

## LOCAL SEARCH ALGORITHM FOR THE SQUARED METRIC $k$ -FACILITY LOCATION PROBLEM WITH LINEAR PENALTIES

YISHUI WANG

Shenzhen Institutes of Advanced Technology  
Chinese Academy of Sciences, Shenzhen 518055, China

DONGMEI ZHANG

School of Computer Science and Technology,  
Shandong Jianzhu University, Jinan 250101, China

PENG ZHANG

School of Software, Shandong University,  
Jinan 250101, China

YONG ZHANG\*

Shenzhen Institutes of Advanced Technology  
Chinese Academy of Sciences, Shenzhen 518055, China

(Communicated by Wenxun Xing)

**ABSTRACT.** In the  $k$ -facility location problem, an important combinatorial optimization problem combining the classical facility location and  $k$ -median problems, we are given the locations of some facilities and clients, and need to open at most  $k$  facilities and connect all clients to opened facilities, minimizing the facility opening and connection cost. This paper considers the squared metric  $k$ -facility location problem with linear penalties, a robust version of the  $k$ -facility location problem. In this problem, we do not have to connect all clients to facilities, but each client that is not served by any facility must pay a penalty cost. The connection costs of facilities and clients are supposed to be squared metric, which is more general than the metric case. We provide a constant approximation algorithm based on the local search scheme with add, drop, and swap operations for this problem. Furthermore, we improve the approximation ratio by using the scaling technique.

**1. Introduction.** There are two classical location problems in the area of combinatorial optimization, the uncapacitated facility location problem and  $k$ -median problem. Since the 1960s, these two problems and their variants have been studied, and have wide applications in management science and computer science. In the uncapacitated facility location problem (FLP), a set of facilities and a set of clients are given. We need to open some facilities, connect every client to an opened facility, and pay the connection costs and the facility opening costs. The goal is to minimize the total costs. The  $k$ -median problem does not have the facility opening

---

2020 *Mathematics Subject Classification.* Primary: 90B80.

*Key words and phrases.* Approximation algorithm; local search;  $k$ -facility location problem; linear penalties; squared metric.

\* Corresponding author: Yong Zhang.

costs, but has the cardinality constraint that the number of opened facilities must be not greater than a given integer  $k$ .

Both the FLP and  $k$ -median problem are NP-hard even if the connection costs are metric, that is, the nonnegativity, symmetry and triangle inequalities are satisfied. So a lot of researches investigated in the approximation algorithm for the problems. For the non-metric FLP, Hochbaum [9] presented an  $O(\log n)$ -approximation algorithm where  $n$  is the number of clients, and proved that the approximation ratio is asymptotically tight. Most works focus on the metric case. The first constant approximation ratio for the metric FLP is 3.16, provided by Shmoys et al. [13]. After that, many constant approximation ratios for this problem were presented. The current best known approximation ratio is 1.488, given by Li [11]. For the metric  $k$ -median problem, the first constant approximation ratio is  $6\frac{2}{3}$ , presented by Charikar et al. [4]. After a series of improvements, Byrka et al. [2] gave the current best approximation ratio  $2.675 + \epsilon$ .

The  $k$ -facility location problem ( $k$ -FLP) combines the FLP and  $k$ -median problem, in which both the facility cost and cardinality constraint are considered. Jain and Vazirani [10] first introduced this problem and use primal-dual scheme to give a 6-approximation algorithm. Zhang [18] provided the current best known approximation ratio  $2 + \sqrt{3} + \epsilon$  with the local search technique.

An important variant of the classical facility location problem is to consider the penalty cost, that is, we need not to connect all clients to the facilities, but the clients that are not connected to any facility must pay some penalty costs. Linear penalties mean that each penalized client should pay a penalty cost independently, and the total penalty cost is the sum of them. Penalties can be applied to the cases that there exists some outliers in the set of clients, so it is more robust than the original model. Charikar et al. [5] first studied the metric FLP with linear penalties, and gave a 3-approximation ratio. Li et al. [12] improved the approximation ratio to 1.5148. Xu et al. [17] considered the universal FLP, an extension of the classical FLP, and gave a  $(7.88 + \epsilon)$ -approximation, which is the first local search algorithm for the penalty version of the facility location problem. The current best approximation ratios of the  $k$ -median and  $k$ -FLP with linear penalties are  $3 + \epsilon$  and  $2 + \sqrt{3} + \epsilon$ , provided by Li et al. [8] and Wang et al. [15], respectively.

The squared metric space is a distance space, in which the square roots of distances between two points in the space are metric, that is, the distances satisfy the nonnegativity, symmetry and squared triangle inequalities ( $\sqrt{c_{i_1 i_2}} \leq \sqrt{c_{i_1 i_3}} + \sqrt{c_{i_2 i_3}}$  for any points  $i_1, i_2$  and  $i_3$ ). Obviously, the squared metric space is more general than the metric space. There are some literatures that study the facility location problem on squared metric space. Fernandes et al. [7] showed that the LP-rounding algorithm provided by Chudak and Shmoys [6] for the metric FLP yields an approximation ratio 2.04011 for the squared metric FLP, and proved that this ratio is the best possible unless  $P = NP$ . Zhang et al. [19] considered the sum of squares facility location problem, in which the facilities and clients are located in Euclidean space, and the connection cost is the square of distance. This problem is equivalent to the squared Euclidean FLP which is a special case of squared metric FLP. They used the local search technique to present a  $(7.77 + \epsilon)$ -approximation for the continuous version of the problem, and a  $(9 + \epsilon)$ -approximation for the discrete version of the problem. Furthermore, other variants of FLP on squared metric space were studied, such as the capacitated FLP [16],  $k$ -FLP [20] and FLP with penalties [14].

In this paper, we consider the squared metric  $k$ -facility location problem with linear penalties (SM- $k$ -FLPLP), and apply the local search algorithm with the add, drop and swap operations to this problem. We prove that the local search algorithm has a polynomial-time complexity and a constant approximation ratio.

The rest of this paper is organized as follows. In Section 2.1, we give the formulation of the squared metric  $k$ -facility location problem and the local search algorithm. In Section 2.2, we give the analysis of the approximation ratio. In Section 2.3, we use the scaling technique to improve the approximation ratio. Finally, we conclude the paper in Section 3.

## 2. Local search for SM- $k$ -FLPLP.

**2.1. Formulation and algorithm.** Given the set of facilities  $\mathcal{F}$ , the set of clients  $\mathcal{D}$ , opening cost  $f_i$  for all  $i \in \mathcal{F}$ , penalty cost  $p_j$  for all  $j \in \mathcal{D}$ , and an integer  $k$ . We assume that facilities and clients are located on a squared metric space and the distance  $d(i, j)$  between any two points  $i, j \in \mathcal{F} \cup \mathcal{D}$  can be calculated in polynomial-time by a given oracle  $\Omega$ . Denote the instance of the SM- $k$ -FLPLP by  $\mathcal{I} = (\mathcal{F}, \mathcal{D}, k, f, p, \Omega)$ . The problem can be formulated as

$$\min_{F \subseteq \mathcal{F}: |F| \leq k} \sum_{i \in F} f_i + \sum_{j \in \mathcal{D}} \min \left\{ p_j, \min_{i \in F} d(i, j) \right\}.$$

Note that we can use a subset of  $\mathcal{F}$  to represent a feasible solution of the SM- $k$ -FLPLP from the above formulation. For the statement, we use  $C_f(F)$ ,  $C_s(F)$  and  $C_p(F)$  to denote the total facility opening cost, connection cost, and penalty cost of the solution  $F$ , respectively. Let  $\text{cost}(F) := C_f(F) + C_s(F) + C_p(F)$ . The local search algorithm (Algorithm 1) is shown below. It starts from a feasible solution of the problem, and finds a better solution in the neighborhood denoted by  $N(F)$  of the current solution  $F$  iteratively, until there is no better solution. The definition of  $N(F)$  is crucial to the performance of the local search algorithm. In this paper, we use the add, drop and swap operations which were used to the  $k$ -FLP in [18] to define  $N(F)$  as follows.

$$\begin{aligned} N(F) \quad := \quad & \{F \cup \{i\} \mid i \in \mathcal{F} \setminus F, |F \cup \{i\}| \leq k\} \cup \\ & \{F \setminus \{i\} \mid i \in F\} \cup \\ & \{F \setminus A \cup B \mid A \subseteq F, B \subseteq \mathcal{F} \setminus F, |A| = |B| \leq q\}, \end{aligned}$$

where  $q \in \mathbb{Z}^+$  is a parameter of the algorithm. The operations  $F \cup \{i\}$ ,  $F \setminus \{i\}$  and  $F \setminus A \cup B$  are denoted by  $\text{add}(i)$ ,  $\text{drop}(i)$  and  $\text{swap}(A, B)$  respectively.

