On Multi-Level Network Facility Location Problem

Edward Kh. Gimadi[1,2] and Yuri V. Shamardin[1]

[1] Sobolev Institute of Mathematics SB RAS, Novosibirsk, Russia
[2] Novosibirsk State University, Novosibirsk, Russia
gimadi@math.nsc.ru, orlab@math.nsc.ru

Abstract. The article is intended to fill the recent review of Ortiz-Astorquiza, C. at al. (2017) on multi-level facility location problem (MLFLP). The article presents the results of some publications, information about which is missing in this review. We are talking about the construction of polynomial exact algorithms for solving some subclasses of the network MLFLP. Namely, the design of polynomial time algorithms for the multi-level FLP on a chain graph and the two-level FLP on a tree graph are discussed. We also show that the a known result of Trubin V. A. and Sharifov F.A. (1992) for the general multi-level FLP on a tree is incorrect.

Keywords: Multi-level FLP · Network · Chain · Tree · Exact algorithm · Polynomial time

1 Introduction

In the multi-level network facility location problem (MLFLP) we are given a set of customers that have some product demands and a set of potential facilities partitioned into p levels. The problem is to open a collection of facilities, such that the customers are assigned to one or multiple sequences of opened facilities, one from each level $p, p-1, \ldots, 1$, while minimizing the total transportation cost and the cost of opening the chosen facilities.

The most common application of the MLFLP is the design of a production-distribution system, where the distribution of a product is for example handled through the system of production plants, warehouses, and retailers [7]. Another popular application field refers to the telecommunication systems and network design, where it is required to effectively connect the terminals into a network building a system of routers and multiplexers [9].

The MLFLP in general is NP-hard even in the case of the one-level setting [1]. In 1977 Kaufman, Eede, and Hansen in [7] first introduced the two-level FLP

In: S. Belim et al. (eds.): OPTA-SCL 2018, Omsk, Russia, published at http://ceur-ws.org

as a warehouse and plant location problem. Since then, the generalizations and modifications of the problem have been extensively studied, various heuristics and approximation algorithms are designed as well as the exact approaches such as branch-and-bound [7, 10, 12] and branch-and-cut [8] methods. We refer the reader to the recent survey paper [9] for a detailed overview of this kind of algorithmic results for the MLFLP. The purpose of the article is to supplement this review with some of the results of earlier publications not mentioned therein. In addition, we recall the advisability of a more compact formulation of the MLFLP using the so-called assignment vectors as decision variables.

Let's give a mathematical formulation for the p-level Network MLFLP. In the network statement of the MLFLP it is assumed that we are given a weighted graph $G(N, E)$, where the set of nodes $N = \{1, \dots, n\}$ represents the set of customers; $M_r \subset N$ is the set of possible r-th level facility location sites, $m_r = |M_r|, \quad 1 \leq r \leq p$; and the set E of the weighed edges represents the transportation network. We assume that the beginning of the production process takes place at the level p, further a facility of the level r receives the product from one or multiple number of facilities of the level $(r+1)$, $1 \leq r \leq p-1$, and finally the product is supplied to the customers on the level 0. Denote by

g_i^r the cost of opening an r-th level facility at node i;

b_j the demand at node j;

$d_{i_p,\dots i_1,j}$ the total cost of the shortest path transportation of product unit from the facility of the level p located at node i_p through the sequence of the facilities of the lower levels $p-1, \dots, 1$ at nodes $i_{p-1}, \dots, i_1$ to the customer at node j.

The MLFLP then can be formulated as:

$$\sum_{j \in N} b_j \sum_{i_p \in M_p} \dots \sum_{i_1 \in M_1} x_{i_p \dots i_1 j} d_{i_p \dots i_1 j} + \sum_{r=1}^{p} \sum_{i \in M_r} g_i^r y_{ri} \to \min \tag{1}$$

subject to

$$\sum_{i_p \in M_p} \dots \sum_{i_1 \in M_1} x_{i_p \dots i_1 j} = 1, \ j \in N, \tag{2}$$

