Improved Local Search for Universal Facility Location[*]

Eric Angel, Nguyen Kim Thang, and Damien Regnault

IBISC, University of Evry Val d'Essonne, France
{angel,thang,regnault}@ibisc.univ-evry.fr

Abstract. We consider the universal facility location problem in which the goal is to assign clients to facilities in order to minimize the sum of connection and facility costs. The connection cost is proportional to the distance each client has to travel to its assigned facility, whereas the cost of a facility is a non-decreasing function depending on the number of clients assigned to the facility. This model generalizes several variants of facility location problems. We present a $(5.83 + \epsilon)$ approximation algorithm for this problem based on local search technique.

1 Introduction

The class of facility location problems is fundamental in operations research and is subject of extensive study. In the classical model, facilities are opened to satisfy client demands and the opening cost of a facility is fixed. However, this model is not fully appropriate in the contexts where the cost of a facility to serve clients (the delay) crucially depends on its allocated capacity. This phenomenon widely occurs in practical situations. The following model of Universal Facility Location captures this phenomenon and is also a generalization of several variants of facility location problems.

Universal Facility Location Let $\mathcal{C}$ be the set of (*clients*) and $\mathcal{F}$ be a set of *facilities* where $n = |\mathcal{C}|$ and $m = |\mathcal{F}|$. Each facility i is characterized by a non-decreasing *cost function* $f_i : \mathbb{N} \to \mathbb{R}^+$ where $f_i(0) = 0$. Consider the complete bipartite graph $G = (\mathcal{C} \cup \mathcal{F}, E)$ where the distances $d(i, j)$ associated to facility i and client j follow the triangle inequality. We denote also by $d(i, i')$ the length of a shortest path between two facilities i and i' in the graph.

The goal is to assign clients to the facilities and install capacities at every facility in order to serve clients. Given a *solution* $S = (u, x)$, where x is the *assignment* ($x_{ij} = 1$ if client j is served by facility i; and 0 otherwise) and u is the *allocation* ($u_i \in \mathbb{N}$ denotes the capacity allocated at facility i, which equals the number of clients assigned to facility i), the *connection cost* is defined as $C_s(S) = \sum_{i \in \mathcal{F}, j \in \mathcal{C}} d(i, j) x_{ij}$ and the *facility cost* is $C_f(S) = \sum_{i \in \mathcal{F}} f_i(u_i)$. The objective is to find a feasible solution minimizing the *total cost* $C(S)$ defined as $C_s(S) + C_f(S)$.

Related works The model of Universal Facility Location captures several variants of Facility Location. Many interesting algorithms with deep, powerful techniques have been designed for the latter. Among others, the technique of local search is extensively studied. Arya et al. [1] introduced the local search technique to study Facility Location and k-median problem. From that, local search plays an important role to study variants of Facility Location.

A particular interesting variant is the Capacitated Facility Location Problem, where $f_i(\cdot)$ is constant if the assigned amount to facility i is smaller than a given capacity; and is infinity otherwise. The first constant approximation ratio for this problem is 8.53 due to Pál and Tardos [7]. Then the approximation ratio for Capacitated Facility Location Problem was improved to $(5.83 + \epsilon)$ by Zhang et al. [9] and recently to $(5 + \epsilon)$ by Bansal et al. [2].

Mahdian and Pál [6] introduced the model of Universal Facility Location and gave a $(7.88 + \epsilon)$-approximation algorithm. Garg et al. [3] proposed extended operations and a schema of analysis to prove a $(5.83 + \epsilon)$-approximation ratio. However, one of their operations is unlikely to be polynomially computable (N. Garg, personal communication, 2012). Subsequently, Vygen [8] improved the approximation ratio to $(6.702 + \epsilon)$. All those algorithms are based on local search approach and the successive improvements are done by extending and generalizing the previous operations together with more subtle analyses.

Besides, Hajiaghayi et al. [4] considered Universal Facility Location with concave cost function and designed a 1.861-approximation algorithm. Recently, Li and Khuller [5] have proved a $(\ln n + 1)$-approximation for the Universal Facility Location in non-metric space.