---

### Algorithm 1 Local search algorithm for the SM- $k$ -FLPLP

---

**Input:** An instance  $\mathcal{I} = (\mathcal{F}, \mathcal{D}, k, f, p, \Omega)$  and a parameter  $q$ .

**Output:** A local optimal solution.

- 1: Let  $F$  be an arbitrary feasible solution of the instance  $\mathcal{I}$  for the SM- $k$ -FLPLP.
  - 2: **while** there exists a solution  $F'$  in  $N(F)$  such that  $\text{cost}(F') < \text{cost}(F)$  **do**
  - 3:   Update  $F = F'$ .
  - 4: **end while**
  - 5: **return**  $F$
- 

For a local search algorithm, the solution returned by the algorithm is called local optimal solution. The locality gap of the algorithm is defined as the worst

ratio (supermum for minimization and infimum for maximization) of the cost of the local optimal solution and the global optimal solution over all instances. We will analyze the locality gap of Algorithm 1 in the next subsection.

**2.2. Analysis.** The high-level of analyzing the locality gap of a local search algorithm is to construct several local operations to the local optimal solution, and then according to the local optimality, we can get several inequalities with respect to the local and global optimal solutions for these operations. Finally the locality gap is derived by combining these inequalities.

To construct the local operations, we need the notion of *capture* and partitions of the local optimal solution denoted by  $F$  and the global optimal solution denoted by  $F^*$ , which were used for the SM- $k$ -FLP in the reference [20]. Let  $s(o) := \operatorname{argmin}_{i \in F} d(i, o)$  for a facility  $o \in F^*$  (note: in this paper,  $\operatorname{argmin}$  takes arbitrary one element that has the minimal value). We say that the facility  $s \in S$  captures the facility  $o \in F^*$  if  $s = s(o)$ , and use  $\operatorname{cap}(s)$  to denote the set of all facilities that are captured by  $s$ . If a facility  $i \in S$  does not capture any facilities in  $F^*$ , then it is called a *good* facility, otherwise, it is called a *bad* facility. Note that each facility in  $F^*$  is captured by one and only one bad facility.

With loss of generality, we assume that  $F \cap F^* = \emptyset$  (if not, we only need to make a copy for each facility in  $F \cap F^*$ ). We introduce two dummy facilities  $u$  and  $u^*$ , and let  $d(u, j) = d(u^*, j) = p_j$  for all  $j \in \mathcal{D}$ . Penalizing a client in the solution  $F$  (or  $F^*$ ) is seen as connecting the client to the dummy facility  $u$  (or  $u^*$ ). Let  $s_j$  and  $o_j$  be the facilities (including the dummy facilities) which are connected to the client  $j$  in the solution  $F$  and  $F^*$  respectively,  $\mathcal{D}_s := \{j \in \mathcal{D} \mid s_j \neq u\}$ ,  $\mathcal{D}_p := \{j \in \mathcal{D} \mid s_j = u\}$ ,  $\mathcal{D}_s^* := \{j \in \mathcal{D} \mid o_j \neq u^*\}$ , and  $\mathcal{D}_p^* := \{j \in \mathcal{D} \mid o_j = u^*\}$ .

For simplicity, we let  $S_j := \min_{i \in F} d(i, j)$ ,  $S_j^* := \min_{i \in F^*} d(i, j)$  for all  $j \in \mathcal{D}$ ,  $C_f := C_f(F)$ ,  $C_s := C_s(F)$ ,  $C_p := C_p(F)$ ,  $C_f^* := C_f(F^*)$ ,  $C_s^* := C_s(F^*)$ , and  $C_p^* := C_p(F^*)$ . To bound the cost of  $F$ , we need the following three lemmas.

**Lemma 2.1.** (a):  $\sum_{j \in \mathcal{D}_s \cap \mathcal{D}_s^*} \sqrt{S_j S_j^*} \leq \sqrt{C_s C_s^*}$ .

(b):  $d(s(o_j), j) \leq S_j + 4S_j^* + 4\sqrt{S_j S_j^*}$ ,  $\forall j \in \mathcal{D}_s \cap \mathcal{D}_s^*$ .

*Proof.* It is easy to prove this lemma by extending the proof of Lemma 2 in Zhang et al. [20]. For the sake of completeness, we give the proof here. Using the Cauchy-Schwarz's inequality, we have

$$\sum_{j \in \mathcal{D}_s \cap \mathcal{D}_s^*} \sqrt{S_j S_j^*} \leq \sqrt{\sum_{j \in \mathcal{D}_s \cap \mathcal{D}_s^*} S_j} \sqrt{\sum_{j \in \mathcal{D}_s \cap \mathcal{D}_s^*} S_j^*} \leq \sqrt{C_s C_s^*}.$$

For any  $j \in \mathcal{D}_s \cap \mathcal{D}_s^*$ , we can use the squared triangle inequality for  $j$ ,  $o_j$ ,  $s_j$  and  $s(o_j)$ , since  $j$  is not penalized in both  $F$  and  $F^*$ . We have

$$\begin{aligned} d(s(o_j), j) &= \left( \sqrt{d(s(o_j), j)} \right)^2 \\ &\leq \left( \sqrt{d(s(o_j), o_j)} + \sqrt{d(o_j, j)} \right)^2 \\ &\leq \left( \sqrt{d(s_j, o_j)} + \sqrt{d(o_j, j)} \right)^2 \\ &\leq \left( \sqrt{d(s_j, j)} + 2\sqrt{d(o_j, j)} \right)^2 \end{aligned}$$

$$\begin{aligned}
&= \left( \sqrt{S_j} + 2\sqrt{S_j^*} \right)^2 \\
&= S_j + 4S_j^* + 4\sqrt{S_j S_j^*},
\end{aligned}$$

where the first and third inequalities follow from the squared triangle inequality, and the second inequality follows from the definition of  $s(\cdot)$ .  $\square$

**Lemma 2.2.** *For a bad facility  $\mathbf{a}$ , let  $\mathbf{b} = \operatorname{argmin}_{i \in \operatorname{cap}(\mathbf{a})} d(i, \mathbf{a})$ . If  $s(o_j) = \mathbf{a}$  for a client  $j \in N_F(\mathbf{a}) \setminus \mathcal{D}_p^*$ , where  $N_F(\mathbf{a})$  denotes the set of clients which are connected to the facility  $\mathbf{a}$  in the solution  $F$ , then we have*

$$d(\mathbf{b}, j) \leq 4S_j + S_j^* + 4\sqrt{S_j S_j^*}.$$

*Proof.* Since  $j \in N_F(\mathbf{a}) \setminus \mathcal{D}_p^*$ , we can use the squared triangle inequality for  $j$ ,  $o_j$ ,  $\mathbf{a}$  and  $\mathbf{b}$ . We have

$$\begin{aligned}
d(\mathbf{b}, j) &= \left( \sqrt{d(\mathbf{b}, j)} \right)^2 \\
&\leq \left( \sqrt{d(\mathbf{b}, \mathbf{a})} + \sqrt{d(\mathbf{a}, j)} \right)^2 \\
&\leq \left( \sqrt{d(o_j, \mathbf{a})} + \sqrt{d(\mathbf{a}, j)} \right)^2 \\
&\leq \left( \sqrt{d(o_j, j)} + 2\sqrt{d(\mathbf{a}, j)} \right)^2 \\
&= \left( \sqrt{S_j} + 2\sqrt{S_j^*} \right)^2 \\
&= 4S_j + S_j^* + 4\sqrt{S_j S_j^*},
\end{aligned}$$

where the first and third inequalities follow from the squared triangle inequality, the second inequality follows from  $\mathbf{b} = \operatorname{argmin}_{i \in \operatorname{cap}(\mathbf{a})} d(i, \mathbf{a})$  and  $s(o_j) = \mathbf{a}$ , and the second equality follows from  $j \in N_F(\mathbf{a}) \setminus \mathcal{D}_p^*$ .  $\square$

For the case that  $|F| > |F^*|$ , we partition the sets  $F$  and  $F^*$  as following procedure. We put all bad facilities  $e_1, e_2, \dots, e_r$  into  $r$  different sets  $A_1, A_2, \dots, A_r$ . Then, we let  $B_l := \operatorname{cap}(e_l)$  and supplement the set  $A_l$  by any good facilities until  $|A_l| = |B_l|$  for all  $l = 1, \dots, r$ . Finally, we put the remaining facilities in  $F$  into the set  $A_{r+1}$  (i. e.  $A_{r+1} := F \setminus \bigcup_{l=1}^r A_l$ ). After this procedure, we get  $F = \bigcup_{l=1}^{r+1} A_l$  and  $F^* = \bigcup_{l=1}^r B_l$ . With this partition, we construct swap and drop operations as follows (see FIGURE 1(a)).