$$\sum_{i_p \in M_p} \dots \sum_{i_{r-1} \in M_{r-1}} \sum_{i_{r+1} \in M_{r+1}} \dots \sum_{i_1 \in M_1} x_{i_p \dots i_1 j} \leq y_{r i_r}, \tag{3}$$
$$j \in N, \ 1 \leq r \leq p, \, i_r \in M_r,$$

$$0 \leq x_{i_p \dots i_1 j} \leq 1, \ j \in N, \, i_1 \in M_1, \, \dots, \, i_p \in M_p, \tag{4}$$

$$y_{ir} \in \{0, 1\}, \ \in M_r, \, 1 \leq r \leq p. \tag{5}$$

Here variables $x_{i_p \dots i_1 j}$ are the allocation variables that stand for the proportion of product transported to the customer at node j through the sequence of facilities of levels $p \dots 1$ at nodes $i_p, \dots i_1$. The variables y_{ri} are the location variables, where $y_{ir} = 1$, if a facility of level r is opened at node i, and $y_{ir} = 0$

otherwise. The first sum in (1) corresponds to the total transportation costs, and the second sum is the total cost of opening the chosen facilities. Constrains (2) stand for each consumer's demand is satisfied, while constrains (3) make sure that the allocation to the facility of the level r at node i_r is possible only if it is open. If the allocation variables satisfy (4), each customer can be served by multiple sequences of opened facilities, and the problem is called multiple allocation MLFLP. If the allocation variables $x_{i_p \ldots i_1 j} \in \{0, 1\}$, each customer can be served by one sequence of opened facilities, and the problem is called single allocation MLFLP.

For the single allocation MLFLP there is a more compact formulation that uses the assignment vectors as variables. The similar formulation was introduced for the one-level facility location problem [3]. Let's use the following notation:

- $\pi^r = (\pi_1^r, \ldots, \pi_n^r)^T$ is the facilities assignment vector on the level r, where
- $\pi_j^r \in M_r$ is the node in which the facility of the level r serving customer j is placed.
- $\pi = (\pi^1, \pi^2, \ldots, \pi^r)$ is the feasible solution of the problem;
- $I^r(\pi) = \bigcup_{j \in N} \{ \pi_j^r \}$ is the set of facilities of the level r opened in the solution π;
- $Y_i^r(\pi)$ is the service area of the facility i of the level r, i.e., the union over all j such that $\pi_j^r = i$, where $i \in M_r$ and $1 \le r \le p$.

It is clear that $I^r(\pi) \subset M_r$ and $\bigcup Y_i^r(\pi) = N$, where the union is taken over all $i \in I^r(\pi),\ 1 \le r \le p$.

Thus, the single allocation MLFLP can be stated as follows:

$$\sum_{r=1}^{p} \sum_{i \in I^r(\pi)} g_i^r + \sum_{j \in N} b_j \sum_{r=1}^{p} d_{\pi_j^r \pi_j^{r-1}} \to \min_{(\pi_j^r)}, \tag{6}$$

where d_{ij} is the length of the shortest path between nodes i and j in G.

In the following sections we consider it appropriate to recall some of the results in [4–6] for the Network MLFLP on the line and the tree graphs, and give some comments on Trubin's article [13].

2 Some Known Facts about the Polynomial Solvability of the Chain MLFLP

In this section we are going to recall some facts about a special case of the single allocation Network MLFLP, where the given network is a chain (path). The transposition cost d_{ij} of the product unit between nodes i and j is the sum of the lengths of the edges in the subchain connecting these nodes.

2.1 An Exact Algorithm A_p for the Chain MLFLP

The first algorithm A_p for the Chain MLFLP is based on the dynamic programming scheme under consideretion the original problem as $\Big\langle (M_r),\, 1 \le r \le p;\, N \Big\rangle$ and the family of subproblems:

$$\Big\{ \Big\langle\, (M_r^{i_r}),\, i_r \in M_r;\, [1,j] \Big\rangle \Big|\, i_r \in M_r,\, 1 \le r \le p; \quad 1 \le j \le n \Big\},$$

where $M_r^s = [1,s] \cap M_r,\, s \in M_r$, and $1 \le r \le p$.

