A Hybrid Nested Partitions Algorithm for Banking Facility Location Problems

Li Xia, Wenjun Yin, Jin Dong, Teresa Wu, Ming Xie, and Yanjia Zhao

***Abstract*—The facility location problem has been studied in many industries including banking network, chain stores, and wireless network. Maximal covering location problem (MCLP) is a general model for this type of problems. Motivated by a real-world banking facility optimization project, we propose an enhanced MCLP model which captures the important features of this practical problem, namely, varied costs and revenues, multitype facilities, and flexible coverage functions. To solve this practical problem, we apply an existing hybrid nested partitions algorithm to the large-scale situation. We further use heuristic-based extensions to generate feasible solutions more efficiently. In addition, the upper bound of this problem is introduced to study the quality of solutions. Numerical results demonstrate the effectiveness and efficiency of our approach.**

***Note to Practitioners*—This paper is motivated by a practical banking facility location problem. The problem is how to choose the facilities (bank branches) location in order to maximize the facility network's profits. It is a large-scale optimization problem in the real world. We formulate this problem with an extended MCLP model and apply a hybrid nested partitions algorithm. Our approach is efficient since it combines the mathematical programming and problem specific heuristic information. Practitioners who want to use this approach should pay attention to the utility of problem structure and model formulation. This approach is also applicable to other location problems, such as the retail chain stores, gas stations, city public facilities, and so on.**

***Index Terms*—Banking facility, maximal covering location problem, mixed integer programming, nested partitions algorithm.**

I. Introduction

Facility location is a critical problem for strategic planners in many industrial applications [1]. For example, for a retail chain opening a new outlet, a manufacturer choosing to position a warehouse, and a banker selecting locations for service points, there is a need to make good decisions for the location and allocation of facilities [12], [17]. The facility location problem prevails in the fiercely competitive business environment. In 2002, U.S. companies spent $910 billion, about 8.7% of the U.S. Gross Domestic Product (GDP), on business logistics systems [15]. In Singapore, the logistic and transportation industry sector contributed about 10.8% of the GDP in year 2003 [4]. The facility location plays an important factor on the efficiency of these expenses. Considering the importance of the problems, manufacturers and retailers like IBM and Wal-Mart have made great efforts to design efficient facility networks for product flow.

Manuscript received July 07, 2009; revised October 17, 2009; accepted January 27, 2010. Date of publication March 18, 2010; date of current version July 02, 2010. This paper was recommended for publication by Associate Editor L. Shi and Editor M. Zhou upon evaluation of the reviewers' comments.

L. Xia is with the IBM China Research Laboratory, Zhongguancun Software Park, Haidian District, Beijing 100193, China, and also with the Division of Mathematical and Computer Science and Engineering, King Abdullah University of Science and Technology, Jeddah 21534, Saudi Arabia (e-mail: xiali@tsinghua.org.cn).

W. Yin, J. Dong, and M. Xie are with the IBM China Research Laboratory, Zhongguancun Software Park, Haidian District, Beijing 100193, China.

T. Wu is with the Department of Industrial Engineering, Arizona State University, Tempe AZ 85287-5906, USA.

Y. Zhao is with the Center for Intelligent and Networked Systems, Department of Automation, Tsinghua University, Beijing 100084, China.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASE.2010.2043430

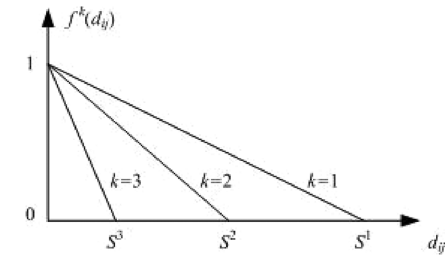

Fig. 1. An illustration of coverage function with $K = \{1, 2, 3\}$, where 1, 2, and 3 indicate the type I (large), type II (medium), and type III (small) facilities.