- Considering the pair  $(A_l, B_l)$  for all  $l \in \{1, 2, \dots, r\}$ , suppose that  $A_l = \{a_1^l, a_2^l, \dots, a_{m_l}^l\}$ ,  $B_l = \{b_1^l, b_2^l, \dots, b_{m_l}^l\}$ , where  $a_1^l$  is the bad facility and  $b_1^l := \operatorname{argmin}_{i \in B_l} d(i, a_1^l)$ . We construct the operation  $\operatorname{swap}(a_i^l, b_i^l)$  for all  $l = 1, \dots, r$  and  $i = 1, \dots, m_l$ , to the local optimal solution  $F$ .
- For the set  $A_{r+1}$ , we construct the operation  $\operatorname{drop}(a)$  for all  $a \in A_{r+1}$ .

For the case that  $|F| \leq |F^*|$ , we construct the partition of  $F$  and  $F^*$  using the following procedure. We put all bad facilities  $e_1, e_2, \dots, e_r$  into  $r$  different sets  $A_1, A_2, \dots, A_r$ . Then, for all  $l = 1, \dots, r$ , we let  $B_l := \operatorname{cap}(e_l)$  and supplement the set  $A_l$  by any good facilities until  $|A_l| = |B_l|$  or there are no enough good facilities. Finally, for each  $B_l$  such that  $|B_l| > |A_l|$ , we take out arbitrarily some facilities from  $B_l$  subject to  $|B_l| = |A_l|$ , and put these facilities into a set denoted by  $B_{r+1}$ .

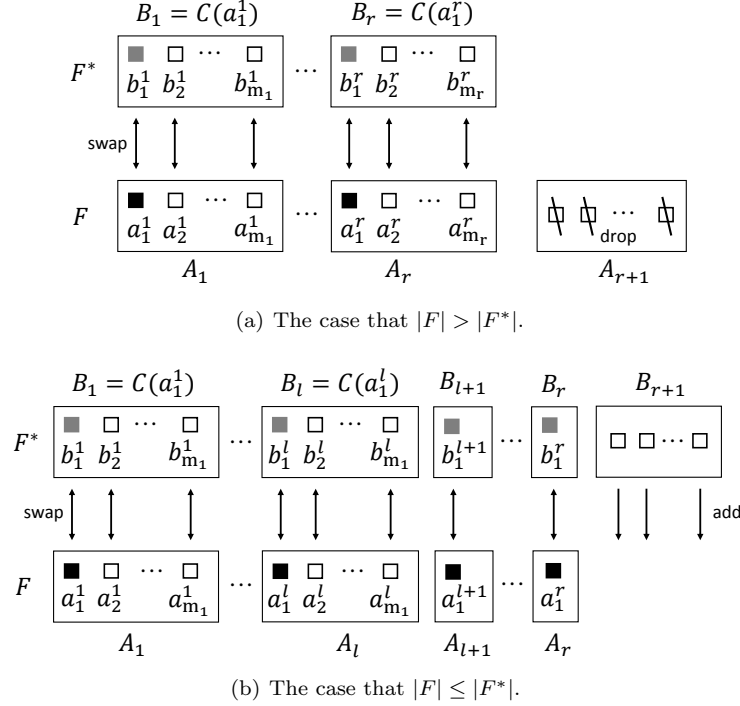


FIGURE 1. Partitions and local operations of  $F$  and  $F^*$  for analyzing the upper bound of  $C_f + C_p$ . The black solid squares represent all bad facilities. Each gray solid square represents the nearest facility among a part of  $F^*$  to the bad facility capturing them.

After this procedure, we get  $F = \dot{\bigcup}_{l=1}^r A_l$  and  $F^* = \dot{\bigcup}_{l=1}^{r+1} B_l$ . With this partition, we construct swap and add operations as follows (see FIGURE 1(b)).

- For the pairs  $\{(A_l, B_l)\}_{l=1, \dots, r}$ , we construct the same swap operations as in the case that  $|F| > |F^*|$ .
- For the set  $B_{r+1}$ , we construct the operation  $add(b)$  for all  $b \in B_{r+1}$ .

Using above local operations, together with the local optimality of  $F$ , we can get the following lemmas which show the upper bound of the penalty and facility opening cost of the local optimal solution  $F$ .

**Lemma 2.3.** *If  $|F| > |F^*|$ , we have*

$$C_f + C_p \leq C_f^* + 6C_s^* + 2C_s + 8\sqrt{C_s^* C_s} + C_p^*.$$

*Proof.* For each swap operation  $swap(a_i^l, b_i^l)$ , we only need to analyze the new cost of a client  $j \in N_F(a_i^l) \cup N_{F^*}(b_i^l)$ . There are some different cases as follows (see FIGURE 3).

**Case 1:**  $j \in N_F(a_i^l) \cap \mathcal{D}_p^*$ .

In this case, the new cost of  $j$  is at most  $p_j$  after the swap operation.

**Case 2:**  $j \in N_F(a_i^l) \cap N_{F^*}(b_i^l)$ .

**Case 2.1:**  $i = 1$ . We can reconnect the client  $j$  to the facility  $b_i^l$ , so the new cost of  $j$  is at most  $d(b_i^l, j) = S_j^*$ .

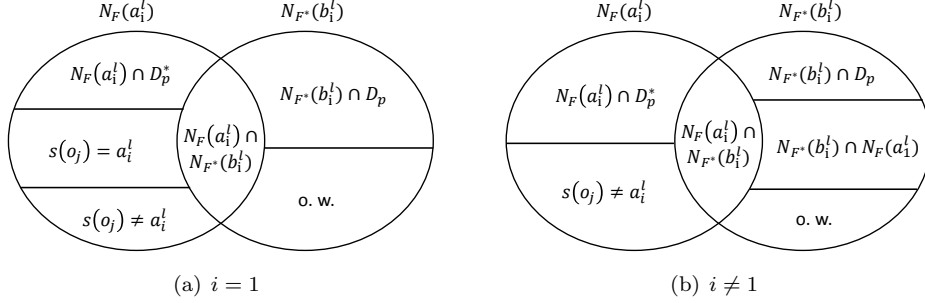


FIGURE 2. Different cases (partition of  $N_F(a_i^l) \cup N_{F^*}(b_i^l)$ ) for analyzing the new cost after the operation  $swap(a_i^l, b_i^l)$ .

**Case 2.2:**  $i \neq 1$ . In this case,  $s(o_j) \neq a_i^l$  because  $s(o_j)$  is a bad facility and  $a_i^l$  is a good facility. So  $s(o_j)$  is not swapped out, implying the new cost of  $j$  is at most  $\min\{S_j^*, d(s(o_j), j)\}$ .

**Case 3:**  $j \in N_F(a_i^l) \setminus (D_p^* \cup N_{F^*}(b_i^l))$  and  $s(o_j) \neq a_i^l$ .

Since  $s(o_j)$  is not swapped out, the new cost of  $j$  can be bounded by  $d(s(o_j), j)$ .

**Case 4:**  $j \in N_F(a_i^l) \setminus (D_p^* \cup N_{F^*}(b_i^l))$  and  $s(o_j) = a_i^l$ .

In this case, we must have  $i = 1$ . So the new cost of  $j$  can be bounded by  $d(b_1^l, j) \leq 4S_j + S_j^* + 4\sqrt{S_j S_j^*}$ , following from Lemma 2.2.

**Case 5:**  $j \in N_{F^*}(b_i^l) \cap N_F(a_1^l)$ .

**Case 5.1:**  $i = 1$ . It is the same to Case 2.1.

**Case 5.2:**  $i \neq 1$ . The new cost of  $j$  is at most  $d(b_i^l, j) = S_j^*$ . Note that we have  $s(o_j) = a_1$  in this case.

**Case 6:**  $j \in N_{F^*}(b_i^l) \cap D_p$ .

The new cost of  $j$  is  $\min\{p_j, d(b_i^l, j)\} = \min\{p_j, S_j^*\}$ .

**Case 7:**  $j \in N_{F^*}(b_i^l) \setminus (D_p \cup N_F(a_1^l) \cup N_F(a_i^l))$ .

The facility  $s_j$  is not swapped out, so the new cost is  $\min\{S_j, S_j^*\}$ .