Denote by $L_j(u)$ the optimal value of the objective function (the optimum) of each defined subproblem, where $u = (i_1, i_2, \ldots, i_p)$, and let $F_j^s(u), 1 \le s \le p$, be the optima of the subproblems $\Big\langle\, (M_r^{i_r}), r \in [1,p];\, [1,j] \Big|\, \pi_j^r = i_r,\, s \le r \le p \Big\rangle$. It is clear, that $F_1(u) = \sum\limits_{r=1}^{p} (g_{i_r}^r + b_1\, c_{i_r i_{r-1}})$, where $i_0 = j$, where we set g_i^r equal to ∞ if $i \notin M_r, 1 \le r \le p$. It is easy to see that the optimum F^* of the original problem $\Big\langle\, M_r,\, 1 \le r \le p;\, N \Big\rangle$ is equal to

$$F^* = \min \Big\{ F_n^1(u) \Big|\, i_r \in M_r, 1 \le r \le p \Big\}.$$

Statement 1 [5]. *The Chain MLFLP can be solved in $O(pnm_1 m_2 \ldots m_p)$-time using the following recurrent relations:*

$$F_j^{s+1}(u) = \min \Big\{ F_j^s(u); F_j^{s+1}(u - e^s) \Big\},$$

$$F_j^1(u) = \sum_{r=1}^{p} (g_{i_r}^r + b_j\, c_{i_r i_{r-1}}) + \min_{1<s\le p+1} \Big\{ F_{j-1}^s(u - h^s) - \sum_{r=s}^{p} g_{i_r}^r \Big\},$$

where $F_j^{p+1}(u) = L_j(u)$, e^s is a p-dimensional s-th orth, $h^s = \sum\limits_{r=1}^{s-1} e^r, 1 \le s \le p$.

Thus, the algorithm has time complexity linear in the number of the customers n and exponential in the number of levels p. Moreover, the bounds on time and space complexities coincide.

2.2 A Polynomial-time Algorithm $\widetilde{A}_p$ for the Chain MLFLP

Another exact Algorithm $\widetilde{A}_p$ for the Chain MLFLP makes essential use of the inclusion property of optimal solutions and of reduction to a special series of the Nearest Neighbor Problems (NNP).

In the NNP we are given an integer segment $(0,\, n]$ and a the cost function $f(x,y)$ of serving each segment $[x,y]$, $0 \le x, y, \le n$. The problem is:

$$\sum_{s=1}^{m} f(x_{s-1}, x_s) \to \min_{0=x_0<\ldots<x_m=n}$$

subject to $1 \le m \le n$.

Statement 2 [5]. *The time complexity of Algorithm $\widetilde{A}_p$ for solving the Chain MLFLP is $O\big(n^3 \sum_{r=1}^{p} m_r\big)$.*

Comparing algorithm A_p with running-time $O(nm_1m_2 \dots m_p)$ and algorithm $\widetilde{A}_p$ with running-time $O\big(n^3 \sum_{r=1}^{p} m_r\big)$, it follows that the algorithm A_p runs faster than $\widetilde{A}_p$, if

$$\frac{pm_1 \dots m_p}{m_1 + \dots + m_p} \le n^2.$$

Let $m = \max\{m_r \mid 1 \le r \le p\}$. Then for the time complexities of A_p and $\widetilde{A}_p$ we have the upper bounds $O(pnm^p)$ and $O(p\,m\,n^3)$, respectively. It is clear that in the case of two- and three-level FLPs, the algorithm A_p is more efficient than A_p, but for the problem with the number of levels $p > 3$ the algorithm $\widetilde{A}_p$ is preferable. Also note, that the space complexity of the algorithm A_p is exponential in the number of levels, while the space complexity of the algorithm $\widetilde{A}_p$ stays polynomial.

3 Multi-level FLP on a Tree

3.1 An Exact Algorithm for the Tree 2-Level FLP

The question on existence of exact polynomial time algorithms for solving the Tree p-level FLP for $p \ge 3$ remains open. Nevertheless, paper [6] gives an exact polynomial-time algorithm solving the Tree 2-level FLP based on the dynamic programming procedure. Let $G = (N, E)$ be a tree network with the set N of nodes and the set E of edges, $|\,E\,| = n - 1$. Let the node 1 be the root of this tree. For all $j \in N$, let :
P_j be a simple path from the root 1 to the node j;
$N_j = \{j' \mid j \in P_{j'}, j' \in N\}$;
$I^r_j(\pi) = \bigcup\{\,\pi^r_{j'} \mid j' \in N_j\}$, $1 \le r \le p$;
$\mu_j(\pi) = \arg\min\{c_{kj} \mid k \in I^2(\pi)\}$.

