event 1.2 If the facility $i \in \mathcal{F} \setminus \tilde{\mathcal{F}}$, we increase the corresponding dual variable β_{ij}.
  Event 2. There is a facility $i \in \mathcal{F} \setminus \tilde{\mathcal{F}}$, such that $\sum_{j \in \mathcal{C}} \beta_{ij} = f_i$. We say that facility i is fully paid, and it can be temporarily opened, record this time by $t(i)$. Update $\tilde{\mathcal{F}} := \tilde{\mathcal{F}} \cup \{i\}$, and define $N_i^{con} = \{j \in \mathcal{C} | \beta_{ij} > 0\}$ to be the neighbor of facility i, i.e., the set of the clients contributing to facility i. For each $j \in U \cap N_i^{con}$, set $i(j) := i$ and call $i(j)$ the connecting witness of client j. Freeze $j \in U \cap N_i^{con}$, and update $N_i^{wit} := N_i^{wit} \cup (U \cap N_i^{con})$, $U := U \setminus N_i^{con}$.
  Event 3. There is a client $j \in U$, such that $\alpha_j = p_j$. Freeze j, and update $\tilde{\mathcal{P}} := \tilde{\mathcal{P}} \cup \{j\}$ and $U := U \setminus \{j\}$.

Step 3. Constructing a primal integer feasible solution $(\hat{x}, \hat{y}, \hat{z}, \hat{r})$.

Let $\hat{\mathcal{F}}$ denote the finally open facility set, i.e., the facility set opened in the final integer solution, $\hat{\mathcal{P}}$ denote the penalty client set and $\hat{\mathcal{O}}$ denote the outlier set.

Step 3.1 Determine outliers. *If $|\tilde{\mathcal{O}}| = q$, set $\hat{\mathcal{O}} := \tilde{\mathcal{O}}$. Otherwise, there must be a facility i_q gets fully paid at time t_q (If not, Event 1 or Event 3 happens, this implies $|U| \geq q$). Choose the clients in $N_{i_q}^{wit}$ with the maximum $q - |\tilde{\mathcal{O}}|$ connection cost from i_q and add these clients to the set $\tilde{\mathcal{O}}$. Let us denote this set as $\hat{\mathcal{O}}$.*

Step 3.2 Determine open facilities. *Consider each facility $i \in \tilde{\mathcal{F}}$. If there is a facility $i' \in \tilde{\mathcal{F}}$, $i' \neq i$, such that $N_i^{con} \cap N_{i'}^{con} \neq \emptyset$, we say that facility i and i' are relevant to each other. We choose any maximal independent subset $\hat{\mathcal{F}} \subseteq \tilde{\mathcal{F}}$, open all facilities in $\hat{\mathcal{F}}$.*

Step 3.3 Determine penalty clients. *Set $\hat{\mathcal{P}} := \tilde{\mathcal{P}} \setminus \bigcup_{i \in \hat{\mathcal{F}}} N_i^{con}$.*

Step 3.4 Connect each client in $\mathcal{C}^{(1)} := \mathcal{C} \setminus (\hat{\mathcal{P}} \cup \hat{\mathcal{O}})$ to its closest open facility in $\hat{\mathcal{F}}$ respectively.

We declare that the dual solution obtained by Step 2 denoted by (α, β, θ) is feasible. First, the dual ascending process guarantees that the first three constraints in (DP) are established. Second, $\theta := t_q$ implies $\alpha_j \leq \theta$ for all clients. The feasibility of the solution $(\hat{x}, \hat{y}, \hat{z}, \hat{r})$ is clearly visible. Note that the new instance $\mathcal{I}^{(1)}$ just changes part of the facility cost. So $(\hat{x}, \hat{y}, \hat{z}, \hat{r})$ and (α, β, θ) are also feasible to the original instance.

Example of Algorithm 1

Given an instance $\mathcal{I}$ with a facility set $\mathcal{F} = \{i_1, i_2, i_3\}$ and a client set $\mathcal{C} = \{j_1, \ldots, j_n\}$, $n > 7$, see Fig. 2. Let ϵ be a small number. The penalty cost $p_{j_{n-5}} = 1 + \epsilon + \frac{\epsilon}{n-7}$, $p_{j_{n-4}} = 1$, all the other penalty cost are ∞. The opening cost $f_{i_1} = \epsilon$, $f_{i_2} = (n-5)\epsilon$, $f_{i_3} = (n-4)\epsilon$. The numbers on the solid line are the corresponding connection cost between the facilities and the clients, all the other connection cost satisfy tight triangle inequality. The number of the outliers $q = 3$.

We can see that the optimal solution of $\mathcal{I}$ is to open facility i_2 and i_3, connect clients as is shown in Fig. 3, penalize client j_{n-4}, choose clients j_{n-2}, j_{n-1}, j_n as outliers. The total optimal cost is $n - 2 + (2n - 9)\epsilon$.