Due to the local optimality of the solution  $F$ , we have

$$\begin{aligned}
0 &\leq \text{cost}(F \setminus \{a_i^l\} \cup \{b_i^l\}) - \text{cost}(F) \\
&\leq -f_{a_i^l} + f_{b_i^l} + \sum_{j \in N_F(a_i^l) \cap D_p^*} (p_j - S_j) + \sum_{j \in N_F(a_i^l) \cap N_{F^*}(b_i^l)} (S_j^* - S_j) \\
&\quad + \sum_{\substack{j \in N_F(a_i^l) \setminus (D_p^* \cup N_{F^*}(b_i^l)) \\ s(o_j) \neq a_i^l}} (d(s(o_j), j) - S_j) \\
&\quad + \sum_{\substack{j \in N_F(a_i^l) \setminus (D_p^* \cup N_{F^*}(b_i^l)) \\ s(o_j) = a_i^l}} (d(b_1^l, j) - S_j) + \sum_{j \in N_{F^*}(b_i^l) \cap D_p} (\min\{p_j, S_j^*\} - p_j) \\
&\quad + \sum_{j \in N_{F^*}(b_i^l) \setminus (D_p \cup N_F(a_1^l))} (\min\{S_j, S_j^*\} - S_j)
\end{aligned} \tag{1}$$

for  $i = 1$ , and

$$0 \leq \text{cost}(F \setminus \{a_i^l\} \cup \{b_i^l\}) - \text{cost}(F)$$

$$\begin{aligned}
\leq & -f_{a_i^l} + f_{b_i^l} + \sum_{j \in N_F(a_i^l) \cap \mathcal{D}_p^*} (p_j - S_j) + \sum_{j \in N_F(a_i^l) \cap N_{F^*}(b_i^l)} (d(s(o_j), j) - S_j) \\
& + \sum_{\substack{j \in N_F(a_i^l) \setminus (\mathcal{D}_p^* \cup N_{F^*}(b_i^l)) \\ s(o_j) \neq a_i^l}} (d(s(o_j), j) - S_j) \\
& + \sum_{j \in N_{F^*}(b_i^l) \cap N_F(a_1^l)} (S_j^* - S_j) + \sum_{j \in N_{F^*}(b_i^l) \cap \mathcal{D}_p} (\min\{p_j, S_j^*\} - p_j) \\
& + \sum_{j \in N_{F^*}(b_i^l) \setminus (\mathcal{D}_p \cup N_F(a_1^l) \cup N_F(a_i^l))} (\min\{S_j, S_j^*\} - S_j) \tag{2}
\end{aligned}$$

for  $i = 2, 3, \dots, m_l$ .

Then, we consider the operation  $drop(a)$  for all  $a \in A_{r+1}$ . There are two cases for the new cost of  $j \in N_F(a)$  as follows.

**Case 1':**  $j \in N_F(a) \cap \mathcal{D}_p^*$ . In this case, the new cost of  $j$  is at most  $p_j$ .

**Case 2':**  $j \in N_F(a) \setminus \mathcal{D}_p^*$ . Since  $a$  is a good facility, the bad facility  $s(o_j)$  is not dropped, implying the new cost of  $j$  can be bounded by  $\min\{d(s(o_j), j)\}$ .

Due to the local optimality of the solution  $F$ , we have

$$\begin{aligned}
0 & \leq \text{cost}(F \setminus \{a\}) - \text{cost}(F) \\
& \leq -f_a + \sum_{j \in N_F(a) \cap \mathcal{D}_p^*} (p_j - S_j) + \sum_{j \in N_F(a) \setminus \mathcal{D}_p^*} (d(s(o_j), j) - S_j). \tag{3}
\end{aligned}$$

Summing the inequalities (1), (2), and (3) for all swap and drop operations, we have

$$\begin{aligned}
0 & \leq -\sum_{a \in F} f_a + \sum_{b \in F^*} f_b + \sum_{a \in F} \sum_{j \in N_F(a) \cap \mathcal{D}_p^*} (p_j - S_j) \\
& + \sum_{l=1}^r \left( \sum_{j \in N_F(a_1^l) \cap N_{F^*}(b_1^l)} (S_j^* - S_j) + \sum_{i=2}^{m_l} \sum_{j \in N_F(a_i^l) \cap N_{F^*}(b_i^l)} (d(s(o_j), j) - S_j) \right. \\
& + \sum_{i=1}^{m_l} \sum_{\substack{j \in N_F(a_i^l) \setminus (\mathcal{D}_p^* \cup N_{F^*}(b_i^l)) \\ s(o_j) \neq a_i^l}} (d(s(o_j), j) - S_j) \\
& + \sum_{\substack{j \in N_F(a_1^l) \setminus (\mathcal{D}_p^* \cup N_{F^*}(b_1^l)) \\ s(o_j) = a_1^l}} (d(b_1^l, j) - S_j) + \sum_{i=2}^{m_l} \sum_{j \in N_{F^*}(b_i^l) \cap N_F(a_1^l)} (S_j^* - S_j) \\
& + \sum_{i=1}^{m_l} \sum_{j \in N_{F^*}(b_i^l) \cap \mathcal{D}_p} (\min\{p_j, S_j^*\} - p_j) \\
& + \sum_{i=1}^{m_l} \sum_{j \in N_{F^*}(b_i^l) \setminus (\mathcal{D}_p \cup N_F(a_1^l) \cup N_F(a_i^l))} (\min\{S_j, S_j^*\} - S_j) \Big) \\
& + \sum_{a \in A_{r+1}} \sum_{j \in N_F(a) \setminus \mathcal{D}_p^*} (d(s(o_j), j) - S_j) \\
& \leq -\sum_{a \in F} f_a + \sum_{b \in F^*} f_b + \sum_{j \in \mathcal{D}_s \cap \mathcal{D}_p^*} (p_j - S_j)
\end{aligned}$$



$$\begin{aligned}
& + \sum_{l=1}^r \left( \sum_{\substack{j \in N_F(a_1^l) \setminus \mathcal{D}_p^* \\ s(o_j)=a_1^l}} (S_j^* - S_j) + \sum_{\substack{j \in N_F(a_1^l) \setminus (\mathcal{D}_p^* \cup N_{F^*}(b_1^l)) \\ s(o_j)=a_1^l}} (d(b_1^l, j) - S_j) \right) \\
& + \sum_{j \in \mathcal{D}_s^* \cap \mathcal{D}_p} (\min\{p_j, S_j^*\} - p_j) + \sum_{a \in F} \sum_{\substack{j \in N_F(a) \setminus \mathcal{D}_p^* \\ s(o_j) \neq a_i^l}} (d(s(o_j), j) - S_j) \\
\leq & - \sum_{a \in F} f_a + \sum_{b \in F^*} f_b + \sum_{j \in \mathcal{D}_s \cap \mathcal{D}_p^*} (p_j - S_j) + \sum_{j \in \mathcal{D}_s^* \cap \mathcal{D}_p} (\min\{p_j, S_j^*\} - p_j) \\
& + \sum_{l=1}^r \left( \sum_{\substack{j \in N_F(a_1^l) \setminus \mathcal{D}_p^* \\ s(o_j)=a_1^l}} \left( (S_j^* - S_j) + (4S_j + S_j^* + 4\sqrt{S_j S_j^*} - S_j) \right) \right) \\
& + \sum_{a \in F} \sum_{\substack{j \in N_F(a) \setminus \mathcal{D}_p^* \\ s(o_j) \neq a_i^l}} (S_j + 4S_j^* + 4\sqrt{S_j S_j^*} - S_j) \\
\leq & - \sum_{a \in F} f_a + \sum_{b \in F^*} f_b + \sum_{j \in \mathcal{D}_s \cap \mathcal{D}_p^*} p_j + \sum_{j \in \mathcal{D}_p \cap \mathcal{D}_p^*} p_j - \sum_{j \in \mathcal{D}_p \cap \mathcal{D}_p^*} p_j - \sum_{j \in \mathcal{D}_s^* \cap \mathcal{D}_p} p_j \\
& + \sum_{j \in \mathcal{D}_s^* \cap \mathcal{D}_p} \min\{p_j, S_j^*\} + \sum_{j \in \mathcal{D}_s \cap \mathcal{D}_s^*} (2S_j + 6S_j^* + 8\sqrt{S_j S_j^*}) \\
\leq & -C_f + C_f^* + C_p^* - C_p + 2C_s + 6C_s^* + 8\sqrt{C_s C_s^*}
\end{aligned}$$

where the third inequality follows from Lemmas 2.1(b) and 2.2, the fifth inequality follows from Lemma 2.1(a). The proof is completed.  $\square$

**Lemma 2.4.** *If  $|F| \leq |F^*|$ , we have*

$$C_f + C_p \leq C_f^* + 6C_s^* + 2C_s + 8\sqrt{C_s^* C_s} + C_p^*.$$

*Proof.* Same to the proof of Lemma 2.3, we can get the inequalities (1) and (2) for the swap operations.

For the add operation  $add(b)$  for all  $b \in B_{r+1}$ , the new cost of each client  $j \in N_{F^*}(b)$  can be bounded by  $\min\{S_j^*, S_j, p_j\}$ , and the old cost of  $j$  is  $S_j$  if  $j \notin \mathcal{D}_p$  otherwise  $p_j$ . Due to the local optimality, we have

$$\begin{aligned}
0 & \leq f_b + \sum_{j \in N_{F^*}(b) \cap \mathcal{D}_p} (S_j^* - p_j) + \sum_{j \in N_{F^*}(b) \setminus \mathcal{D}_p} (\min\{S_j^*, S_j\} - S_j) \\
& \leq f_b + \sum_{j \in N_{F^*}(b) \cap \mathcal{D}_p} (S_j^* - p_j) + \sum_{j \in N_{F^*}(b) \cap N_F(a_1^l), s(o_j)=a_1^l} (S_j^* - S_j), \quad (4)
\end{aligned}$$

where  $a_1^l$  is the bad facility capturing  $b$ .