Contributions We present a $(5.83 + \epsilon)$-approximation algorithm also based on local search. The contribution of the paper is a simple, polynomially computable operation called `Open-close`. With this operation together with other operations, we manage to show the improved performance on the approximation of the Universal Facility Location. Note that the analysis follows closely the ones in [9, 3] with `Open-close` as the main operation.

2 Algorithm and Analysis

2.1 Operations

In this section we describe the set of operations that will be used in the algorithm.

- `Add`(s, δ): increase the capacity of facility s by δ, and find the minimum cost assignment of demands to facilities, given their allocated capacities.
- `Open`(s, δ): increase the capacity of s by sending δ units of flow from one or several facilities $i_1, i_2, \ldots$ to s via the shortest paths between $i_1, i_2, \ldots$ and s (and decrease the capacity of $i_1, i_2, \ldots$).
- `Close`(s, δ): Inversely, decrease the capacity of s by sending δ units of flow from s to one or several facilities $i_1, i_2, \ldots$ via shortest path between s and $i_1, i_2, \ldots$ (and increase the capacity of $i_1, i_2, \ldots$).

– `Open-close`$(s, t, \delta_s, \delta_t)$: increase the capacity of s by δ_s and decrease the capacity of t by δ_t. This operation consists in orienting some amount from t to s and then routing some amounts from one or several facilities $i_1, i_2, \ldots$ to s and from t to one or several facilities $i'_1, i'_2, \ldots$. The transfers are carried out via shortest paths between facilities.

In the following for each operation, given its input, we show how to compute the minimum cost (of the operation on the input) in polynomial time. The min-cost of operation `Add`(s, δ) has been shown to be efficiently computable [6, 8]. Note that the operations `Open`(s, δ) and `Close`(s, δ) are particular cases of the operation `Open-close`$(s, t, \delta_s, \delta_t)$, hence it is sufficient to prove that the min-cost of the latter could be computed in polynomial time.

Lemma 1. *Let $S = (u, x)$ be a solution and s, t be two facilities and $0 \le \delta_s, \delta_t \le n$. Then, the minimum cost of the operation `Open-close`$(s, t, \delta_s, \delta_t)$ can be computed in polynomial time with respect to n, m.*

Proof. Observe that if in operation `Open-close`$(s, t, \delta_s, \delta_t)$, some amount is sent from t to some facility i and later is reoriented to s then we can modify the transfer in such a way that the amount is routed directly from t to s. Since the distance d follows the triangle inequality, the modification results in a solution at least as good as the previous one. Hence, in the sequel we assume that for any facility, either it receives some flow or it sends out some flow.

We compute the minimum cost of `Open-close`$(s, t, \delta_s, \delta_t)$. Name $1, \ldots, m-2$ the facilities of $\mathcal{F} \setminus \{s, t\}$. Let $0 \le \delta \le \min\{\delta_s, \delta_t\}$ be the amount of flow directly sent from t to s. We need to route $\delta_s - \delta$ flow units to s and $\delta_t - \delta$ units out of t. Let $g(i, a, b)$ be the minimum cost of having already sent a flow units to s and having already sent b flow units out of t after considering the facilities $1, 2, \ldots, i$. We have:

$$\texttt{Open-close}(s, t, \delta_s, \delta_t) = \min_{0 \le \delta \le \min\{\delta_s, \delta_t\}} g(m-2, \delta_s - \delta, \delta_t - \delta) + \delta \cdot d(s, t) +$$

$$\left(f_s(u_s + \delta_s) - f_s(u_s) + f_t(u_t - \delta_t) - f_t(u_t) \right)$$