It is easy to see that, by the Algorithm 1, set $f_{i_3} := 0$, as time goes on, the following events will happen. At time $t_1 = 1$, the clients $j_1, \ldots, j_{n-5}$ touch the facilities i_1 and i_2 respectively, and client j_{n-4} is penalized; at time $t_2 = 1 + \epsilon$, facility i_1 is fully paid; at

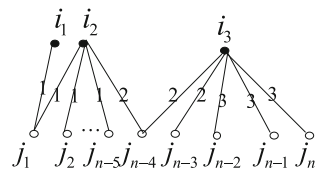**Fig. 2** Tight instance $\mathcal{I}$

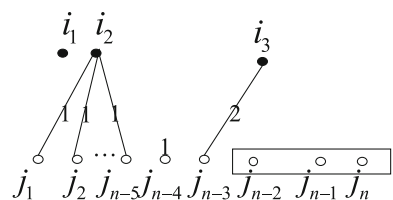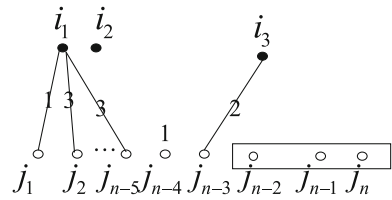**Fig. 3** The optimal solution of $\mathcal{I}$

Fig. 4 The solution of
algorithm 1

time $t_3 = 1 + \epsilon + \frac{\epsilon}{n-7}$, facility i_2 is fully paid, client j_{n-5} is penalized; at time $t_4 = 2$,
the client j_{n-3} touches the facility i_3. We then construct a feasible primal integer solution:
open facilities i_1, i_3, connect clients as is shown in Fig. 4, penalize client j_{n-4}, choose clients
j_{n-2}, j_{n-1}, j_n as outliers. The total cost is $3n - 14 + (n-3)\epsilon$.

The ratio between the result obtained by the Algorithm 1 and the optimal value is 3.

3.2 Analysis

In this subsection, we analyze the approximation factor of Algorithm 1, i.e., analyze the
relationship between the cost of the solution obtained from Algorithm 1 and the cost of
the optimal solution denoted by OPT. Denote $OPT^{(1)}$ as the optimal solution cost of the
instance $\mathcal{I}^{(1)}$. We have $OPT \geq f_{\max} + OPT^{(1)}$. At the same time we introduce $F^{(1)}, C^{(1)}$
and $P^{(1)}$, which indicate the opening cost, connection cost and penalty cost of the solution
$(\hat{x}, \hat{y}, \hat{z}, \hat{r})$ respectively. Furthermore, let $F_q^{(1)}$ denote the facility cost of $\hat{\mathcal{F}} \setminus \{i_q\}$.

In order to bound the total cost of the solution $(\hat{x}, \hat{y}, \hat{z}, \hat{r})$, we provide the following
lemmas to bound $F_q^{(1)}, C^{(1)}$ and $P^{(1)}$ by the cost of the dual solution respectively.

According to the construction of $\hat{\mathcal{F}}$ at Step 3.2 of Algorithm 1, we have the following
lemma.

Lemma 1

$$F_q^{(1)} \leq \sum_{i \in \hat{\mathcal{F}}} \sum_{j \in N_i^{con} \setminus (N_{i_q}^{con} \cap \hat{\mathcal{O}})} \beta_{ij}.$$

Proof Note that $F_q^{(1)} = \sum\limits_{i \in \hat{\mathcal{F}} \setminus \{i_q\}} f_i$ and $f_i = \sum\limits_{j \in N_i^{con}} \beta_{ij}$ for each $i \in \hat{\mathcal{F}}$.

It follows from the construction of $\hat{\mathcal{F}}$ at Step 3.2 of Algorithm 1 that

$$F_q^{(1)} = \sum_{i \in \hat{\mathcal{F}} \setminus \{i_q\}} \sum_{j \in N_i^{con}} \beta_{ij} \leq \sum_{i \in \hat{\mathcal{F}}} \sum_{j \in N_i^{con} \setminus (N_{i_q}^{con} \cap \hat{\mathcal{O}})} \beta_{ij}.$$

$\square$

For convenience, let us denote

$$\mathcal{C}^{con} := \bigcup_{i \in \hat{\mathcal{F}}} N_i^{con} \setminus (N_{i_q}^{con} \cap \hat{\mathcal{O}}),$$

$$\mathcal{C}^{tou} := \bigcup_{i \in \hat{\mathcal{F}}} (N_i^{wit} \setminus N_i^{con}),$$

$$\mathcal{C}^{clo} := \mathcal{C}^{(1)} \setminus \bigcup_{i \in \hat{\mathcal{F}}} (N_i^{wit} \cup N_i^{con}).$$