Summing the inequalities (1), (2), and (4) for all swap and add operations, and use the the similar analysis in Lemma 2.3, we have

$$0 \leq - \sum_{a \in F} f_a + \sum_{b \in F^*} f_b + \sum_{a \in F} \sum_{j \in N_F(a) \cap \mathcal{D}_p^*} (p_j - S_j) + \sum_{b \in B_{r+1}} \sum_{j \in N_{F^*}(b) \cap \mathcal{D}_p} (S_j^* - p_j)$$

$$\begin{aligned}
& + \sum_{l=1}^r \left( \sum_{j \in N_F(a_1^l) \cap N_{F^*}(b_1^l)} (S_j^* - S_j) + \sum_{i=2}^{m_l} \sum_{j \in N_F(a_i^l) \cap N_{F^*}(b_i^l)} (d(s(o_j), j) - S_j) \right. \\
& + \sum_{i=1}^{m_l} \sum_{\substack{j \in N_F(a_i^l) \setminus (\mathcal{D}_p^* \cup N_{F^*}(b_i^l)) \\ s(o_j) \neq a_i^l}} (d(s(o_j), j) - S_j) \\
& + \sum_{\substack{j \in N_F(a_1^l) \setminus (\mathcal{D}_p^* \cup N_{F^*}(b_1^l)) \\ s(o_j) = a_1^l}} (d(b_1^l, j) - S_j) + \sum_{i=2}^{m_l} \sum_{j \in N_{F^*}(b_i^l) \cap N_F(a_1^l)} (S_j^* - S_j) \\
& + \sum_{i=1}^{m_l} \sum_{j \in N_{F^*}(b_i^l) \cap \mathcal{D}_p} (\min\{p_j, S_j^*\} - p_j) \\
& + \sum_{i=1}^{m_l} \sum_{j \in N_{F^*}(b_i^l) \setminus (\mathcal{D}_p \cup N_F(a_1^l) \cup N_F(a_i^l))} (\min\{S_j, S_j^*\} - S_j) \Big) \\
& + \sum_{b \in B_{r+1}} \sum_{j \in N_{F^*}(b) \cap N_F(a_1^l), s(o_j) = a_1^l} (S_j^* - S_j) \\
\leq & - \sum_{a \in F} f_a + \sum_{b \in F^*} f_b + \sum_{j \in \mathcal{D}_s \cap \mathcal{D}_p^*} (p_j - S_j) + \sum_{j \in \mathcal{D}_s^* \cap \mathcal{D}_p} (S_j^* - p_j) \\
& + \sum_{l=1}^r \left( \sum_{\substack{j \in N_F(a_1^l) \setminus \mathcal{D}_p^* \\ s(o_j) = a_1^l}} \left( (S_j^* - S_j) + (4S_j + S_j^* + 4\sqrt{S_j S_j^*} - S_j) \right) \right) \\
& + \sum_{a \in F} \sum_{\substack{j \in N_F(a) \setminus \mathcal{D}_p^* \\ s(o_j) \neq a_i^l}} (S_j + 4S_j^* + 4\sqrt{S_j S_j^*} - S_j) \\
\leq & -C_f + C_f^* + C_p^* - C_p + 2C_s + 6C_s^* + 8\sqrt{C_s C_s^*}.
\end{aligned}$$

The proof is completed.  $\square$

Next, we will analyze the upper bound of the connection and penalty cost of the solution  $F$ , using above two types of partitions of  $F$  and  $F^*$ . Let  $N_{F^*}(B) := \bigcup_{b \in B} N_{F^*}(b)$  for all  $B \subseteq F^*$ , and  $N_F(A) := \bigcup_{a \in A} N_F(a)$  for all  $A \subseteq S$ . For the case that  $|F| > |F^*|$ , we construct swap and drop operations as follows.

- For each pair  $(A_l, B_l)$  with  $|A_l| = |B_l| \leq q$ , we consider the operation  $swap(A_l, B_l)$  (see FIGURE 3(a)).
- For each pair  $(A_l, B_l)$  with  $|A_l| = |B_l| = q' > q$ , we consider the operations  $swap(a, b)$  for all  $a \in A_l \setminus \{a_1^l\}$  and  $b \in B_l$  (see FIGURE 3(b)).
- For each facility  $a \in A_{r+1}$ , we consider the operation  $drop(a)$ .

For the case that  $|F| \leq |F^*|$ , we construct the operation  $add(o)$  for all  $o \in F^*$ . Using these local operations, we can get the upper bounds of the connection and penalty cost of  $F$ , as shown in the following two lemmas.

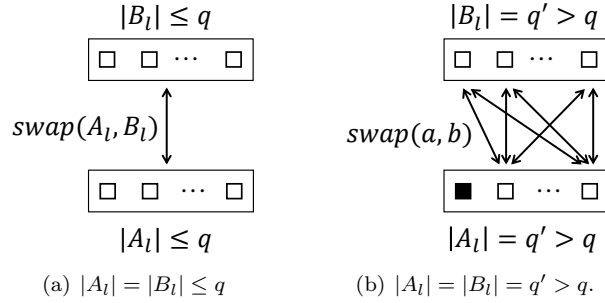


FIGURE 3. Swap operations (case of  $|F| > |F^*|$ ) for analyzing the upper bound of  $C_s + C_p$ . The black solid square is a bad facility.

**Lemma 2.5.** *If  $|F| > |F^*|$ , we have*

$$C_s + C_p \leq C_f^* + \left(5 + \frac{4}{q}\right) C_s^* + \left(4 + \frac{4}{q}\right) \sqrt{C_s^* C_s} + \left(1 + \frac{1}{q}\right) C_p^*.$$

*Proof.* For the operation  $swap(A_l, B_l)$ , we consider the following three cases for the new cost of  $j \in N_{F^*}(B_l) \cup N_F(A_l)$ .

**Case 1:**  $j \in N_{F^*}(B_l)$ . In this case, the facility  $o_j$  is swapped into the new solution, implying the new cost of  $j$  is bounded by  $S_j^*$ . The old cost of  $j$  is  $S_j$  if  $j \notin \mathcal{D}_p$ , otherwise  $p_j$ .

**Case 2:**  $j \in N_F(A_l) \setminus (N_{F^*}(B_l) \cup \mathcal{D}_p^*)$ . Since  $j \notin N_{F^*}(B_l)$ , we have  $s(o_j) \notin A_l$ , implying the new cost of  $j$  can be bounded by  $d(s(o_j), j)$ . The old cost of  $j$  is  $S_j$ .

**Case 3:**  $j \in N_F(A_l) \setminus N_{F^*}(B_l) \cap \mathcal{D}_p^*$ . The new cost of  $j$  is bounded by  $p_j$  and the old cost of  $j$  is  $S_j$ .

Due to the local optimality, we have

$$\begin{aligned}
0 &\leq \text{cost}(F \setminus A_l \cup B_l) - \text{cost}(F) \\
&\leq - \sum_{a \in A_l} f_a + \sum_{b \in B_l} f_b + \sum_{j \in N_{F^*}(B_l) \setminus \mathcal{D}_p} (S_j^* - S_j) + \sum_{j \in N_{F^*}(B_l) \cap \mathcal{D}_p} (S_j^* - p_j) \\
&\quad + \sum_{j \in N_F(A_l) \setminus (N_{F^*}(B_l) \cup \mathcal{D}_p^*)} (d(s(o_j), j) - S_j) + \sum_{j \in N_F(A_l) \setminus N_{F^*}(B_l) \cap \mathcal{D}_p^*} (p_j - S_j).
\end{aligned} \tag{5}$$

For the operation  $swap(a, b)$  for all  $a \in A_l \setminus \{a_1^l\}$  and  $b \in B_l$ , there are three cases for the new cost of  $j \in N_F(a) \cup N_{F^*}(b)$  as follows.

**Case 1':**  $j \in N_{F^*}(b)$ . The new cost of  $j$  is bounded by  $S_j^*$ . The old cost of  $j$  is  $S_j$  if  $j \notin \mathcal{D}_p$ , otherwise  $p_j$ .

**Case 2':**  $j \in N_F(a) \setminus (N_{F^*}(b) \cup \mathcal{D}_p^*)$ . The new cost of  $j$  is bounded by  $d(s(o_j), j)$  and the old cost of  $j$  is  $S_j$ .

**Case 3':**  $j \in N_F(a) \setminus N_{F^*}(b) \cap \mathcal{D}_p^*$ . The new cost of  $j$  is bounded by  $p_j$  and the old cost of  $j$  is  $S_j$ .