Now we compute $g(i, a, b)$ for $1 \le i \le m - 2$, $0 \le a \le \delta_s - \delta$, $0 \le b \le \delta_t - \delta$ by dynamic programming. At facility i, either i will transfer some amount to facility s or i will receive some amount from t. So we derive the recursive formula

$$g(i, a, b) = \min \Bigg\{ \min_{0 \le w \le a} g(i-1, a-w, b) + \left[w \cdot d(i, s) + f_i(u_i - w) - f_i(u_i) \right],$$

$$\min_{0 \le w \le b} g(i-1, a, b-w) + \left[w \cdot d(i, t) + f_i(u_i + w) - f_i(u_i) \right] \Bigg\}$$

for $2 \le i \le n - 2$, $0 \le a \le \delta_s - \delta$, $0 \le b \le \delta_t - \delta$ and

$$g(1, a, b) = \begin{cases} a \cdot d(1, s) + f_1(u_1 - a) - f_1(u_1) & \text{if } b = 0 \\ b \cdot d(1, t) + f_1(u_1 + b) - f_1(u_1) & \text{if } a = 0 \\ \infty & \text{if } a \ne 0, b \ne 0. \end{cases}$$

where the last case indicates that a facility can either receive or send out some amount but not both. As δ_s and δ_t are bounded by n, we can compute $\texttt{Open-close}(s, t, \delta_s, \delta_t)$ in $O(mn^4)$. □

2.2 The Local Search Algorithm

Fix $\epsilon > 0$ be a small constant. Let S be an arbitrary feasible solution. As long as there still exits some operation $\texttt{Add}(s, \delta)$ for $0 \leq \delta \leq n$ or $\texttt{Open-close}(s, t, \delta_s, \delta_t)$ for $0 \leq \delta_s, \delta_t \leq n$ (using Lemma 1) such that after the operation the cost is reduced by at least $\epsilon C(S)$, improve S by the operation. Otherwise, return S.

By the results of the previous section, at each step we can verify in polynomial time whether there is some improvement due to the operations. Moreover, the algorithm halts after at most $\frac{1}{\epsilon} \log \frac{C(S)}{C(S^*)}$ iterations where S^* is a global optimum. Hence, the running time of the algorithm is polynomial in the size of the input.

2.3 The Analysis

Note that the solution returned by the algorithm is not a local optimum (according to the operation given in previous section), but is an approximate one. However, the cost of the latter is only $(1 + 3\epsilon)$ factor worse than the bound of a local optimum. Hence, by standard argument in local search, it is sufficient to prove the bound r of a local optimum (with respect to the operations described in the previous section) to a global optimum. Consequently, the approximation ratio is $r(1 + \epsilon')$ where $\epsilon' = 3\epsilon$.

Let $S = (u, x)$ and $S^* = (u^*, x^*)$ be a local optimum solution and a global optimum, respectively. With respect to the $\texttt{Add}$ operation, the connection cost has been bounded by the following lemma.

Lemma 2 ([6, 8]). $C_s(S) \leq C_s(S^*) + C_f(S^*)$.

The remaining of the paper is devoted to bound $C_f(S)$ in function of $C_s(S^*)$ and $C_f(S^*)$ by the following strategy.

Strategy Define $\mathcal{F}^+ := \{i \in F : u_i > u_i^*\}$ and $\mathcal{F}^- := \{i \in F : u_i < u_i^*\}$. The idea of the proof is to transfer some capacity amounts from facilities in $\mathcal{F}^+$ to facilities in $\mathcal{F}^-$ based on the operations defined in the previous section while maintaining the following properties.

- For each facility $i \in \mathcal{F}^+$, move once the exact amount of $(u_i - u_i^*)$ units from i to some facilities in $\mathcal{F}^-$. We say that facility i is *closed*.
- For each facility $i \in \mathcal{F}^-$, the amount that i receives each time is at most $u_i^* - u_i$. Each time i receives some capacity amount, we say that facility i is *opened*.
- The *transportation cost* of the transfer — the cost to route capacity amounts between facilities where each unit travelling from facility i to i' along a path incurs a cost as the total length of that path — is small.