Fig. 5 The partition of clients
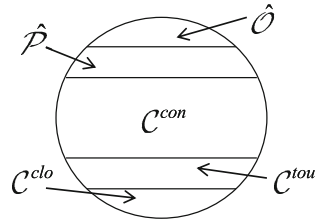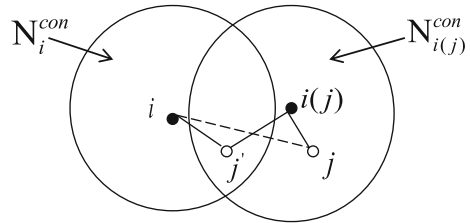set $\mathcal{C}$

Fig. 6 The evaluation of
connection cost for client
$j \in \mathcal{C}^{col}$

Note that $\mathcal{C} = \mathcal{C}^{con} \cup \mathcal{C}^{tou} \cup \mathcal{C}^{clo} \cup \hat{\mathcal{P}} \cup \hat{\mathcal{O}}$. The clients in $\mathcal{C}^{con}$ contribute to some finally open facilities, the clients in $\mathcal{C}^{tou}$ touch some finally open facilities, and the connecting witnesses of the clients in $\mathcal{C}^{clo}$ are closed by some finally open facilities. See Fig. 5.

We bound the connection cost in the following lemma.

Lemma 2

$$C^{(1)} \leq \sum_{i \in \hat{\mathcal{F}}} \sum_{j \in N_i^{con} \setminus (N_{i_q}^{con} \cap \hat{\mathcal{O}})} c_{ij} + \sum_{j \in \mathcal{C}^{tou}} \alpha_j + 3 \sum_{j \in \mathcal{C}^{clo}} \alpha_j.$$

Proof For any client $j \in \mathcal{C}^{(1)}, i \in \hat{\mathcal{F}}$, we connect clients in the following ways:

(a) Connect the client $j \in \mathcal{C}^{con}$ to the open facility to which it contributes.
(b) Connect the client $j \in \mathcal{C}^{tou}$ to the open facility to which it touches.
(c) Connect the client $j \in \mathcal{C}^{col}$ to the open facility which closes its connecting witness.

We can easily see that in the above assignment, the connection costs are $\sum_{i \in \hat{\mathcal{F}}} \sum_{j \in N_i^{con} \setminus (N_{i_q}^{con} \cap \hat{\mathcal{O}})} c_{ij}$
and $\sum_{j \in \mathcal{C}^{tou}} c_{ij} = \sum_{j \in \mathcal{C}^{tou}} \alpha_j$ for the clients in $\mathcal{C}^{con}$ and $\mathcal{C}^{tou}$ respectively.

For client $j \in \mathcal{C}^{col}$, since its connecting witness $i(j)$ is closed at Step 3.2 in Algorithm 1, there must exist a finally open facility $i \in \hat{\mathcal{F}}$ and a client j', such that $j' \in N_{i(j)}^{con} \cap N_i^{con}$. According to the above placement, client j is assigned to i (see Fig. 6).

Recall that $t(i(j))$ and $t(i)$ are the temporarily open time of facility $i(j)$ and i at Step 2 respectively. Obviously, $\alpha_j \geq t(i(j))$. And since $j' \in N_{i(j)}^{con} \cap N_i^{con}$, we have $c_{ij'} \leq \alpha_{j'}, c_{i(j)j'} \leq \alpha_{j'}$ and $\alpha_{j'} \leq \min\{t(i(j)), t(i)\} \leq t(i(j))$. Combined with the triangle inequality, we get

$$c_{ij} \leq c_{ij'} + c_{i(j)j'} + c_{i(j)j} \leq 2\alpha_{j'} + \alpha_j \leq 2t(i(j)) + \alpha_j \leq 3\alpha_j.$$

$\square$

According to Step 2.3 and Step 3.3 of Algorithm 1, we obtain the following lemma.

Lemma 3

$$P^{(1)} = \sum_{j \in \hat{\mathcal{P}}} \alpha_j.$$

Now we are ready to give the main result of this subsection.

Theorem 1 *Algorithm 1 is a 3-approximation algorithm for the RFLPWP.*

Proof The solution $(\hat{x}, \hat{y}, \hat{z}, \hat{r})$ is feasible in the original instance, and the corresponding opening cost, connection cost and penalty cost are $f_{\max} + F^{(1)}$, $C^{(1)}$ and $P^{(1)}$ respectively. By the definition of $F_q^{(1)}$ and $f_{\max}$, we have $f_{\max} + F^{(1)} \leq f_{\max} + f_{i_q} + F_q^{(1)} \leq 2 f_{\max} + F_q^{(1)}$. By Lemmas 1–3, we obtain

$$3F_q^{(1)} + C^{(1)} + 3P^{(1)} \leq 3 \left(\sum_{i \in \hat{\mathcal{F}}} \sum_{j \in N_i^{con} \setminus (N_{i_q}^{con} \cap \hat{\mathcal{O}})} \beta_{ij} + \sum_{i \in \hat{\mathcal{F}}} \sum_{j \in N_i^{con} \setminus (N_{i_q}^{con} \cap \hat{\mathcal{O}})} c_{ij} \right)$$