Due to the local optimality, we have

$$0 \leq \text{cost}(F \setminus \{a\} \cup \{b\}) - \text{cost}(F)$$

$$\begin{aligned}
&\leq -f_a + f_b + \sum_{j \in N_{F^*}(b) \setminus \mathcal{D}_p} (S_j^* - S_j) + \sum_{j \in N_{F^*}(b) \cap \mathcal{D}_p} (S_j^* - p_j) \\
&\quad + \sum_{j \in N_F(a) \setminus (N_{F^*}(b) \cup \mathcal{D}_p^*)} (d(s(o_j), j) - S_j) + \sum_{j \in N_F(a) \setminus N_{F^*}(b) \cap \mathcal{D}_p^*} (p_j - S_j).
\end{aligned} \tag{6}$$

Summing the inequalities (6) for all swap operations in the pair  $(A_l, B_l)$ , we have

$$\begin{aligned}
0 &\leq - \sum_{a \in A_l \setminus \{a_1^l\}} \sum_{b \in B_l} f_a + \sum_{b \in B_l} \sum_{a \in A_l \setminus \{a_1^l\}} f_b \\
&\quad + \sum_{b \in B_l} \sum_{a \in A_l \setminus \{a_1^l\}} \left( \sum_{j \in N_{F^*}(b) \setminus \mathcal{D}_p} (S_j^* - S_j) + \sum_{j \in N_{F^*}(b) \cap \mathcal{D}_p} (S_j^* - p_j) \right) \\
&\quad + \sum_{a \in A_l \setminus \{a_1^l\}} \sum_{b \in B_l} \left( \sum_{j \in N_F(a) \setminus (N_{F^*}(b) \cup \mathcal{D}_p^*)} (d(s(o_j), j) - S_j) + \right. \\
&\quad \quad \left. \sum_{j \in N_F(A_l \setminus \{a_1^l\}) \setminus N_{F^*}(B_l) \cap \mathcal{D}_p^*} (p_j - S_j) \right) \\
&\leq -q' \sum_{a \in A_l \setminus \{a_1^l\}} f_a + (q' - 1) \sum_{b \in B_l} f_b \\
&\quad + (q' - 1) \sum_{b \in B_l} \left( \sum_{j \in N_{F^*}(b) \setminus \mathcal{D}_p} (S_j^* - S_j) + \sum_{j \in N_{F^*}(b) \cap \mathcal{D}_p} (S_j^* - p_j) \right) \\
&\quad + q' \sum_{a \in A_l \setminus \{a_1^l\}} \left( \sum_{j \in N_F(a) \setminus \mathcal{D}_p^*} (d(s(o_j), j) - S_j) + \sum_{j \in N_F(a) \cap \mathcal{D}_p^*} (p_j - S_j) \right) \\
&\leq \sum_{b \in B_l} f_b + \sum_{b \in B_l} \left( \sum_{j \in N_{F^*}(b) \setminus \mathcal{D}_p} (S_j^* - S_j) + \sum_{j \in N_{F^*}(b) \cap \mathcal{D}_p} (S_j^* - p_j) \right) \\
&\quad + \left(1 + \frac{1}{q}\right) \sum_{a \in A_l \setminus \{a_1^l\}} \left( \sum_{j \in N_F(a) \setminus \mathcal{D}_p^*} (d(s(o_j), j) - S_j) \right. \\
&\quad \quad \left. + \sum_{j \in N_F(a) \cap \mathcal{D}_p^*} (p_j - S_j) \right),
\end{aligned} \tag{7}$$

where the third inequality follows from the fact that  $q'/(q' - 1) \leq 1 + 1/q$ .

For the operation  $drop(a)$  for all  $a \in A_{r+1}$ , we can get the inequality (3) from the local optimality. Summing the inequalities (5) for all pairs with  $|A_l| = |B_l| \leq q$ , (7) for all pairs with  $|A_l| = |B_l| = q' > q$ , and (3) for all  $a \in A_{r+1}$ , we have

$$0 \leq \sum_{o \in F^*} f_o + \sum_{j \in \mathcal{D}_s^* \setminus \mathcal{D}_p} (S_j^* - S_j) + \sum_{j \in \mathcal{D}_s^* \cap \mathcal{D}_p} (S_j^* - p_j)$$

$$\begin{aligned}
& + \left(1 + \frac{1}{q}\right) \cdot \left( \sum_{j \in \mathcal{D}_s \setminus \mathcal{D}_p^*} (d(s(o_j), j) - S_j) + \sum_{j \in \mathcal{D}_s \cap \mathcal{D}_p^*} (p_j - S_j) \right) \\
\leq & \sum_{o \in F^*} f_o + \sum_{j \in \mathcal{D}_s^* \setminus \mathcal{D}_p} (S_j^* - S_j) + \sum_{j \in \mathcal{D}_s^* \cap \mathcal{D}_p} (S_j^* - p_j) + \sum_{j \in \mathcal{D}_p \cap \mathcal{D}_p^*} (p_j - p_j) \\
& + \left(1 + \frac{1}{q}\right) \cdot \left( \sum_{j \in \mathcal{D}_s \setminus \mathcal{D}_p^*} (S_j + 4S_j^* + 4\sqrt{S_j S_j^*} - S_j) + \sum_{j \in \mathcal{D}_s \cap \mathcal{D}_p^*} (p_j - S_j) \right) \\
\leq & \sum_{o \in F^*} f_o + \sum_{j \in \mathcal{D}_s^*} S_j^* - \sum_{j \in \mathcal{D}_s} S_j - \sum_{j \in \mathcal{D}_p} p_j + \left(1 + \frac{1}{q}\right) \sum_{j \in \mathcal{D}_p^*} p_j \\
& + \left(1 + \frac{1}{q}\right) \sum_{j \in \mathcal{D}_s \cap \mathcal{D}_p^*} (4S_j^* + 4\sqrt{S_j S_j^*}) \\
\leq & C_f^* + C_s^* - C_s - C_p + \left(1 + \frac{1}{q}\right) (C_p^* + 4C_s^* + 4\sqrt{C_s C_s^*}) \\
\leq & C_f^* + \left(5 + \frac{4}{q}\right) C_s^* + \left(4 + \frac{4}{q}\right) \sqrt{C_s C_s^*} + \left(1 + \frac{1}{q}\right) C_p^* - C_s - C_p,
\end{aligned}$$

where the second inequality follows from Lemma 2.1(b) and the fourth inequality follows from Lemma 2.1(a).  $\square$

**Lemma 2.6.** *If  $|F| \leq |F^*|$ , we have*

$$C_s + C_p \leq C_f^* + C_s^* + C_p^*.$$

*Proof.* We consider the operation  $add(o)$  for all  $o \in F^*$ . Then it is easy to get the following inequality from the local optimality.

$$\begin{aligned}
0 & \leq \sum_{o \in F^*} (\text{cost}(F \cup \{o\}) - \text{cost}(F)) \\
& \leq \sum_{o \in F^*} \left( f_o + \sum_{j \in N_{F^*}(o) \setminus \mathcal{D}_p} (S_j^* - S_j) + \sum_{j \in N_{F^*}(o) \cap \mathcal{D}_p} (S_j^* - p_j) \right) \\
& \quad + \sum_{j \in \mathcal{D}_s \cap \mathcal{D}_p^*} (p_j - S_j) + \sum_{j \in \mathcal{D}_p \cap \mathcal{D}_p^*} (p_j - p_j) \\
& \leq \sum_{o \in F^*} f_o + \sum_{j \in \mathcal{D}_s^*} S_j^* + \sum_{j \in \mathcal{D}_p^*} p_j - \sum_{j \in \mathcal{D}_s} S_j - \sum_{j \in \mathcal{D}_p} p_j \\
& = C_f^* + C_s^* + C_p^* - C_s - C_p
\end{aligned}$$

which completes the proof.  $\square$

From Lemmas 2.3 and 2.4, we have

$$\text{cost}(F) = C_f + C_p + C_s \leq C_f^* + 6C_s^* + 3C_s + 8\sqrt{C_s^* C_s} + C_p^*. \quad (8)$$

We need to bound  $\sqrt{C_s^* C_s}$  by  $C_f^*$ ,  $C_s^*$  and  $C_p^*$ . For this end, we use Lemmas 2.5 and 2.6 to get

$$0 \geq C_s - C_f^* - \left(5 + \frac{4}{q}\right) C_s^* - \left(4 + \frac{4}{q}\right) \sqrt{C_s^* C_s} - \left(1 + \frac{1}{q}\right) C_p^*$$

$$\begin{aligned}
&= \left( \sqrt{C_s} - \left(2 + \frac{2}{q}\right) \sqrt{C_s^*} \right)^2 - \left( C_f^* + \left(1 + \frac{1}{q}\right) C_p^* + \left(3 + \frac{2}{q}\right)^2 C_s^* \right) \\
&= \left( \sqrt{C_s} - \left(2 + \frac{2}{q}\right) \sqrt{C_s^*} + \sqrt{C_f^* + \left(1 + \frac{1}{q}\right) C_p^* + \left(3 + \frac{2}{q}\right)^2 C_s^*} \right) \\
&\quad \cdot \left( \sqrt{C_s} - \left(2 + \frac{2}{q}\right) \sqrt{C_s^*} - \sqrt{C_f^* + \left(1 + \frac{1}{q}\right) C_p^* + \left(3 + \frac{2}{q}\right)^2 C_s^*} \right). \quad (9)
\end{aligned}$$