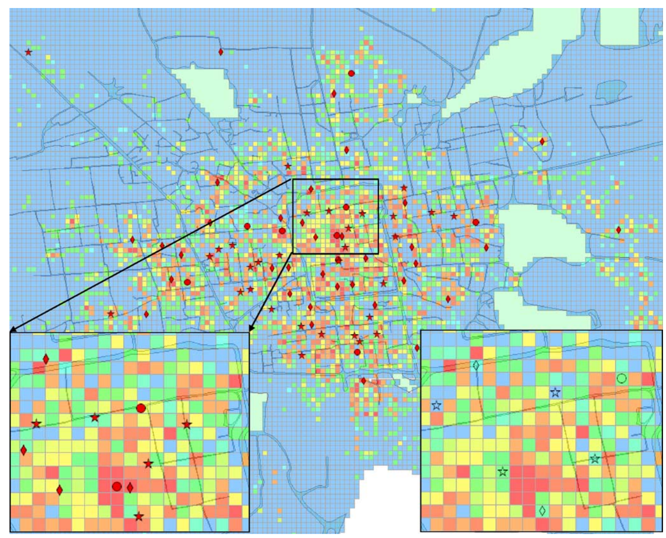

Fig. 2. The demand amount of cells and current locations of banks in the main city area of Suzhou, China, the left bottom picture is a zoomed particular area, while the right bottom picture is the optimized location of facilities in this area.

Some key issues associated with facility location decisions include cost reduction, fast response time, equitable service supply. Together, a model finds the maximum covered business potentials subject to some constraints. This model is called maximal covering location problem (MCLP), which has been proved to be one of the most useful facility location models from both theoretical and practical points of view [12], [17]. MCLP was originally introduced in early 1970's [5]. It has since then been applied to various practical studies, such as the emergency medical service location problem [7], [11] .

Motivated by an IBM project, we study a banking facility location problem where the bank branches are considered as facilities using MCLP model in this paper. In this situation, the location of banking facilities is a very critical decision since the investment of a banking facility is usually over millions of dollars. A wrong decision of branch location may induce serious investment losses. In this research, we study a scenario of a middle size city of 2 million populations in southeast China. The city region is discretized to more than 45,000 cells with each cell representing a 200 by 200 meters geographical area, partly illustrated in Fig. 2. All cells are considered as candidate facility locations, where the banking facilities (service points) may be sited. The possible revenue generated from each cell is a linear function of customer demands which are determined by the customer distribution in this cell region. For simplicity, we assume that the revenue of a cell is equal to the customer demand within that cell. The customer demand can be estimated based on Geographic Information Systems (GIS) data. The GIS data are provided by the professional data vendors. These GIS data include all kinds of points of interests, such as the residence, office building, supermarkets, etc. Moreover, the economic properties of points of interests are also included in the GIS data, such as the resident-number and sale prices of residence, the building area and rent prices of office buildings, etc. Based on the GIS data, the customer demand can be estimated which will be reported in a separate study. In this paper, we assume that the customer demand is known. The banking facilities have three types according to the facility size. Once one type of facility is set at one cell, we can calculate the revenue based on the demand information around this facility location, and the cost based on the operational and rental cost of the type of the facility. All the revenue and cost data are estimated based on our banking clients' financial information and we assume they are known. Thus, the banking facility location problem can be modeled as an MCLP where the objective is to optimally select the locations of facilities to obtain the maximum profit (revenue minus cost) with respect to certain constraints.

To apply MCLP to banking facility location problem we studied, some extensions of MCLP are required. First, the objective function of an MCLP is to maximize the covered demands by facilities, whereas in this case the objective is to maximize the profits produced by facilities. The profit considers both revenue and cost. The revenue is a function of the amount of covered demands within a certain range of facilities. The cost is a function of operational and rental cost which is dependent on the location and the type of the facility. The revenue and the cost both depend on the facility locations. Thus, a more generalized and complicated objective function should be developed. Second, an MCLP usually models the coverage function as a binary function, i.e., the demand of a cell is either fully covered (if it is inside the covering range of a certain facility), or not covered at all. To enhance the applicability, [3] explores a partial coverage of demands, with the degree of coverage being a nonincreasing step function of the distance to the nearest facility. To more realistically model the problem, we introduce linear decreasing function of distance as the coverage function. Third, unlike the traditional MCLP, we consider multiple types of facilities including type I (large), type II (medium), and type III (small) service facilities. Different types of facilities have different cost rates and coverage functions.

It is known that the MCLP can be solved using mixed integer programming (MIP). However, as the problem scale increases, the basic MIP algorithms may be inefficient. Considering the number of location variables of the problem we study is over 45,000, an efficient solution technique is required. Earlier researches have explored the linear programming (LP)-based relaxation of the mixed integer programming formulation [14]. Various heuristics, such as greedy-based approximation [20], decomposition [8], volume algorithm [2], and metaheuristic concentration [13], are developed to obtain faster computation times with near-optimal results. Lagrangian and surrogate relaxations-based heuristic and column generation are studied in [6], [18] and [9] for the MCLP. Yet, these algorithms are effective for the specific problems and may not be generalizable for a banking location problem we studied. Secondly, scalability remains a challenge.