$$+ \sum_{j \in \mathcal{C}^{tou}} \alpha_j + 3 \sum_{j \in \mathcal{C}^{clo}} \alpha_j + 3 \sum_{j \in \hat{\mathcal{P}}} \alpha_j$$

$$= 3 \sum_{j \in \mathcal{C}^{con}} \alpha_j + \sum_{j \in \mathcal{C}^{tou}} \alpha_j + 3 \sum_{j \in \mathcal{C}^{clo}} \alpha_j + 3 \sum_{j \in \hat{\mathcal{P}}} \alpha_j$$

$$\leq 3 \sum_{j \in \mathcal{C} \setminus \hat{\mathcal{O}}} \alpha_j.$$

According to the definition of θ and the fact that the value of the dual solution is a lower bound on $OPT^{(1)}$, we get $\sum_{j \in \mathcal{C} \setminus \hat{\mathcal{O}}} \alpha_j = \sum_{j \in \mathcal{C}} \alpha_j - q\theta \leq OPT^{(1)}$. Thus, the total cost incurred by Algorithm 1 is no more than

$$f_{i_q} + f_{\max} + 3F_q^{(1)} + C^{(1)} + 3P^{(1)} \leq 2 f_{\max} + 3 OPT^{(1)} \leq 3 OPT.$$

$\square$

4 Improved 2-approximation algorithm

In this section, we propose an improved 2-approximation algorithm for the RFLPWP by combining with the greedy augmentation technique in [5,14].

4.1 The improved algorithm

Algorithm 2 *(The improved algorithm)*

Step 0. Given a positive constant δ. For any given instance $\mathcal{I}^{(1)}$, set $f_i' := \delta f_i$, for all $i \in \mathcal{F}$. Denote by $\mathcal{I}^{(2)}$ the scaled instance.

Step 1. Run Algorithm 1 on $\mathcal{I}^{(2)}$. Denote by $(\hat{x}', \hat{y}', \hat{z}', \hat{r}')$ the corresponding feasible solution. Obviously this solution is also feasible for $\mathcal{I}^{(1)}$. Let $\hat{\mathcal{F}}'$, $\hat{\mathcal{P}}'$ and $\hat{\mathcal{O}}'$ be the opening facility set, penalty clients set and outliers set of the solution $(\hat{x}', \hat{y}', \hat{z}', \hat{r}')$ for $\mathcal{I}^{(2)}$, respectively. Furthermore, let $\hat{\mathcal{F}}_g := \hat{\mathcal{F}}'$, $\hat{\mathcal{P}}_g := \hat{\mathcal{P}}'$, and $\hat{\mathcal{O}}_g := \hat{\mathcal{O}}'$ be the corresponding facility set, penalty clients set and outliers set for $\mathcal{I}^{(1)}$, respectively.
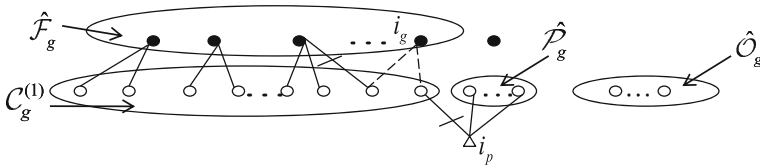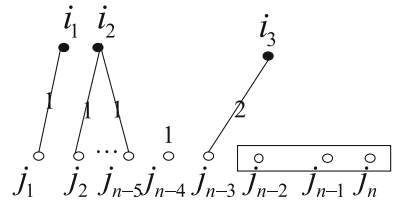
Fig. 7 The greedy improvement (Algorithm 2)

Fig. 8 The solution of Algorithm 2

Step 2. Greedy improvement.

Step 2.0 *Consider $\mathcal{I}^{(1)}$. For the clients in $\mathcal{C} \setminus \hat{\mathcal{O}}_g$, we add a dummy facility i_p to the feasible solution $(\hat{x}', \hat{y}', \hat{z}', \hat{r}')$ with opening cost $f_{i_p} = 0$ and connection cost $c_{i_p j} = p_j$, $\forall j \in \mathcal{C} \setminus \hat{\mathcal{O}}_g$. For each client $j \in \mathcal{C} \setminus \hat{\mathcal{O}}_g$, let $\pi(j)$ be the closest facility in $\hat{\mathcal{F}}'$. Note that for each client $j \in \hat{\mathcal{P}}_g$, $\pi(j) = i_p$. Let $\mathcal{F} := \mathcal{F} \cup \{i_p\}$, and $\hat{\mathcal{F}}_g := \hat{\mathcal{F}}_g \cup \{i_p\}$.*