It is easy to see that

$$\sqrt{C_s} - \left(2 + \frac{2}{q}\right) \sqrt{C_s^*} + \sqrt{C_f^* + \left(1 + \frac{1}{q}\right) C_p^* + \left(3 + \frac{2}{q}\right)^2 C_s^*} > 0.$$

Together with (9), we have

$$\sqrt{C_s} \leq \left(2 + \frac{2}{q}\right) \sqrt{C_s^*} + \sqrt{C_f^* + \left(1 + \frac{1}{q}\right) C_p^* + \left(3 + \frac{2}{q}\right)^2 C_s^*}. \quad (10)$$

From the inequalities (8) and (10), we have

$$\begin{aligned}
\text{cost}(F) &\leq C_f^* + 6C_s^* + C_p^* \\
&\quad + 3 \left( \left(2 + \frac{2}{q}\right) \sqrt{C_s^*} + \sqrt{C_f^* + \left(1 + \frac{1}{q}\right) C_p^* + \left(3 + \frac{2}{q}\right)^2 C_s^*} \right)^2 \\
&\quad + 8\sqrt{C_s^*} \left( \left(2 + \frac{2}{q}\right) \sqrt{C_s^*} + \sqrt{C_f^* + \left(1 + \frac{1}{q}\right) C_p^* + \left(3 + \frac{2}{q}\right)^2 C_s^*} \right) \\
&= 4C_f^* + \left(61 + \frac{76}{q} + \frac{24}{q^2}\right) C_s^* + \left(3 + \frac{3}{q}\right) C_p^* \\
&\quad + \left(20 + \frac{12}{q}\right) \sqrt{C_s^*} \sqrt{C_f^* + \left(1 + \frac{1}{q}\right) C_p^* + \left(3 + \frac{2}{q}\right)^2 C_s^*} \\
&\leq 4C_f^* + \left(61 + \frac{76}{q} + \frac{24}{q^2}\right) C_s^* + \left(3 + \frac{3}{q}\right) C_p^* \\
&\quad + \left(10 + \frac{6}{q}\right) \left( C_s^* + C_f^* + \left(1 + \frac{1}{q}\right) C_p^* + \left(3 + \frac{2}{q}\right)^2 C_s^* \right) \\
&\leq \left(14 + \frac{6}{q}\right) C_f^* + \left(161 + \frac{256}{q} + \frac{136}{q^2} + \frac{24}{q^3}\right) C_s^* \\
&\quad + \left(13 + \frac{19}{q} + \frac{6}{q^2}\right) C_p^* \\
&\leq \left(161 + \frac{256}{q} + \frac{136}{q^2} + \frac{24}{q^3}\right) \text{cost}(F^*). \quad (11)
\end{aligned}$$

Immediately, we get the following theorem.

**Theorem 2.7.** *For the SM- $k$ -FLPLP, the locality gap of Algorithm 1 is bounded by*

$$161 + \frac{256}{q} + \frac{136}{q^2} + \frac{24}{q^3}.$$

Note that Algorithm 1 may suffer exponential number of iterations. To reduce the time complexity, we propose Algorithm 1' by replacing  $\text{cost}(F') < \text{cost}(F)$  in Algorithm 1 to  $\text{cost}(F') < (1 - \epsilon/Q)\text{cost}(F)$ , where  $Q$  is an upper bound of the number of local search operations constructed in the analysis for bounding the cost and  $\epsilon$  is any small positive number. From Arya et al. [1], we have the following result.

**Lemma 2.8** (Arya et al. [1]). *The number of iterations of Algorithm 1' is at most  $\log(\text{cost}(F_0)/\text{cost}(F^*)) / \log \frac{1}{1-\epsilon/Q}$ , where  $F_0$  is the initial feasible solution for the SM- $k$ -FLPLP. If the locality gap of Algorithm 1 is  $\alpha$ , then the locality gap of Algorithm 1' is  $\alpha + \epsilon'$ , where  $\epsilon' = \frac{\alpha\epsilon}{1-\epsilon} \sim \epsilon$ .*

It is easy to prove that we can set  $Q = k^2$  from our analysis (for the case of  $F > F^*$ , we constructed at most  $k + k(k-1)$  swap and drop operations; for the case of  $F \leq F^*$ , we constructed at most  $2k$  swap and add operations), and  $|N(F)| = O(|\mathcal{F}|^q)$  from the definition of  $N(F)$ . Besides,  $\log(\text{cost}(F_0)/\text{cost}(F^*))$  and the time for calculating  $\text{cost}(F)$  are polynomial in the input size, so Algorithm 1' is polynomial-time according to Lemma 2.8. Immediately, we obtain the following theorem.

**Theorem 2.9.** *For the SM- $k$ -FLPLP, there exists a  $(161 + 256/q + 136/q^2 + 24/q^3 + \epsilon)$ -approximation algorithm based on the local search technique, where  $q$  is the upper bound of the number of swapped facilities in each iteration, and  $\epsilon$  is any small positive constant.*

**2.3. Improve the approximation ratio by scaling.** From the inequality (11), we observe that it has a big gap between the approximation guarantees of  $C_f^*$ ,  $C_s^*$  and  $C_p^*$ . Using the scaling technique (refer to [3]), we can reduce this gap to improve the approximation ratio. The algorithm with scaling is shown below.

---

**Algorithm 2** Scaling algorithm for the SM- $k$ -FLPLP

---

**Input:** Instance  $\mathcal{I} = (\mathcal{F}, \mathcal{D}, k, f, p, \Omega)$ , parameters  $q$  and  $\delta$ .

**Output:** A local optimal solution.

- 1: Construct a new instance  $\mathcal{I}' = (\mathcal{F}, \mathcal{D}, k, f', p', \Omega)$  by scaling the facility opening and penalty costs, i. e.

$$f'_i := \delta f_i, \quad \forall i \in \mathcal{F}, \quad p'_j := \delta p_j, \quad \forall j \in \mathcal{D}.$$

- 2: Call Algorithm 1 on the new instance  $\mathcal{I}'$  to get a solution  $F$ .

- 3: **return**  $F$
- 

We use  $C_f(F, \mathcal{I})$ ,  $C_s(F, \mathcal{I})$  and  $C_p(F, \mathcal{I})$  to denote the facility opening, connection and penalty costs of the solution  $F$  on the instance  $\mathcal{I}$ , and use  $C_f^*(\mathcal{I})$ ,  $C_s^*(\mathcal{I})$  and  $C_p^*(\mathcal{I})$  to denote the facility opening, connection and penalty costs of the optimal solution of the instance  $\mathcal{I}$ .

From the analysis (including all proofs) in Section 2.2, we can see that when replacing the optimal solution  $F^*$  with any feasible solution  $X$ , the analysis results

still hold. Since the optimal solution  $F^*$  of  $\mathcal{I}$  is a feasible solution of  $\mathcal{I}'$ , we have the following two inequalities from Lemmas 2.3 and 2.4, and the inequality (10).

$$\begin{aligned} C_f(F, \mathcal{I}') + C_p(F, \mathcal{I}') &\leq C_f(F^*, \mathcal{I}') + 6C_s(F^*, \mathcal{I}') + 2C_s(F, \mathcal{I}') \\ &\quad + 8\sqrt{C_s(F^*, \mathcal{I}')C_s(F, \mathcal{I}')} + C_p(F^*, \mathcal{I}') \end{aligned} \quad (12)$$

and

$$\begin{aligned} \sqrt{C_s(F, \mathcal{I}')} &\leq \left(2 + \frac{2}{q}\right) \sqrt{C_s(F^*, \mathcal{I}')} \\ &\quad + \sqrt{C_f(F^*, \mathcal{I}') + \left(1 + \frac{1}{q}\right) C_p(F^*, \mathcal{I}') + \left(3 + \frac{2}{q}\right)^2 C_s(F^*, \mathcal{I}')}, \end{aligned} \quad (13)$$

where  $F$  is the local optimal solution of the instance  $\mathcal{I}'$ . Using the inequality (12), we have

$$\begin{aligned} C_f(F, \mathcal{I}) + C_p(F, \mathcal{I}) &= \frac{C_f(F, \mathcal{I}') + C_p(F, \mathcal{I}')}{\delta} \\ &\leq \frac{C_f(F^*, \mathcal{I}') + 6C_s(F^*, \mathcal{I}') + 2C_s(F, \mathcal{I}') + 8\sqrt{C_s(F^*, \mathcal{I}')C_s(F, \mathcal{I}')} + C_p(F^*, \mathcal{I}')}{\delta} \\ &\leq C_f(F^*, \mathcal{I}) + C_p(F^*, \mathcal{I}) + \frac{6C_s(F^*, \mathcal{I}) + 2C_s(F, \mathcal{I}) + 8\sqrt{C_s(F^*, \mathcal{I})C_s(F, \mathcal{I})}}{\delta}. \end{aligned}$$