Statement 3 [6]. *There exists an optimal solution of the 2-level FLP on a tree, such that for all $j \in N$, the following inclusions*

$$I^1_j(\pi) \subset N_j \cup \{\pi^1_j\},$$

$$I^2_j(\pi) \subset N_j \cup \{\pi^2_j\} \cup \{\mu_j(\pi)\}$$

hold.

Let $\langle M_1,\, M_2;\, N\rangle$ be the original 2-level FLP on a tree. Consider the family of the following subproblems:

$$\big\{\langle\, M_1,\, M_2;\, N_j\rangle \mid \pi^1_j = i,\, \pi^2_j = k; \mu_j(\pi) = k',\, i \in M_1,\, k, k' \in M_2,\, 1 \le j \le n\big\}.$$

Denote by $F_j(i, k, k')$ the optimal value of the objective function of each defined subproblem.

Statement 4 [6]. *The Two-level FLP on a tree can be solved in $O(nm_1m_2^2)$-time using for all $j \in N$, $i \in M_1$ and $k, k' \in M_2$, the following recurrent relations:*

$$F_j(i,k,k') = g_i^1 + g_{kk'}^2 + b_j(c_{ki} + c_{ij}) +$$

$$+ \sum_{l \in S_j} \min\{F_l; F_l(k') - g_{k'}^2; F_l(k,k') - g_{kk'}^2; F_l(i,k,k') - g_i^1 - g_{kk'}^2\},$$

where $F_j(k,k') = \min_{i \in M_1} F_j(i,k,k')$, $F_j(k') = \min_{k \in M_2} F_j(k,k')$, $F_j = \min_{k' \in M_2} F_j(k')$, $g_{kk}^2 = g_k^2$ *and* $g_{kk'}^2 = g_k^2 + g_{k'}^2$ *for* $k \neq k'$. *Note that* $F^* = F_1$.

3.2 On Trubin's Result for the Tree MLFLP

In paper [13] the multiple allocation MLFLP with p levels on a tree network was studied. The authors claimed that the MLFLP on an n-vertex tree T_1 can be reduced to a Simple Facility Location Problem (SFLP) on an n^p-vertex tree T_2. Since the latter problem can be solved in polynomial time, the authors claimed that the MLFLP on a tree with a fixed number of levels p possesses a polynomial time algorithm as well. Thus, for example, using an $O(n \log n)$ algorithm [11] for the SFLP, one can obtain an $O(n^p \log n)$ algorithm for the MLFLP on a tree.

The aim of this section is to prove that the reduction from the MLFLP to the SFLP presented in [13] is incorrect.

The reduction proposed in [13] maps the MLFLP on tree T_1 to the following FLP on T_2, which consists of several connected copies of the initial tree T_1 (Fig. 1). The tree T_2 is built in p steps. At the fist step $T_2 = T_1$. After $k-1$ steps T_2 consists of vertices v_i numbered as $(i_1, \ldots, i_{k-1})$. At step k to each vertex $v_i = (i_1, \ldots, i_{k-1})$ of T_2 we attach the initial tree T_1 rooted at vertex i_{k-1}. Each new vertex j from the attached tree T_1 is numbered as $(i_1, \ldots, i_{k-1}, j)$ in the constructed tree T_2. Opening a facility in the p-vector numerated vertex $(i_1, \ldots, i_p)$ of T_2 in the FLP corresponds to opening a facility of the level 1 at the vertex i_1, a facility of the level 2 at the vertex i_2 and so on in the tree T_1 of the initial 2-level FLP. Thus the vertices of T_2 enumerates all possible sequences of opened facilities on the p levels in the original problem. The customers of the FLP on tree T_2 having their original demand are located in vertices $(i, i, \ldots, i)$, $1 \leq i \leq n$, in the Fig. 1 these vertices are $(1,1)$, $(2,2)$ and $(3,3)$. The cost $g_{i_1,\ldots,i_p}$ of opening a facility a vertex in T_2 is said to be equal to the sum

$$g_{i_1,\ldots,i_p} = g_{i_1}^1 + g_{i_2}^2 + \ldots + g_{i_p}^p \tag{7}$$

where g_i^r is the cost of opening an r-th level facility, at vertex i in the original problem on thetree T_1.