Step 2.1 *Greedily find the facility*

$$i_g := \arg \max_{i \in \mathcal{F} \setminus \hat{\mathcal{F}}_g} \left\{ \frac{\text{gain}(i)}{f_i} \right\},$$

where

$$\text{gain}(i) := \sum_{j \in \mathcal{C}: c_{\pi(j)j} \geq c_{ij}} (c_{\pi(j)j} - c_{ij}) - f_i, \quad \forall i \in \mathcal{F} \setminus \hat{\mathcal{F}}_g.$$

Step 2.2 *If $\text{gain}(i_g) > 0$, update $\hat{\mathcal{F}}_g := \hat{\mathcal{F}}_g \cup \{i_g\}$, and $\hat{\mathcal{P}}_g := \hat{\mathcal{P}}_g \setminus \{j | c_{\pi(j)j} \geq c_{i_g j}\}$, go to Step 2.1; otherwise, go to Step 3.*

Step 3. *Set $\mathcal{C}_g^{(1)} := \mathcal{C} \setminus (\hat{\mathcal{P}}_g \cup \hat{\mathcal{O}}_g)$. Connect $j \in \mathcal{C}_g^{(1)}$ to its closest facility $i \in \hat{\mathcal{F}}_g$. Denote by $(\hat{x}_g, \hat{y}_g, \hat{z}_g, \hat{r}_g)$ the corresponding solution (cf. Fig. 7).*

Example of Algorithm 2

Consider the instance $\mathcal{I}$ in Sect. 3.2. Run Algorithm 2. The critical processes are as follows:

(1) Set $f_i' := \delta f_i$, for all $i \in \mathcal{F}$;
(2) Run Algorithm 1 on the scaled instance, we obtain a solution as is shown in Fig. 4;
(3) For the greedy improvement, since $\text{gain}(i_2) > 0$, we add to open facility i_2, and obtain a solution as is shown in Fig. 8, whose total cost is $n - 2 + (2n - 8)\epsilon$.

4.2 Analysis of algorithm 2

Let F, C, P be the corresponding opening cost, connection cost and penalty cost of solution $(\hat{x}', \hat{y}', \hat{z}', \hat{r}')$ for $\mathcal{I}^{(1)}$, and F^*, C^*, P^* be the optimal opening cost, connection cost and penalty cost for $\mathcal{I}^{(1)}$, respectively. Denote by $\mathcal{F}^*$ the opening facility set of the optimal solution $OPT^{(1)}$. Without loss of generality, we assume that $F^* > 0$.

Lemma 4 *If Step 2 of Algorithm 2 starts from a solution $(\hat{x}', \hat{y}', \hat{z}', \hat{r}')$ with $C + P > C^* + P^* + F^*$, then there is at least one facility i with* gain$(i) > 0$.

Proof For each client $j \in \mathcal{C} \setminus \hat{\mathcal{O}}$, add a facility i_0 to $\mathcal{F}^*$ with opening cost $f_{i_0} = 0$ and connection cost $c_{i_0 j} = p_j$. Let $\mathcal{F}^* := \mathcal{F}^* \cup \{i_0\}$, $\pi^*(j)$ be the closest facility in $\mathcal{F}^*$ to client j.

Consider the iteration in Step 2 of Algorithm 2. We add facility i to $\hat{\mathcal{F}}_g$ if and only if $\pi^*(j) = i$, and connect client j to facility i. Define

$$\text{gain}'(i) := \sum_{j \in \mathcal{C}: \pi^*(j) = i} (c_{\pi(j)j} - c_{ij}) - f_i.$$

It is easy to verify that

$$\sum_{i \in \mathcal{F}^*} \text{gain}(i) \geq \sum_{i \in \mathcal{F}^*} \text{gain}'(i) = C + P - F^* - C^* - P^* > 0,$$

which implies that there is at least one facility i with gain$(i) > 0$.   □

Recall that Step 2 of Algorithm 2 is an iterative process. Denote by C_l, P_l, F_l the corresponding opening cost, connection cost and penalty cost, respectively, after finishing the lth iteration.

Lemma 5 *There exists a facility $\tilde{i} \in \mathcal{F}$ such that*

$$\frac{\text{gain}(\tilde{i})}{f_{\tilde{i}}} \geq \frac{C_{l-1} + P_{l-1} - F^* - C^* - P^*}{F^*}. \tag{1}$$

in the lth iteration.

Proof

$$\max_{i \in \mathcal{F}^*} \left\{ \frac{\text{gain}(i)}{f_i} \right\} F^* = \max_{i \in \mathcal{F}^*} \left\{ \frac{\text{gain}(i)}{f_i} \right\} \sum_{i \in F^*} f_i$$

$$\geq \sum_{i \in F^*} \text{gain}(i).$$

Similar to the proof of Lemma 4, we have

$$\sum_{i \in \mathcal{F}^*} \text{gain}(i) \geq \sum_{i \in \mathcal{F}^*} \text{gain}'(i) = C_{l-1} + P_{l-1} - F^* - C^* - P^*,$$