To handle this large-scale optimization problem, we study nested partitions, a randomized method for solving global optimization problems. The nested partitions method was originally proposed by Shi and Olafsson [16], and it systematically partitions the feasible solution space and concentrates computation in the most promising regions. It is a method combining global search with local search. [16] presents the details of nested partitions, and proves that it converges to a global optimum in finite time. [10] presents a framework of hybrid algorithms combining the mathematical programming approach with nested partitions , which retains the capability of mathematical programming in handling constraints and the efficiency of nested partitions in obtaining approximate solutions. [15] applies nested partitions to an allocation and transportation problem in distributed supply chain networks and obtains optimal results within a reasonable amount of time. The literature indicates that nested partitions method has the promise to handle the large-scale optimization problems. In this paper, we apply the hybrid approach combining mathematical programming with the nested partitions algorithm [10] to the banking location problem. Additionally, some heuristic rules are embedded in the algorithm to address the scalability issue.

II. Problem Formulation

The case studied is from a project for one of the largest banks in a middle size city of southeast China. In the banking facility network, two kinds of participants are considered: the customers who ask for bank service and the bank who wants to maximize its profit (revenue minus cost). Assume the customers are located in the geographic area which can be discretized to a set of cells. A *demand node* is defined at each cell, representing the demand within the cell. The candidate locations to site banks are called *candidate facilities*, which are specific points in the cells. The mathematical formulation of this problem is described in the following section.

A. Formulation of the Banking Facility Location Problem

Sets and indices:

I, i	the set and index of demand nodes;
J, j	the set and index of candidate facility locations;
K, k	the set and index of facility types, $k \in K$.

Parameters:

N^k	the number of the k-type facilities allowed;
O^k	the operation cost for the k-type facilities;
R_j^k	the rental cost for the k-type facility at node j;
S^k	the maximum coverage distance of k-type facility;
d_{ij}	the distance between node i and node j;
w_i	the demand amount (for simplicity, can be viewed as potential revenue) of node i;
$f^k(d_{ij})$	the coverage function, which specifies the maximum covered percentage of demand node i by the k-type facility at j

$$f^k(d_{ij}) = \max\left\{1 - \frac{d_{ij}}{S^k}, 0\right\}. \quad (1)$$

Note that $f^k(d_{ij})$ is a function of the distance d_{ij} given the facility type k. For simplicity, we use a linear decreasing function to represent $f^k(d_{ij})$ (as shown in Fig. 1). Note coverage function can be modified according to the requirements in practice.

With the notation described above, the problem can be formulated as the following MIP model.

1) Model 1: Banking Facility Location Problem:
Decision variables:

x_j^k	$\in \{0,1\}, \forall j \in J, k \in K$. $x_j^k = 1$ if a k-type facility is located at j; otherwise $x_j^k = 0$.
y_i	$\in R, 0 \le y_i \le 1, \forall i \in I$, indicating the percentage of demands covered at node i.

Objective function:

$$\max : \sum_{i\in I} w_i y_i - \sum_{j\in J}\sum_{k\in K} (O^k + R_j^k) x_j^k. \tag{2}$$

Subject to:

$$\sum_{k\in K} x_j^k \le 1, \ \ \forall j \in J \tag{3}$$

$$\sum_{j\in J} x_j^k \le N^k, \ \ \forall k \in K \tag{4}$$

$$y_i \le 1, \ \ \forall i \in I \tag{5}$$

$$y_i \le \sum_{j\in J}\sum_{k\in K} f^k(d_{ij}) \cdot x_j^k, \ \ \forall i \in I. \tag{6}$$

The first term of the objective function (2) is the total revenue generated by these facilities, where we assume that the revenue can be viewed as the demand within that cell. The second term of the objective function is the overall cost for locating the facilities. Thus, the objective function (2) is to maximize the profit of facility networks, that is, the total revenue of covered demands minus the costs of sited facilities. Constraint (3) ensures that for each candidate location, only one type of facility can be chosen. Constraint (4) limits the number of each type of facility. Constraint (5) ensures that each demand node can be covered by at most 100% of its demand amount. Constraint (6) ensures that the coverage of each demand node is no more than what all the facilities can cover for this node. This problem has $|K| * |J|$ binary decision variables and $|I|$ continuous decision variables.