Note that a transfer is not a sequence of successive operations but is a "union" of different operations. Ideally, in the second property each facility in $\mathcal{F}^-$ is opened once. However, the operations fulfilling this purpose may not be computed in polynomial time. Let r be the maximum number of times a facility in $\mathcal{F}^-$ is open in such a transfer. Suppose that there exists a transfer with the desired properties. We show how the strategy leads to useful bounds of the facility cost. Let C_t be the transportation cost of the transfer. As S is local optimum, any operation with respect to the solution S must have non-negative cost. Denote $\delta_{i,r'}$ be the amount transferred to facility $i \in \mathcal{F}^-$ at its r'-th opening. Combining all inequalities corresponding to operations in the transfer, we have

$$\sum_{i \in \mathcal{F}^+} (f_i(u_i^*) - f_i(u_i)) + \sum_{r'=1}^{r} \sum_{i \in \mathcal{F}^-} (f_i(u_i + \delta_{i,r'}) - f_i(u_i)) + C_t \geq 0. \quad (1)$$

Therefore,

$$\sum_{i \in \mathcal{F}^+} (f_i(u_i^*) - f_i(u_i)) + r \cdot \sum_{i \in \mathcal{F}^-} (f_i(u_i^*) - f_i(u_i)) + C_t \geq 0. \quad (2)$$

since for $i \in \mathcal{F}^-$, $f_i(u_i^*) \geq f_i(u_i + \delta_{i,r'})$, which is due to $\delta_{i,r'} \leq u_i^* - u_i$ by the second property, thus the lefthand side of (1) is upper-bounded by that of (2). Summing both sides of (2) by $\sum_{i:i \notin \mathcal{F}^- \cup \mathcal{F}^+} f_i(u_i)$ and rearranging the terms, we get $C_f(S) \leq rC_f(S^*) + C_t$. Whenever C_t is small, we can derive a bound on the facility cost $C_f(S)$. We say that a transfer is *feasible* if it satisfies the first two properties. We will look for feasible transfers with small transportation cost and r as small as possible.

Consider a transportation problem: finding a min cost flow such that each facility $i \in \mathcal{F}^+$ sends out $u_i - u_i^*$ units of flow and each facility $i \in \mathcal{F}^-$ receives $u_i^* - u_i$ units. The cost of shipping one flow unit between i and i' equals $d(i, i')$. Mahdian and Pál [6] proved that the minimum cost flow was at most $C_s(S) + C_s(S^*)$. Moreover, the support graph of the min cost transportation forms a forest with edges going between $\mathcal{F}^+$ and $\mathcal{F}^-$. The transfers that we will define later are carried out based on this forest.

Root each tree in the forest at some fixed facility in $\mathcal{F}^-$. For each vertex v, denote $K(v)$ the set of its children. For each vertex $t \in \mathcal{F}^-$, let T_t be the subtree of depth *exactly* 2 rooted at t containing all its children and grand children. (We can add some dummy vertices where the in-flow and out-flow are 0 such that every tree T_t has depth 2.) Let y be the optimal flow of the transportation problem where $y(s, t)$ the flow between $s \in \mathcal{F}^+$ and $t \in \mathcal{F}^-$. We denote $y(s, V^-) = \sum_{t \in V^-} y(s, t)$ and $y(V^+, t) = \sum_{s \in V^+} y(s, t)$ for $V^- \subset \mathcal{F}^-$ and $V^+ \subset \mathcal{F}^+$, respectively. For special cases where $V^- = \mathcal{F}^-$ and $V^+ = \mathcal{F}^+$, we simply denote $y(\cdot, t)$ as $y(\mathcal{F}^+, t)$ and $y(s, \cdot)$ as $y(s, \mathcal{F}^-)$ the total flows received at t and the total flow sent from s, respectively. Hence, for $s \in \mathcal{F}^+$ and $t \in \mathcal{F}^-$ $y(\cdot, t) = u_t^* - u_t$ and $y(s, \cdot) = u_s - u_s^*$.

In the remaining, we will give a feasible transfer which closes each facility in $\mathcal{F}^+$ once, opens each facility in $\mathcal{F}^+$ at most three times and the transportation

cost of the transfer is also bounded by twice that of the optimal flow y. The transfer scheme follows the same scheme in [3] with `Open-close` as the main operation. For completeness, we present the schema in the following.