which indicates (1).   □

Let F^g, C^g, P^g be the corresponding opening cost, connection cost and penalty cost of solution $(\hat{x}_g, \hat{y}_g, \hat{z}_g, \hat{r}_g)$ for $\mathcal{I}^{(1)}$, respectively. Furthermore, let F_q and F_q^g be the facility cost of $\hat{\mathcal{F}}' \setminus \{i_q'\}$ and $\hat{\mathcal{F}}_g \setminus \{i_q'\}$ for $\mathcal{I}^{(1)}$, respectively; let F_q' be the facility cost of $\hat{\mathcal{F}}' \setminus \{i_q'\}$ for $\mathcal{I}^{(2)}$, where i_q' is the fully paid facility when the unfrozen clients number is less than q in Step 1 of Algorithm 2. Let F', C', P' be the corresponding opening cost, connection cost and penalty cost of solution $(\hat{x}', \hat{y}', \hat{z}', \hat{r}')$ for $\mathcal{I}^{(2)}$, and F'^*, C'^*, P'^* be the corresponding opening cost, connection cost and penalty cost of the optimal solution for $\mathcal{I}^{(2)}$, respectively.

Lemma 6

$$F_q^g + C^g + P^g \leq \max \left\{ 1 + \ln(3\delta), 2 - \frac{1}{3\delta}, 1 + \frac{2}{3\delta} \right\} OPT^{(1)}.$$

Proof Similar to the proof of Theorem 1, one can prove that

$$3F_q' + C' + 3P' \leq 3(F'^* + C'^* + P'^*).$$

Thus,

$$3\delta F_q + C + P \leq 3(\delta F^* + C^* + P^*). \tag{2}$$

Now we consider the following two cases.

Case 1. $C + P \leq F^* + C^* + P^*$.

Then we have

$$
\begin{aligned}
F_q + C + P &= \frac{3\delta F_q + C + P}{3\delta} + \left(1 - \frac{1}{3\delta}\right)(C + P) \\
&\leq \frac{3(\delta F^* + C^* + P^*)}{3\delta} + \left(1 - \frac{1}{3\delta}\right)(F^* + C^* + P^*) \\
&= \left(2 - \frac{1}{3\delta}\right)F^* + \left(1 + \frac{2}{3\delta}\right)(C^* + P^*).
\end{aligned}
$$

Then after Step 2 of Algorithm 2, we still have

$$F_q^g + C^g + P^g \leq \left(2 - \frac{1}{3\delta}\right)F^* + \left(1 + \frac{2}{3\delta}\right)(C^* + P^*). \tag{3}$$

Case 2. $C + P > F^* + C^* + P^*$.

By Lemma 4, we have $C_l + P_l \leq F^* + C^* + P^*$ when Algorithm 2 stops. Let k be the smallest integer such that $C_k + P_k \leq F^* + C^* + P^*$. By the definition of $gain(i)$ and (1), we have

$$\frac{C_{l-1} + P_{l-1} - C_l - P_l - F_l + F_{l-1}}{F_l - F_{l-1}} \geq \frac{C_{l-1} + P_{l-1} - F^* - C^* - P^*}{F^*},$$

for all $1 \leq l \leq k$. Reformulating the above relation, we have

$$F_l - F_{l-1} \leq F^* \left(\frac{C_{l-1} + P_{l-1} - C_l - P_l}{C_{l-1} + P_{l-1} - C^* - P^*}\right),$$

which indicates

$$F_k \leq F + F^* \sum_{l=1}^{k} \left(\frac{C_{l-1} + P_{l-1} - C_l - P_l}{C_{l-1} + P_{l-1} - C^* - P^*}\right).$$

From the above inequality, we have

$$
\begin{aligned}
F^g + C^g + P^g &\leq F_k + C_k + P_k \\
&\leq F + F^* \sum_{l=1}^{k} \left(\frac{C_{l-1} + P_{l-1} - C_l - P_l}{C_{l-1} + P_{l-1} - C^* - P^*}\right) + C_k + P_k. \tag{4}
\end{aligned}
$$

Treating the right hand side of (4) as a function of variables C_k and P_k, we can see that it achieves its maximum at $C_k + P_k = F^* + C^* + P^*$. So we can assume $C_k + P_k = F^* + C^* + P^*$ in the following proof. Noting that $1 - x \leq \ln\frac{1}{x}$ for any $x > 0$, and

$$\frac{C_l + P_l - C^* - P^*}{C_{l-1} + P_{l-1} - C^* - P^*} > 0,$$

we have

$$\frac{C_{l-1} + P_{l-1} - C_l - P_l}{C_{l-1} + P_{l-1} - C^* - P^*} = 1 - \frac{C_l + P_l - C^* - P^*}{C_{l-1} + P_{l-1} - C^* - P^*}$$

$$\leq \ln\left(\frac{C_l + P_l - C^* - P^*}{C_{l-1} + P_{l-1} - C^* - P^*}\right). \tag{5}$$