B. Properties of the Studied Problem

It is interesting to note that some candidate facilities and demand nodes can be ignored for this study. That is, the candidate facilities can be removed from the candidate pool if it cannot generate positive profit, a demand node can be ignored if its demand amount is zero or it is not covered by any possible facility. Removing the redundant candidate facilities and demand nodes can significantly reduce the size of the problem. It is obvious that such a reduced problem space still preserves the optimal solutions of the original problem.

C. Upper Bound Schemes

Due to the complexity and size of the problem, it is challenging to find the optimal solution of the problem with direct optimization techniques. Therefore, we study the upper bound for the problem to assess the solution quality.

In order to reduce the model complexity, we relax the constraints and let binary x_j^k be continuous and different types of facility be one type. It is defined as follows.

Definition 1: LP-Based Type-Relaxed Upper Bound (ub_{LPT}). It is the optimal objective function of the following problem:

$$x_j \in R, \ 0 \le x_j \le 1, \ \ \forall j \in J \tag{7}$$

$$N = \sum_{k\in K} N^k \tag{8}$$

$$O = \min_{\forall k\in K} \left\{O^k\right\} \tag{9}$$

$$R_j = \min_{\forall k\in K} \left\{R_j^k\right\}, \ \forall j \in J \tag{10}$$

$$S = \max_{\forall k\in K} \left\{S^k\right\}. \tag{11}$$

Objective function:

$$\max : \sum_{i\in I} w_i y_i - \sum_{j\in J} (O + R_j) x_j. \tag{12}$$

Subject to:

$$\sum_{j\in J} x_j \le N \tag{13}$$

$$y_i \le 1, \ \ \forall i \in I \tag{14}$$

$$y_i \le \sum_{j\in J} f(d_{ij}) \cdot x_j, \ \ \forall i \in I. \tag{15}$$

The intuition behind *Definition 1* is clear. We create one pseudo-candidate facility type and assume all the facility types are identical. With (7), we simplify the binary decision variables as continuous variables within interval [0, 1]. We assume the number of facilities equals to the sum of all types of candidate facilities in the original problem as (8) indicates. The pseudo-facility has the minimal operational and rental cost while the maximal coverage radius, as (9), (10), and (11) indicate. The coverage function $f(d_{ij})$ of (15) is equal to the situation of (1), where S^k is replaced by S. Let the optimal objective function of this new problem as ub_{LPT}, we know that it is the upper bound of the original problem in Model 1.

III. A Hybrid Nested Partitions Algorithm

It is in general computationally intractable to solve the large-scale optimization problems formulated in the previous section due to the NP-hard characteristic. In this section, we first introduce a hybrid algorithm integrating mathematical programming and the nested partitions (NP) method to address this challenge. We then develop some heuristics and extend the algorithm. The readers are referred to Shi and Olafasson [16] for details of nested partitions which consist of (1) partitioning: aiming to cluster the good solutions in the promising regions; (2) random sampling: aiming to generate new candidate solutions; (3) estimating the promising index: aiming to assess the performance of each region; (4) backtracking: aiming to backtrack to the earlier regions for better solution if necessary.

In the standard nested partitions method, complete solutions (samples) are generated in each sampling step. However, when dealing with MIP models, [10] reports that it is more advantageous to only sample a number of partial solutions, $\mathcal{P}$, where the decisions are not all finalized yet. Note each partial solution represents a set of samples. Thus, the first step of applying nested partitions algorithm is to determine a proper form of partial solutions. Here, we apply an LP-based sampling procedure to generate partial solutions. Similar to [10], we relax the binary decision variables x_j^k to be continuous and transfer the original problem to an LP problem. But this relaxed LP problem still requires lots of computation efforts for large-scale problems, especially when we consider the situation with multiple types of facilities. Thus, we further relax the problem to a one-type facility LP problem, which is defined by *Definition 1*. Second, we solve the LP problem and let $\tilde{v}_j = \sum_{k\in K} \tilde{x}_j^k$. By normalizing $\tilde{v}_j, j \in J$ within [0, 1], we obtain the sampling weight v_j of banking facility location $j, \forall j \in J$. The sample weight is used to guide the sample procedure. If one location j has a larger weight v_j, it would be sampled as a facility location with a larger probability. At the same time, we obtain the upper bound, ub_{LPT}, for the original problem. Next, based on the sampling weights, a partial solution can be generated by randomly selecting P candidate facility locations based on standard weighted sampling [19] and closing all the other $|J| - P$ facility locations, where P is the number of facility locations in a partial solution and it is predetermined.