Adding  $C_s(F, \mathcal{I})$  to both sides of above inequality, we have

$$\begin{aligned} &C_f(F, \mathcal{I}) + C_p(F, \mathcal{I}) + C_s(F, \mathcal{I}) \\ &\leq C_f(F^*, \mathcal{I}) + C_p(F^*, \mathcal{I}) + \left(1 + \frac{2}{\delta}\right) C_s(F, \mathcal{I}) + \frac{6C_s(F^*, \mathcal{I}) + 8\sqrt{C_s(F^*, \mathcal{I})C_s(F, \mathcal{I})}}{\delta} \\ &= C_f^*(\mathcal{I}) + C_p^*(\mathcal{I}) + \left(1 + \frac{2}{\delta}\right) C_s(F, \mathcal{I}) + \frac{6C_s^*(\mathcal{I}) + 8\sqrt{C_s^*(\mathcal{I})C_s(F, \mathcal{I})}}{\delta}. \end{aligned}$$

Together with the inequality (13) and after some mathematical deductions, we have

$$\begin{aligned} &C_f(F, \mathcal{I}) + C_p(F, \mathcal{I}) + C_s(F, \mathcal{I}) \\ &\leq \max \left\{ 11 + 3\delta + \frac{4 + 2\delta}{q}, 11 + 3\delta + \frac{14 + 5\delta}{q} + \frac{4 + 2\delta}{q^2}, \right. \\ &\quad \left. 33 + \frac{128}{\delta} + \frac{64 + 192/\delta}{q} + \frac{40 + 96/\delta}{q^2} + \frac{8 + 16/\delta}{q^3} \right\} \cdot \text{cost}(F^*). \end{aligned}$$

By setting

$$\begin{aligned} \delta &= \frac{11 + \frac{25}{q} + \frac{18}{q^2} + \frac{4}{q^3}}{3 + \frac{5}{q} + \frac{2}{q^2}} + \\ &\quad \frac{\sqrt{\left(11 + \frac{25}{q} + \frac{18}{q^2} + \frac{4}{q^3}\right)^2 + \left(3 + \frac{5}{q} + \frac{2}{q^2}\right) \left(128 + \frac{192}{q} + \frac{96}{q^2} + \frac{16}{q^3}\right)}}{3 + \frac{5}{q} + \frac{2}{q^2}}, \end{aligned}$$

we get that the locality gap of Algorithm 2 is

$$11 + 3\delta + \frac{14 + 5\delta}{q} + \frac{4 + 2\delta}{q^2}.$$



When  $q$  is enough large, the locality gap is approximately  $22 + \sqrt{505}$ . Consequently, we have the following theorem.

**Theorem 2.10.** *For the SM- $k$ -FLPLP, there exists a  $(22 + \sqrt{505} + \epsilon)$ -approximation algorithm based on the local search technique, where  $\epsilon$  is any small positive constant.*

**3. Conclusion.** In this paper, we consider the squared metric  $k$ -facility location problem with linear penalties, a variant of the classical  $k$ -median problem. We apply the local search scheme with add, drop, and swap operations to this problem, and prove that it is a  $(161 + 256/q + 136/q^2 + 24/q^3 + \epsilon)$ -approximation algorithm where  $\epsilon$  is any small positive number and  $q$  is the upper bound of the number of swapped facilities in each iteration. When  $q$  is large enough, the approximation ratio approaches to  $162 + \epsilon$ . Furthermore, we improve the approximation ratio to  $22 + \sqrt{505} + \epsilon$  by using the scaling technique.

In the future, we can study how to improve the performance of the local search scheme on this problem and other important variants of  $k$ -median problem with generalized metric, submodular penalties, capacitated constraint, etc.

**Acknowledgments.** The second author is supported by National Natural Science Foundation of China (No. 11871081). The third author is supported by National Natural Science Foundation of China (Nos. 61672323 and 61972228). The fourth author is supported by National Natural Science Foundation of China (No. 61433012), Shenzhen research grant (KQJSCX20180330170311901, JCYJ20180305180840138, GGF2017073114031767) and Shenzhen Discipline Construction Project for Urban Computing and Data Intelligence.

## REFERENCES

- [1] V. Arya, N. Garg, R. Khandekar, A. Meyerson, K. Munagala and V. Pandit, [Local search heuristics for  \$k\$ -median and facility location problems](#), *SIAM Journal on Computing*, **33** (2004), 544–562.
- [2] J. Byrka, T. Pensyl, B. Rybicki, A. Srinivasan and K. Trinh, [An improved approximation for  \$k\$ -median and positive correlation in budgeted optimization](#), *ACM Transactions on Algorithms*, **13** (2017), Art. 23, 31 pp.
- [3] M. Charikar and S. Guha, [Improved combinatorial algorithms for facility location problems](#), *SIAM Journal on Computing*, **34** (2005), 803–824.
- [4] M. Charikar, S. Guha, É. Tardos and D. B. Shmoys, [A constant-factor approximation algorithm for the  \$k\$ -median problem](#), *Annual ACM Symposium on Theory of Computing (Atlanta, GA, 1999)*, *ACM, New York*, (1999), 1–10.
- [5] M. Charikar, S. Khuller, D. M. Mount and G. Narasimhan, Algorithms for facility location problems with outliers, *Proceedings of the 12th Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, Philadelphia, PA*, (2001), 642–651.
- [6] F. A. Chudak and D. B. Shmoys, [Improved approximation algorithms for the uncapacitated facility location problem](#), *SIAM Journal on Computing*, **33** (2003), 1–25.
- [7] C. G. Fernandes, L. A. A. Meira, F. K. Miyazawa and L. L. C. Pedrosa, [A systematic approach to bound factor-revealing LPs and its application to the metric and squared metric facility location problems](#), *Mathematical Programming*, **153** (2015), 655–685.
- [8] M. Hajiaghayi, R. Khandekar and G. Kortsarz, [Local search algorithms for the red-blue median problem](#), *Algorithmica*, **63** (2012), 795–814.
- [9] D. S. Hochbaum, [Heuristics for the fixed cost median problem](#), *Mathematical Programming*, **22** (1982), 148–162.
- [10] K. Jain and V. V. Vazirani, [Approximation algorithms for metric facility location and  \$k\$ -median problems using the primal-dual schema and Lagrangian relaxation](#), *Journal of the ACM*, **48** (2001), 274–296.
- [11] S. Li, [A 1.488 approximation algorithm for the uncapacitated facility location problem](#), *Information and Computation*, **222** (2013), 45–58.

- [12] Y. Li, D. L. Du, N. H. Xiu and D. C. Xu, [Improved approximation algorithms for the facility location problems with linear/submodular penalties](#), *Algorithmica*, **73** (2015), 460–482.
- [13] D. B. Shmoys, É. Tardos, and K. Aardal, [Approximation algorithms for facility location problems](#), *Proceedings of the 29th Annual ACM Symposium on Theory of Computing, ACM, New York, NY*, (1997), 265–274.
- [14] Y. S. Wang, D. C. Xu, D. L. Du and C. C. Wu, [An approximation algorithm for the  \$n\$ th power metric facility location problem with linear penalties](#), *Optimization Letters*, **11** (2017), 983–993.
- [15] Y. S. Wang, D. C. Xu, D. L. Du and C. C. Wu, [An approximation algorithm for  \$k\$ -facility location problem with linear penalties using local search scheme](#), *Journal of Combinatorial Optimization*, **36** (2018), 264–279.
- [16] Y. C. Xu, D. C. Xu, D. L. Du and D. M. Zhang, [Approximation algorithm for squared metric facility location problem with nonuniform capacities](#), *Discrete Applied Mathematics*, **264** (2019), 208–217.
- [17] Y. C. Xu, D. C. Xu, D. L. Du and C. C. Wu, [Local search algorithm for universal facility location problem with linear penalties](#), *Journal of Global Optimization*, **67** (2017), 367–378.
- [18] P. Zhang, [A new approximation algorithm for the  \$k\$ -facility location problem](#), *Theoretical Computer Science*, **384** (2007), 126–135.
- [19] D. M. Zhang, D. C. Xu, Y. S. Wang, P. Zhang and Z. N. Zhang, [Local search approximation algorithms for the sum of squares facility location problems](#), *Journal of Global Optimization*, **74** (2019), 909–932.
- [20] D. M. Zhang, D. C. Xu, Y. S. Wang, P. Zhang and Z. N. Zhang, [A local search approximation algorithm for a squared metric  \$k\$ -facility location problem](#), *Journal of Combinatorial Optimization*, **35** (2018), 1168–1184.

Received October 2019; revised November 2019.

*E-mail address:* [ys.wang1@siat.ac.cn](mailto:ys.wang1@siat.ac.cn)

*E-mail address:* [zhangdongmei@sdjzu.edu.cn](mailto:zhangdongmei@sdjzu.edu.cn)

*E-mail address:* [algzhang@sdu.edu.cn](mailto:algzhang@sdu.edu.cn)

*E-mail address:* [zhangyong@siat.ac.cn](mailto:zhangyong@siat.ac.cn)