From (4) and (5), we have

$$F^g + C^g + P^g \leq F + F^* \sum_{l=1}^{k} \ln\left(\frac{C_l + P_l - C^* - P^*}{C_{l-1} + P_{l-1} - C^* - P^*}\right) + C_k + P_k$$

$$= F + F^* \ln\left(\frac{C + P - C^* - P^*}{C_k + P_k - C^* - P^*}\right) + C_k + P_k$$

$$= F + F^* \ln\left(\frac{C + P - C^* - P^*}{F^*}\right) + F^* + C^* + P^*.$$

If follows from the above inequality and the definitions of F_q^g and F_q that

$$F_q^g + C^g + P^g \leq F_q + F^* \ln\left(\frac{C + P - C^* - P^*}{F^*}\right) + F^* + C^* + P^*.$$

Combining the above inequality with (2), we obtain

$$F_q^g + C_g + P_g \leq F_q + F^* \ln\left(\frac{3\delta F^* - 3\delta F_q + 2C^* + 2P^*}{F^*}\right) + F^* + C^* + P^*.$$

One can view the right hand side of the above inequality as a function of F_q. Solving its maximum value, we get

$$F_q^g + C_g + P_g \leq (1 + \ln(3\delta))F^* + \left(1 + \frac{2}{3\delta}\right)(C^* + P^*). \tag{6}$$

Noting that $OPT^{(1)} = F^* + C^* + P^*$, we complete the proof by summarizing the above two cases together with (3) and (6). □

Now we are ready to give our improved approximation factor in the following theorem.

Theorem 2 *Algorithm 2 is a 2-approximation algorithm for the RFLPWP.*

Proof By the definition of $f_{\max}$ and f_{i_q}, combining with Lemma 6, and setting $\delta := 0.7192$, we have

$$f_{\max} + f_{i_q} + F_q^g + C^g + P^g \leq 2f_{\max} + 1.8526\,OPT^{(1)}$$

$$\leq 2(f_{\max} + OPT^{(1)})$$

$$\leq 2OPT,$$

which concludes the theorem. □

5 Discussions

In this paper, we propose a new model, the RFLPWP, which unifies both the FLPWP and the RFLP. We formulate it as an integer linear program and present the corresponding LP

relaxation. By exploring the structure of the dual program (cf. [17]), we propose a primal–dual (combinatorial) 3-approximation algorithm. Combining with the greedy augmentation technique in [5,14], we further improve the approximation ratio to 2.

There are serval interesting questions to study in the future. Recall that the best known approximation factor for the FLPWP and UFLP are 1.514 and 1.488 respectively. As a variation of the above two problems, it will be interesting to further improve the approximation factor of 2 for the RFLPWP. Since the UFLP can be viewed as a special set cover problem, it would be natural to consider the robust set cover problem with penalties.

References

1. Ageev, A., Ye, Y., Zhang, J.: Improved combinatorial approximation algorithms for the k-level facility location problem. SIAM J. Discrete Math. **18**, 207–217 (2003)
2. Aggarwal, A., Anand, L., Bansal, M., Garg, N., Gupta, N., Gupta, S., Jain, S.: A 3-approximation for facility location with uniform capacities. In: Proceedings of IPCO, pp. 149–162 (2010)
3. Bumb, A.: An approximation algorithm for the maximization version of the two level uncapacitated facility location problem. Oper. Res. Lett. **29**, 155–161 (2001)
4. Byrka, J., Srinivasan, A., Swamy, C.: Fault-tolerant facility location: a randomized dependent LP-rounding algorithm. In: Proceedings of IPCO, pp. 244–257 (2010)
5. Charikar, M., Guha, S.: Improved combinatorial algorithms for facility location problems. SIAM J. Comput. **34**, 803–824 (2005)
6. Charikar, M., Khuller, S., Mount, D.M., Naraasimban, G.: Algorithms for facility location problems with outliers. In: Proceedings of SODA, pp. 642–651 (2001)
7. Chechik, S., Peleg, D.: Robust fault-tolerant facility location. In: Proceedings of STACS, pp. 191–202 (2010)
8. Chen, X., Chen, B.: Approximation algorithms for soft-capacitated facility location in capacitated network design. Algorithmica **53**, 263–297 (2007)
9. Chudak, F.A., Shmoys, D.B.: Improved approximation algorithms for the uncapacitated facility location problem. SIAM J. Comput. **33**, 1–25 (2003)
10. Chudak, F.A., Williamson, D.P.: Improved approximation algorithms for capacitated facility location problems. Math. Program. **102**, 207–222 (2005)
11. Cornuejols, G., Nemhauser, G.L., Wolsey, L.A.: The uncapacitated facility location problem. In: Mirchandani, P.B., Francis, R.L. (eds.) Discrete Location Theory, pp. 119–171 (1990)
12. Du, D., Lu, R., Xu, D.: A primal–dual approximation algorithm for the facility location problem with submodular penalties. Algorithmica **63**, 191–200 (2012)
13. Grandoni, F., Rothvoβ, T.: Approximation algorithm for single and connected facility location. In: Proceedings of IPCO, pp. 248–260 (2011)
14. Guha, S., Khuller, S.: Greedy strikes back: improved facility location algorithms. J. Algorithm **31**, 228–248 (1999)
15. Hajiaghayi, M., Mahdian, M., Mirrokni, V.S.: The facility location problem with general cost functions. Networks **42**, 42–47 (2003)
16. Jain, K., Mahdian, M., Markakis, E., Saberi, A., Vazirani, V.V.: Greedy facility location algorithms analyzed using dual fitting with factor-revealing LP. J. ACM. **50**, 795–824 (2003)
17. Jain, K., Vazirani, V.V.: Approximation algorithms for metric facility location and k-median problems using the primal–dual schema and Lagrangian relaxation. J. ACM. **48**, 274–296 (2001)
18. Jung, H., Hasan, M.K., Chwa, K.-Y.: A 6.55 factor primal–dual approximation algorithm for the connected facility location problem. J. Comb. Optim. **18**, 258–271 (2009)
19. Kuehn, A.A., Hamburger, M.J.: A heuristic program for locating warehouses. Manag. Sci. **9**, 643–666 (1963)
20. Levi, R., Shmoys, D.B., Swamy, C.: LP-based approximation algorithms for capacitated facility location. Math. Program. **131**, 365–379 (2012)
21. Li, S.: A 1.488-approximation algorithm for the uncapacitated facility location problem. In: Proceedings of ICALP, Part II, pp. 77–88 (2011)