For each partial solution, we need to obtain a good sample to evaluate the promising index of the solution. Once samples are determined, we use mathematical programming to solve the associated problem (2)–(6), by setting $x_j^k = 0, \forall k \in K$ if facility location j is closed in the partial solution. Since $P \leq |J|$, this problem size is much smaller than the original problem. After the evaluation of partial solutions sampled in both the promising region and the surrounding region, the promising index can be calculated as the value of the best sample within that region [19].

Next, we continue with the partitioning or backtracking processes. On one hand, if the most promising index is at the current promising region, we further partition this region to generate a new promising region. For all the locations which are undecided in the current promising region but are sited by facilities in the best sample, we compare their corresponding sample weights and choose the one location with the largest sample weight. Then, the new promising region is generated by setting this location determined as it is in the best sample. On the other hand, if the promising index of the surrounding region is better, i.e., the best-so-far solution appears in the surrounding region, we backtrack into the region which contains the original promising region and the current best solution.

Based on the above description, the hybrid nested partitions algorithm can be outlined as follows.

Step 0) Eliminate the redundant candidate facilities and redundant demand nodes to reduce the problem size.

Step 1) Set the initial promising region as the total *solution space*, and the initial surrounding region as an empty set.

Step 2) Solve the LP problem of *Definition 1* to obtain a relaxed solution for the current promising region. With the LP solution as weights, sample over the promising region and surrounding region to generate *partial solutions*.

Step 3) For each partial solution, solve the related MIP problem to obtain the corresponding solutions. Calculate the *promise index* of the promising region and surrounding region. If the promising region has a better promise index, go to Step 4; otherwise, go to Step 5.

Step 4) Execute the *partitioning* operation at the current promising region and take its best subregion as the new promising region for the next iteration. Then go to Step 6.

Step 5) Perform *backtracking* operation. The resulting region is set as the next promising region.

Step 6) If the number of sited facilities meets the given $N^k, \forall k \in K$, or the gap between the best-so-far solution and LP upper bound becomes sufficiently small, the algorithm stops and outputs the current best solution; otherwise, go to Step 2.

The above hybrid nested partitions algorithm has promise for large-scale problems. However, when dealing with problems with extremely large size, the computation expense of above algorithm is still of great concern. We embed heuristic information into the algorithm aiming to reduce the computation furthermore.

Apparently, the computation time of the algorithm depends on the initial solution pool. We develop heuristics to generate good solutions. The facilities are picked based on the initial sample weights, $v_j^0, \forall j \in J$, which are obtained based on a heuristic procedure as follows. This procedure starts with an empty solution set and then adds the best facility location to this set one by one. In other words, we pick the first facility location which generates the most increment to objective function (2). Denote the index of this facility location as j_1, and denote the increment of objective function by this facility as $\tilde{v}_{j_1}^0$. For the second facility location, we pick the location j_2 that covers the most demands not covered by the first one, and denote the increment of objective function as $\tilde{v}_{j_2}^0$. This procedure continues until no positive increment of objective function can be obtained. Normalize these increment values and get the initial weight v_j^0 for each facility location j. Facility locations with initial weights higher than a given threshold are picked out and the number of these locations Also, the role of the heuristic at the end of Section III needs to be specified are usually smaller than the total required number of facilities N. These picked facility locations will be fixed as good locations for our original problem and we will concentrate the efforts to look for the locations of left facilities. Thus, it equals to the number of facility locations of initial problems N is reduced and the problem complexity will be reduced greatly. Although the global optimality is not guaranteed during such a procedure, considerable computations are saved and the algorithm can be extended to practical systems with extremely large-scale. Numerical results demonstrate the effectiveness and efficiency of this extension later in Section IV.