Consider a facility t of $\mathcal{F}^-$ and the subtree T_t. Note that $K(t) = T_t \cap \mathcal{F}^+$. We will classify the facilities into groups indicating where the main part of the sent (received) flow amount goes to (comes from). We say that a facility $t \in \mathcal{F}^-$ is *strong* if $y(\cdot, t) \geq 2y(K(t), t)$; and *weak* otherwise. Intuitively, t is strong means that the main part of flow that t receives comes from its parent. Similarly, a facility $s \in \mathcal{F}^+$ is *dominant* if $y(s, t) \geq y(s, K(s))$; and *non-dominant* otherwise. Again, intuitively a dominant facility routes out the main part of its out-flow to its parent. In the tree T_t, let $\mathtt{Dom}(t)$ and $\mathtt{NDom}(t)$ be the sets of dominant and non-dominant facilities of $K(t)$, respectively. For each facility $s \in K(t)$, let $S(s)$ and $W(s)$ be the set of strong and weak facilities in $K(s)$, respectively.
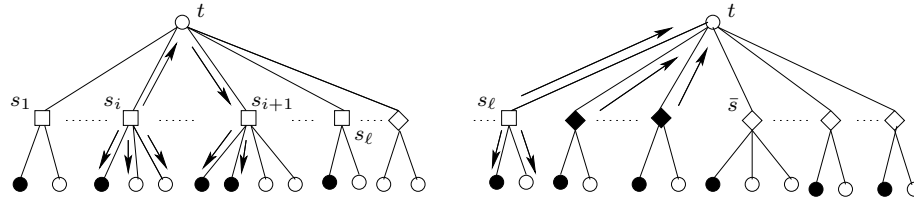
Fig. 1. In the figures, the squares and diamonds represent non-dominant and dominant facilities in $\mathcal{F}^+ \cap T_t$, respectively. Besides, the circles and black circles represent weak and strong facilities in $\mathcal{F}^- \cap T_t$, respectively. The figure in the left illustrates operation $\mathtt{Close}(s_i, y(s_i, \cdot))$ in Step 1. The figure in the right illustrates operation $\mathtt{Open\text{-}close}(t, s_\ell, y(s_\ell, t) + \sum_{s \in \mathtt{Dom}_2} y(s, \cdot), y(s_\ell, \cdot))$ in Step 2c where black diamonds stands for facilities in $\mathtt{Dom}_2$.

Consider the transfer by performing the following operations. (An illustration is shown in Figure 1.)

1. For each facility $s \in \mathtt{NDom}(t)$, define $\mathtt{Rem}(s) := \max\{y(s, t) - y(s, W(s)), 0\}$. Order facilities in $\mathtt{NDom}(t)$ as $\{s_1, \dots, s_\ell\}$ such that $\mathtt{Rem}(s_1) \leq \dots \leq \mathtt{Rem}(s_\ell)$. For $i \in \{1, \dots, \ell - 1\}$, consider $\mathtt{Close}(s_i, y(s_i, \cdot))$: decrease the capacity at s_i by moving out $2y(s_i, t')$ units to every facility $t' \in W(s_i)$, $y(s_i, t')$ units to every facility $t' \in S(s_i)$ and $\mathtt{Rem}(s_i)$ units to facilities in $S(s_{i+1})$. The latter must be distributed in such a way that each facility in $t' \in S(s_{i+1})$ receives at most $y(s_{i+1}, t')$ units. That can always be done since $\mathtt{Rem}(s_i) \leq \mathtt{Rem}(s_{i+1}) \leq y(s_{i+1}, K(s_{i+1})) - y(s_{i+1}, W(s_{i+1})) = y(s_{i+1}, S(s_{i+1}))$ where the second inequality is because s_{i+1} is non-dominant.
2. We adopt different procedures depending on different cases.
   a. **If t is strong.**
      Consider $\mathtt{Open\text{-}close}(t, s_\ell, y(s_\ell, t) + \sum_{s \in \mathtt{Dom}(t)} y(s, \cdot), y(s_\ell, \cdot))$: facility t receives $y(s, \cdot)$ units from each facility $s \in \mathtt{Dom}(t)$ in addition with $y(s_\ell, t)$ units from s_ℓ; send out $y(s_\ell, t')$ units from s_ℓ to every facility $t' \in K(s_\ell) \cup t$.