22. Li, Y., Du, D., Xiu, N., Xu, D.: Improved approximation algorithms for the facility location problems with linear/submodular penalty. In: Proceedings of COCOON, pp. 292–303 (2013)
23. Li, G., Li, Y., Shu, J., Xu, D.: A cross-monotonic cost sharing scheme for the concave facility location game. J. Global Optim. **56**, 1325–1335 (2013)
24. Li, Y., Xu, D., Du, D., Xiu, N.: Improved approximation algorithms for the robust fault-tolerant facility location problem. Inform. Process. Lett. **112**, 361–364 (2012)
25. Mahdian, M., Pál, M.: Universal facility location. In: Proceedings of ESA, pp. 409–421 (2003)
26. Mahdian, M., Ye, Y., Zhang, J.: Improved approximation algorithms for metric facility location problems. In: Proceedings of APPROX, pp. 229–242 (2002)
27. Mahdian, M., Ye, Y., Zhang, J.: A 2-approximation algorithm for the soft-capacitated facility location problem. In: Proceedings of APPROX, pp. 129–140 (2003)
28. Pál, M., Tardös, E., Wexler, T.: Facility location with hard capacities. In: Proceedings of FOCS, pp. 329–338 (2001)
29. Romeijn, H.E., Sharkey, T.C., Max Shen, Z.-J., Zhang, J.: Integrating facility location and production planning decisions. Networks **55**, 78–89 (2010)
30. Shmoys, D.B., Tardös, E., Aardal, K.I.: Approximation algorithms for facility location problems. In: Proceedings of STOC, pp. 265–274 (1997)
31. Shu, J.: An efficient greedy heuristic for warehouse-retailer network design optimization. Transp. Sci. **44**, 183–192 (2010)
32. Sviridenko, M.: An improved approximation algorithm for the metric uncapacitated facility location problem. In: Proceedings of IPCO, pp. 240–257 (2002)
33. Vazirani, V.V.: Approximation Algorithm. Springer, New York (2001)
34. Williamson, D.P., Shmoys, D.B.: The Design of Approximation Algorithms. Cambridge University Press, Cambridge (2011)
35. Xu, D.: A cross-monotonic cost sharing method for the facility location game with service installation costs. Sci. China Ser. A. **52**, 2530–2536 (2009)
36. Xu, D., Du, D.: The k-level facility location game. Oper. Res. Lett. **34**, 421–426 (2006)
37. Xu, G., Xu, J.: An LP rounding algorithm for approximating uncapacitated facility location problem with penalties. Inform. Process. Lett. **94**, 119–123 (2005)
38. Xu, G., Xu, J.: An improved approximation algorithm for uncapacitated facility location problem with penalties. J. Comb. Optim. **17**, 424–436 (2008)
39. Ye, Y., Zhang, J.: An approximation algorithm for the dynamic facility location problem. In: Combinatorial Optimization in Communication Networks, Kluwer Academic Publishers, pp. 623–637 (2005)
40. Zhang, J.: Approximating the two-level facility location problem via a quasi-greedy approach. Math. Program. **108**, 159–176 (2006)
41. Zhang, P.: A new approximation algorithm for the k-facility location problem. Theor. Comput. Sci. **384**, 126–135 (2007)
42. Zhang, J., Chen, B., Ye, Y.: A multiexchange local search algorithm for the capacitated facility location problem. Math. Oper. Res. **30**, 389–403 (2005)