TABLE I
PARAMETERS FOR THE PRACTICAL TEST

scale para.	value	bank numb.	value	opr. cost	value	rent cost	value	cover radius	value
$\|I\|$	45136	N^1	12	O^1	92	R^1	8.3	S^1	7.3
$\|J\|$	45136	N^2	38	O^2	62	R^2	6.4	S^2	5.2
$\|K\|$	3	N^3	33	O^3	12	R^3	5.4	S^3	3.1

TABLE II
COMPARISON OF HEURISTIC HYBRID NESTED PARTITIONS ALGORITHM AND CURRENT SCHEME APPLIED IN PRACTICE

CS	NP	n-cpu (s)	Upper bound ub_{LPT}	u-cpu (s)	Gap_1 (%) CS and ub_{LPT}	Gap_2 (%) NP and ub_{LPT}
$1.94*10^6$	$4.98*10^6$	40243	$6.10*10^6$	241	68.2	18.4

In this table, the column "CS" shows the performance of current facility scheme applied in practice; the column "NP" denotes the solution performance of our algorithm; the column "n-cpu" is the computation time of our algorithm; the column "ub_{LPT}" indicates the LP-based type-relaxed upper bound of the problem; the column "Gap_1" indicates the relative difference of current scheme and the upper bound ub_{LPT}, i.e., $\text{Gap}_1 = (ub_{LPT} - \text{CS})/(ub_{LPT})*100\%$; the column "$\text{Gap}_2$" denotes the relative difference of our algorithm solution and the upper bound, i.e., $\text{Gap} = (ub_{LPT} - \text{NP})/(ub_{LPT})*100\%$.

IV. NUMERICAL RESULTS

In this section, we conduct one case study to demonstrate the applicability of the extension of hybrid algorithm. The banking location problem studied is for a city in China, where the number of demand nodes and candidate facility locations are both 45,000. We compare the algorithm result with a loose upper bound ub_{LPT} which can be obtained within a reasonable computation time for the large-scale problem. Moreover, the algorithm result is also compared with the current scheme of bank locations in this city.

Table I summarizes the parameters used in the study. Additionally, we set $R_j^k = R^k \cdot w_j$ and $P = 2\sum_{k \in K} N^k$. All these parameters are determined from the industry project. The number of each type of facility, N^k, is predefined by the bank operators. The operation cost, rent cost, and coverage radius of each facility are all estimated based on banks practical situation.

The demand amount of each cell is estimated based on practical GIS data. The detailed distribution of demand amounts is illustrated by Fig. 2, where the colors of cells range from white, blue, green, yellow, orange, pink to red, and sequentially represent a series of increasing demand amounts. The real location of current banking facilities is also shown in Fig. 2, where circle, diamond, and star symbols represent types I, II, and III banking facility, respectively. Please note, we only illustrate the main city area of this city and there are many other blue cells in the rural area which are not displayed in Fig. 2.

Before we use the hybrid nested partitions algorithm with heuristic rules to optimize this practical problem, we use the concept of redundant facility locations and demand nodes to reduce the search space. After the elimination of redundant candidate facility locations and demand nodes, the numbers of valid demand nodes and candidate locations are $|I|' = 1598$ and $|J|' = 415$, respectively. This improves the efficiency of our algorithm. Since the optimal solution for the large-scale problem is unknown, we compare the solution of our algorithm with a current scheme applied in practice and the LP-based type-relaxed upper bound, i.e., ub_{LPT}. The detailed results are shown in Table II.

From Table II, we conclude that the profit of the current facility scheme is $1.94 * 10^6$ with the practical parameter setting, whereas the profit of the solution of our algorithm is $4.98 * 10^6$, which significantly outperforms the current scheme. Furthermore, we compare the result of our algorithm and the current scheme with the upper bound of this problem, and it shows that the relative difference between our solution and the upper bound is within 18.4%, whereas the gap between the performance of the current location scheme and the upper bound reaches to 68.2%. Note that the upper bound used here is an LP-based type-relaxed upper bound of this problem, i.e., ub_{LPT}, which is quite loose. This result demonstrates that our algorithm is promising to handle the extremely large-scale problem in practice.

V. Discussion and Conclusion

This paper applies a heuristic-based hybrid nested partitions algorithm to solve the facility location problem in practice, especially for the banking facility location problem in a practical project. The approach is based on the nested partitions methodology and utilizes the problem structure to improve the algorithm efficiency. In order to make the sampling process of nested partitions more efficient, a mathematical programming solver is applied to guide the sample process. This hybrid approach is further extended with problem-specific heuristic rules to make the algorithm applicable to large-scale problems. Numerical results demonstrate the effectiveness and efficiency of the algorithm.