Let's introduce variables $z_{i_1,\ldots,i_p}$ equal to 1, if we open a facility at vertex $(i_1, \ldots, i_p) \in T_2$, and 0, otherwise. The authors [13] claim that the SFLP on T_2, where the total cost of opening facilities calculates as

$$\sum_{i_1 \in M_1} \cdots \sum_{i_p \in M_p} g_{i_1,\ldots,i_p} z_{i_1,\ldots,i_p}$$

is equivalent to the problem (1)-(5) on T_1. The incorrectness of this claim follows from the

Counterexample. Consider the following 2-level FLP (1)-(5) on the tree T_1 in the form of a line (path) with the set of vertices $\{1, 2, 3\}$ and the set of edges $\{\{1, 2\}, \{2, 3\}\}$ (Fig. 1). Let $a_1 = 2$ and $a_2 = 3$ be the cost of transportation of a product unit along the edges $\{1, 2\}$ and $\{2, 3\}$, respectively. Let d_{ij} be the total cost of transportation of a product unit along the simple path between vertices i and j in T_1, $i, j = 1, 2, 3$. Set the demand $b_1 = b_2 = b_3 = 1$, and the following costs:

— of opening second-level facilities: $g^2_1 = \infty$, $g^2_2 = 5$, $g^2_3 = \infty$;
— of opening first-level facilities: $g^1_1 = 1$, $g^2_2 = \infty$, $g^2_3 = 1$.

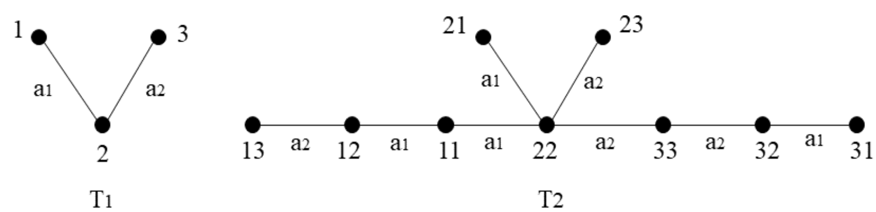

Fig. 1. The initial tree T_1 for the MLFLP and the constructed tree T_2 for the FLP

The location variables y_{ri} in the optimal solution for (1)-(5) on the tree T_1 obviously satisfy: $y_{21} = y_{23} = y_{12} = 0;\ y_{22} = 1;\ y_{11}, y_{13} \in \{0, 1\}$. Let $S(y_{11}, y_{13})$ be the total cost of a feasible solution for our the example of MLFLP on T_1. Then

$$S(1, 0) = a_1b_1 + 2a_1b_2 + (2a_1 + a_2)b_3 + g^2_2 + g^1_1 = 19.$$

Similarly, it is easy to see that $S(1, 1) = 16, S(0, 1) = 23$. Thus the optimal solution is opening a second level facility at vertex 2 and first level facilities at vertices 1 and 3. The optimal value S^* of the objective function is 16.

Now let's consider the SFLP on tree T_2 obtained from the two-level FLP on T_1. It is clear that $z_{12}, z_{32} \in \{0, 1\}$ and $z_{ij} = 0$ for all other vertices $(i, j) \in T_2$. Let $Q(z_{12}, z_{32})$ be the total transportation and facility opening cost for a solution of the considered example on T_2. Thus:

$$Q(1, 0) = g_{12} + a_1b_1 + 2a_1b_2 + (a_2 + 2a_1)b_3 = 19,$$

$$Q(0, 1) = g_{23} + (a_1 + a_2)b_1 + a_2b_2 + 2a_2b_3 = 23,$$

$$Q(1, 1) = g_{12} + g_{23} + a_1b_1 + b_2 \min\{2a_1, a_2\} + b_3 \min\{2a_1 + a_2, 2a_2\} = 21.$$

Thus the optimum Q^* of the problem on T_2 is equal to 19, which is larger then S^*. The obtained optimal solution for the problem consists in opening a facility at vertex $(1,2) \in T_2$ which corresponds to opening a second level facility at vertex 2 and a first level facility at vertex 1 in the original problem on T_1.