b. **If t is weak and there is a facility $h \in \texttt{Dom}(t)$ such that $y(h, t) \geq y(\cdot, t)/2$.**
   - Consider $\texttt{Close}(h, y(h, \cdot))$: decrease the capacity at h by sending out $y(h, t')$ units to every facility $t' \in K(h) \cup t$.
   - Consider $\texttt{Open-close}(t, s_\ell, y(s_\ell, t) + \sum_{s \in \texttt{Dom}(t), s \neq h} y(s, \cdot), y(s_\ell, \cdot))$: facility t receives $y(s, \cdot)$ units from each facility $s \in \texttt{Dom}(t) \setminus \{h\}$ in addition with $y(s_\ell, t)$ units from s_ℓ; send out $y(s_\ell, t')$ units from s_ℓ to every facility $t' \in K(s_\ell) \cup t$.

c. **If t is weak and there is no facility $h \in \texttt{Dom}(t)$ such that $y(h, t) \geq y(\cdot, t)/2$.**
   In this case, Zhang et al. [9] has proved that there exist a facility $\bar{s} \in \texttt{Dom}(t)$ such that the set $\texttt{Dom}(t) \setminus \{\bar{s}\}$ can be partitioned into $\texttt{Dom}_1$ and $\texttt{Dom}_2$ satisfying $y(\bar{s}, t) + \sum_{s \in \texttt{Dom}_1} y(s, \cdot) \leq y(\cdot, t)$ and $y(s_\ell, t) + \sum_{s \in \texttt{Dom}_2} y(s, \cdot) \leq y(\cdot, t)$.
   - Consider $\texttt{Open-close}(t, \bar{s}, y(\bar{s}, t) + \sum_{s \in \texttt{Dom}_1} y(s, \cdot), y(\bar{s}, \cdot))$: facility t receives $y(s, \cdot)$ units from each facility $s \in \texttt{Dom}_1$ in addition with $y(\bar{s}, t)$ units from $\bar{s}$; send out $y(\bar{s}, t')$ units from $\bar{s}$ to every facility $t' \in K(\bar{s}) \cup t$.
   - Similarly, consider $\texttt{Open-close}(t, s_\ell, y(s_\ell, t) + \sum_{s \in \texttt{Dom}_2} y(s, \cdot), y(s_\ell, \cdot))$: facility t receives $y(s, \cdot)$ units from each facility $s \in \texttt{Dom}_2$ in addition with $y(s_\ell, t)$ units from s_ℓ; send out $y(s_\ell, t')$ units from s_ℓ to every facility $t' \in K(s_\ell) \cup t$.

Lemma 3. *The transfer is feasible.*

Proof. Fix a tree T_t. By the transfer procedure, each facility $s \in T_t \cap \mathcal{F}^+$ sends out exactly $y(s, \cdot)$ units. It remains to prove that every facility $t' \in T_t \cap \mathcal{F}^-$ receives each time at most $y(\cdot, t')$.

Consider $t' \in T_t \cap \mathcal{F}^- \setminus \{t\}$ and suppose that $t' \in K(s)$ for some $s \in T_t \cap \mathcal{F}^+$. By the transfer on tree T_t, at any operation, if t' is weak then it receives at most $2y(s, t')$, which is bounded by $y(\cdot, t')$; otherwise (if t' is strong) the received amount is at most $y(s, t')$. In any case, t' receives at most $y(\cdot, t')$.

Consider the root t of tree T_t. Note that facility t only receives flow in Step 2 of the procedure. If t is strong, the total amount sent to t is

$$y(s_\ell, t) + \sum_{s \in \texttt{Dom}(t)} y(s, \cdot) \leq y(s_\ell, t) + \sum_{s \in \texttt{Dom}(t)} 2y(s, t) \leq 2y(K(t), t) \leq y(\cdot, t)$$

where the first and the last inequalities follow by the definition of dominant facilities and the fact that t is strong, respectively.