In the future, we will generalize the problem formulation by introducing the constraint of capacitated facilities. Additionally, we simplify the problem by not considering the bank competitor. In reality, the competition raised from other bank organization deserves further considerations and should be included in the model. Another valuable work is to develop a tighter upper bound of this problem, since the current upper bound in this paper is loose.

References

[1] R. H. Ballou, *Business Logistics Management: Planning, Organizing, and Controlling the Supply Chain*. Upper Saddle River, NJ: Prentice-Hall, 1999.

[2] F. Barahona and F. A. Chudak, "Near-optimal solutions to large-scale facility location problems," *Discrete Opt.*, vol. 2, no. 1, pp. 35–50, 2005.

[3] O. Berman and D. Krass, "The generalized maximal covering location problem," *Comput. Oper. Res.*, vol. 29, pp. 563–581, 2002.

[4] L. F. M. Cheong, "New models in logistics network design and implications for 3PL companies," Ph.D. Dissertation, MIT ALLIANCE, Singapore, 2005.

[5] R. Church and C. ReVelle, "The maximal covering location model," *Papers of the Regional Science Association*, vol. 32, no. 1, pp. 101–118, 1974.

[6] R. D. Galvaro, L. G. A. Espejo, and B. Boffey, "A comparison of lagrangean and surrogate relaxations for the maximal covering location problem," *Eur. J. Oper. Res.*, vol. 124, pp. 377–389, 2000.

[7] M. Gendreau, G. Laporte, and F. Semet, "The maximal expected coverage relocation problem for emergency vehicles," *J. Oper. Res. Soc.*, vol. 57, no. 1, pp. 22–28, 2006.

[8] S. G. Kolliopoulos and S. Rao, "A nearly linear-time approximation scheme for the Euclidean k-median problem," *SIAM J. Comput.*, vol. 37, no. 3, pp. 757–782, 2008.

[9] M. A. Pereira, L. A. N. Lorena, and E. L. F. Senne, "A column generation approach for the maximal covering location problem," *Int. Trans. Oper. Res.*, vol. 14, no. 4, pp. 349–364, 2007.

[10] L. Pi, Y. Pan, and L. Shi, "Hybrid nested partitions and mathematical programming approach and its applications," *IEEE Trans. Autom. Sci. Eng.*, vol. 5, pp. 573–586, 2008.

[11] H. K. Rajagopalan, C. Saydam, and J. Xiao, "A multiperiod set covering location model for dynamic redeployment of ambulances," *Comput. Oper. Res.*, vol. 35, no. 3, pp. 814–826, 2008.

[12] C. ReVelle and H. Eiselt, "Location analysis: A synthesis and survey," *Eur. J. Oper. Res.*, vol. 165, pp. 1–19, 2005.

[13] C. ReVelle, M. Scholssberg, and J. Williams, "Solving the maximal covering location problem with heuristic concentration," *Comput. Oper. Res.*, vol. 35, pp. 427–435, 2008.

[14] G. Sahin and H. Sural, "A review of hierarchical facility location models," *Comput. Oper. Res.*, vol. 34, no. 8, pp. 2310–2331, 2007.

[15] L. Shi, R. R. Meyer, M. Bozbay, and A. J. Miller, "A nested partitions framework for solving large-scale multicommodity facility location problems," *J. Syst. Sci. Syst. Eng.*, vol. 13, no. 2, pp. 158–179, 2004.

[16] L. Shi and S. Olafsson, "Nested partitions method for global optimization," *Oper. Res.*, vol. 48, pp. 390–407, 2000.

[17] L. V. Snyder, "Facility location under uncertainty: A review," *IIE Trans.*, vol. 38, pp. 547–564, 2006.

[18] G. A. Tanonkou, L. Benyoucef, and X. Xie, "Design of stochastic distribution networks using Lagrangian relaxation," *IEEE Trans. Autom. Sci. Eng.*, vol. 5, pp. 597–608, 2008.

[19] C. K. Wong and M. C. Easton, "An efficient method for weighted sampling without replacement," *SIAM J. Comput.*, vol. 9, no. 1, pp. 111–113, 1980.

[20] J. Zhang, "Approximating the two-level facility location problem via a quasi-greedy approach," *Math. Program.*, vol. 108, no. 1, pp. 159–176, 2006.