The mistake in [13] obviously consists in setting the costs of opening the facilities for the problem on T_2 as (7), since in this case the costs of opening the facilities on higher levels of the initial problem may be summed up several times. This leads to different costs of the optimal solutions in these two problems. The mistake can be fixed as follows. Set for all $r = 1, \dots, p$

$$P_r(z) = \{i_r | z_{i_1,\dots,i_p} = 1, (i_1, \dots, i_p) \in T_2\}$$

,

and set the costs of opening a facility at vertex $(i_1, \dots, i_p) \in T_2$ as

$$\sum_{r=1}^{p} \sum_{i \in P_r(z)} g_i^r.$$

This way the initial problem (1)-(5) on the tree T_1 indeed is equivalent to the location problem on the tree T_2, but the latter one is not the Simple Facility Location Problem.

4 Conclusion

The purpose of this article was, in the first place, to recall some of the results of constructing exact algorithms for solving the Network MLFLP on the chain and the tree graphs that were missed in the recent large review of Ortiz-Astorquiza, C. at al. (2017) [9].

Secondly, it was important to show that the polynomial-time algorithm presented by Trubin and Sharifov [13] for solving the Tree MLFLP is incorrect even in the particular case of the line graph.

Thus, we can state that at the present time the question of the complexity status of the Tree MLFLP (for the number of levels greater than 2) remains open and is waiting for its solution.

Acknowledgement. This research was supported by Russian Science Foundation (project 16-11-10041)

References

1. Garey, M.R., Johnson, D.S.: Computers and Intractability. Freeman, San Francisco (1979)
2. Gimadi, E.Kh.: Choice of optimal scales in a class of location, unification, and standardization problems. Upravlyaemye Sistemy 6, 57–70. Novosibirsk (1970), (in Russian)
3. Gimadi, E. Kh.: The problem of distribution on a network with centrally connected service areas. Upravlyaemye Sistemy 25, 38–47. Novosibirsk (1984), (in Russian)

4. Gimadi, E.Kh.: Exact algorithm for some multi-level location problems on a chain and a tree. In: Oper. Research Proceedings. pp. 72–77. Springer-Verlag, Berlin (1997)
5. Gimadi, E.Kh.: Effective algorithms for solving the multi-level plant location problem. In: Operations Research and Discrete Analysis. Mathematics and Its Applications. vol. 391, pp. 51–69. Kluwer Academic Publishers, Springer, Dordrecht (1997)
6. Gimadi, E.Kh. , Kurochkin, A.A.: An effective algorithm for the two-stage location problem on a tree-like network. Journal of Applied and Industrial Mathematics 7(2), 177–186 (2013)
7. Kaufman, L., Eede, M.V., Haunsen, P.A.: A plant and warehouse location problem. Oper. Res. Quart. 28(3), 547–554 (1977)
8. Landete, M., Marin, A.: New facets for the two-stage uncapacitated facility location polytope. Computational Optimization and Applications 44(3), 487–519 (2009)
9. Ortiz-Astorquiza, C., Contreras, I., Laporte, G.: Multi-level facility location problems. European Journal of Operational Research, 1–15 (2017)
10. Ro, H., Tcha, D.: A branch and bound algorithm for the two-level uncapacitated facility location problem with some side constraints. European J. Oper. Res. 18(3), 349–358 (1984)
11. Shah, R., Farach-Colton, M.: Undiscretized dynamic programming: Faster algorithms for facility location and related problems on trees. In: Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms (SODA). pp. 108-115 (2002)
12. Tcha, D.W., Lee, B.: A branch and bound algorithm for the multi-level uncapacitated facility location problem. European J. Oper. Res. 18(1), 35–43 (1984)
13. Trubin, V.A., Sharifov, F.A.: The simplest multi-level location problem on a tree-like network. Kibernet. Sistem. Anal. 6, 128–135. Kiev (1992), (in Russian)