If t is weak and there is facility $h \in \texttt{Dom}(t)$ such that $y(h, t) \geq y(\cdot, t)/2$. Facility t receives $y(h, t)$ units in the operation that closes facility h. In the operation that opens facility t and closes facility s_ℓ, the amount sent to t is

$$y(s_\ell, t) + \sum_{s \in \texttt{Dom}(t), s \neq h} y(s, \cdot) \leq y(s_\ell, t) + \sum_{s \in \texttt{Dom}(t), s \neq h} 2y(s, t)$$
$$\leq 2y(K(t) \setminus \{h\}, t) \leq 2(y(\cdot, t) - y(h, t)) \leq y(\cdot, t).$$

If t is weak and there is no facility $h \in \mathtt{Dom}(t)$ such that $y(h,t) \geq y(\cdot,t)/2$. The amounts that i receives are $y(\bar{s},t) + \sum_{s \in \mathtt{Dom}_1} y(s,\cdot)$ and $y(s_\ell,t) + \sum_{s \in \mathtt{Dom}_2} y(s,\cdot)$ by the first and second operations of this case, respectively. As mentioned in the transfer procedure, those amounts are both bounded by $y(\cdot,t)$. Hence, in any case, t receives at most $y(\cdot,t)$ units at each operation. $\qquad\square$

Lemma 4. *In the transfer, each facility in $\mathcal{F}^+$ is closed exactly once, each facility in $\mathcal{F}^-$ is opened at most three times and the flow across every edge $e = (s,t)$ in the forest (the support graph of the optimal solution in the transportation problem) is at most $2y(s,t)$.*

Proof. Clearly by the decomposition of the forest into subtrees, a facility in $\mathcal{F}^+$ belongs to exactly one subtree and a facility in $\mathcal{F}^-$ belongs to at most two subtrees T_t. Hence, by the previous lemma, every facility in $\mathcal{F}^+$ is closed exactly once. Consider a facility $t \in \mathcal{F}^-$. Let $\tilde{t}$ be the grand parent of t. If t is strong then it will be opened once by the transfer procedure on tree T_t and it is opened at most twice during the procedure on tree $T_{\tilde{t}}$. If t is weak then it will be opened at most twice by the procedure on tree T_t and once by the procedure on tree $T_{\tilde{t}}$. In any case, t is opened at most three time in the transfer.

Observe that edges in the subtrees T_t for all t are disjoint, so the flow across an edge e in the transfer is the one across the edge in the transfer restricted on the subtree containing e. Consider an edge e in a subtree T_t and the following cases.

Case 1: $e = (s,t)$ where $s \in T_t \cap \mathcal{F}^+$. If s is dominant, then the total flow routed along (s,t) is at most $y(s,\cdot)$, which is bounded by $2y(s,t)$. If s is non-dominant, suppose that $s = s_i$ for some $1 \leq i \leq \ell$ where $\mathtt{NDom(t)} = \{s_1, \ldots, s_\ell\}$ ordered according to the $\mathtt{Rem}$ functions. The total flow across (s_i, t) is due to: (1) the operation closing s_{i-1}; and (2) the one closing s_i (especially for s_ℓ, that is an operation opening t and closing s_ℓ). The first operation sends through (s_i, t) a flow $\mathtt{Rem}(s_{i-1}) \leq \mathtt{Rem}(s_i) \leq y(s_i, t)$; the second operation routes along (s_i, t) a flow at most $y(s_i, t)$. Therefore, the total flow across e is at most $2y(e)$.

Case 2: $e = (s,t')$ where $s \in T_t \cap \mathcal{F}^+$ and $t' \in T_t \cap \mathcal{F}^- \setminus \{t\}$. If $s \in \mathtt{Dom}(t)$ then by the transfer procedure the flow routed through (s,t') is at most $y(s,t')$. If $s = s_i$ is non-dominant and t' is weak then the flow across (s,t') is either $2y(s,t')$ in case $s \neq s_\ell$ or $y(s,t')$ in case $s = s_\ell$. If $s = s_i$ is non-dominant and t' is strong then the flow sending from s_{i-1} to t' is at most $y(s_i, t')$. So together with the flow $y(s_i, t')$ routing from s_i, the total flow across (s_i, t') is at most $2y(s_i, t')$.

In summary, the flow across every edge e in the forest is at most $2y(e)$. The lemma follows. $\qquad\square$

Theorem 1. *It holds that $C_f(S) \leq 4C_s(S^*) + 5C_f(S^*)$ and $C_s(S) \leq C_s(S^*) + C_f(S^*)$. Consequently, the local search algorithm is $(3 + 2\sqrt{2} + \epsilon)$-approximation ($\approx (5.83 + \epsilon)$-approx).*

Proof. By the previous lemma, the transportation cost of the transfer is at most $2(C_s(S)+C_s(S^*))$ and a facility of $\mathcal{F}^-$ is opened at most three times. Therefore,

$$C_f(S) \leq 3C_f(S^*) + 2(C_s(S) + C_s(S^*)) \leq 4C_s(S^*) + 5C_f(S^*)$$

since $C_s(S) \leq C_s(S^*) + C_f(S^*)$. Hence, the total cost of solution S is

$$C_s(S) + C_f(S) \leq C_s(S^*) + C_f(S^*) + 4C_s(S^*) + 5C_f(S^*) = 5C_s(S^*) + 6C_f(S^*)$$

which yields an approximation ratio $(6 + \epsilon)$.

By a standard scaling technique, we multiply the cost function of every facility by a factor λ (to be defined later) then apply the algorithm to this instance. Let $C'_f(\cdot)$ and $C_f(\cdot)$ be the facility cost function of the modified and original solution. Note that $C'_f(\cdot) = \lambda C_f(\cdot)$. Let S a local optimum (on the modified instance) and S^* be any feasible solution. The same analysis leads to the following inequalities: $C'_f(S) \leq 4C_s(S^*) + 5C'_f(S^*)$ and $C_s(S) \leq C_s(S^*) + C'_f(S^*)$. Therefore, the original total cost

$$C_f(S) + C_s(S) = C'_f(S)/\lambda + C_s(S) \leq \frac{1}{\lambda}(4C_s(S^*) + 5C'_f(S^*)) + C_s(S^*) + C'_f(S^*)$$

$$= (5 + \lambda)C_f(S^*) + (\frac{4}{\lambda} + 1)C_s(S^*).$$

Choosing $\lambda = 2\sqrt{2} - 2$, we get $C(S) \leq (3 + 2\sqrt{2})C(S^*)$. Therefore, the approximation ratio is $(3 + 2\sqrt{2} + \epsilon) \approx (5.83 + \epsilon)$. $\square$

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 J. Computing*, 33(3):544–562, 2004.
2. M. Bansal, N. Garg, and N. Gupta. A 5-approximation for capacitated facility location. In *Proc. 20th European Symposium on Algorithms (ESA)*, pages 133–144, 2012.
3. N. Garg, R. Khandekar, and V. Pandit. Improved approximation for universal facility location. In *Proc. 16th Symposium on Discrete Algorithms*, pages 959–960, 2005.
4. M.T. Hajiaghayi, M. Mahdian, and V.S. Mirrokni. The facility location problem with general cost functions. *Networks*, 42(1):42–47, 2003.
5. J. Li and S. Khuller. Generalized machine activation problems. In *Proc. of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 80–94, 2011.
6. M. Mahdian and M. Pál. Universal facility location. In *Proc. 11th Annual European Symposium on Algorithms (ESA)*, pages 409–421, 2003.
7. M. Pál, É. Tardos, and T. Wexler. Facility location with nonuniform hard capacities. In *Proc. 42nd Annual Symposium on Foundations of Computer Science*, pages 329–338, 2001.
8. J. Vygen. From stars to comets: Improved local search for universal facility location. *Operations Research Letters*, 35(4):427–433, 2007.

9. J. Zhang, B. Chen, and Y. Ye. A multiexchange local search algorithm for the capacitated facility location problem. *Mathematics of Operations Research*, 30(2):389–403, 2005.