Computers & Industrial Engineering 112 (2017) 483–491

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

Competitive facility location problem with foresight considering service distance limitations

Mingyao Qi *, Mingfei Xia, Ying Zhang, Lixin Miao

Research Center on Modern Logistics, Graduate School at Shenzhen, Tsinghua University, Shenzhen 518055, China

ARTICLE INFO

Article history:
Available online 18 April 2017

Keywords:
Competitive facility location problem
Service distance
Tabu search
Gravity-based model
Bi-level Nonlinear Integer Programming

ABSTRACT

This paper presents a bi-level, nonlinear, integer programming model for the competitive facility location problem with foresight. The developed model's objective is to maximize the leader's market share while also taking into consideration the follower's response. In the classical competitive facility location model, it is assumed that the facility competes for all customers, no matter how far away they are. Instead, this paper considers a new kind of customer behavior in which people only patronize facilities within a range they feel is convenient, which is more realistic than the existing models. To solve the model, a two-stage hybrid tabu search algorithm is proposed. A set of randomly generated instances are presented and analyzed statistically in order to illustrate the effectiveness of the proposed algorithm. The results indicate that the proposed algorithm provides an effective means to solve the problems and that service distance is proved to be a significant factor in the model.

1. Introduction

In recent years, China's express delivery industry has recorded a high annual growth. With the boom in this sector, express enterprises are willing to expand their market share. To this end, some express enterprises are determined to launch new express service stores; however, any industry competitors will react by opening new facilities in the future. This situation is one that is often encountered: one company (the leader) opens facilities in the market and another company (the follower) locates its new facilities later. This is the framework of the competitive facility location problem.

The competitive facility location problem (CFLP) differs from the classic facility location problem (FLP) in the respect that it explicitly incorporates the fact that other facilities are already (or will be) present in the market and that any new facility(ies) will have to compete with them for its (their) market share (Plastria, 2001). The competitive facility location is categorized into three categories (Ashtiani, Makui, & Ramezanian, 2013): (1) static competition, in which the competitors are already in the market and the planner of the new facilities knows their information; (2) competition with foresight, in which the potential competitors are not in the market yet but will be present soon after the new facilities are built; therefore, the leader wants to locate a facility in a qualitative way that maximizes its total captured market share after the follower located its facility; and (3) dynamic competition, in which players repeatedly re-optimize their locations. In this paper, the model under study is competition with foresight, in which two competitors successively launch their facilities with the goal of capturing the market share. Moreover, we represent decision-making solutions that consist of the following two stages. In the first stage the leader locates his new facilities to maximize his market share under the condition that he knows follower's objective function. In the second stage the follower, knowing the leader's facility information, places his facilities in order to maximize his market share (Beresnev, 2013). This sequential procedure pursues the optimal decisions for the two players, which is also known as a Stackelberg game.

The location space is also a key ingredient that affects the location model. For instance, when the location space occurs on a plane, then the new facilities can be located continuously on a two-dimensional space. However, if the location space is a discrete set and known a priori, the new facilities will be located from a set of candidate points. The customers and facilities can also be assumed to be in a network consisting of edges and vertices. For this paper's research background, we are dealing with a type of discrete set.

* Corresponding author.
E-mail addresses: qimy@sz.tsinghua.edu.cn (M. Qi), xiamingfei13@163.com (M. Xia), yingzhang8996@gmail.com (Y. Zhang), lxmiao@tsinghua.edu.cn (L. Miao).

http://dx.doi.org/10.1016/j.cie.2017.04.024
0360-8352/

Other fundamental categories of the competitive facility location problem are related to the customer behavior. Two customer behavior models, the deterministic model and the stochastic model, have been proposed in previous literature. In the deterministic model, it is assumed that customers patronize the facility that gives the highest utility. On the other hand, in the random model customers visit any facility with respect to some probability, typically according to the distance and the quality of the facilities.

Hakimi (1983) introduced the leader-follower issue in the competitive facility problem. He used the expression "medianoid" for the follower's problem and "centroid" for the leader's problem. In a centroid $(r|p)$ problem, the leader will locate p new facilities with the belief that the follower will invest r new facilities later. The medianoid problem $(r|X_p)$ is to locate r new facilities for the follower in order to maximize its market share, knowing that the leader has located p new facilities. Furthermore, Hakimi has proven that the leader–follower problems in $(r|X_p)$-medianoid and $(1|p)$-centroid cases are NP-hard (Hakimi, 1983). Eiselt and Laporte (1997) reviewed research work on the leader-follower problem until 1996. Plastria (2001) provided an overview of the static competitive facility location. Kress and Pesch (2012) reviewed sequential competitive location problems on networks. Shiode and Drezner (2003) presented the competitive facility location problem on a tree network with stochastic weights. Ahn, Cheng, Cheong, Golin, and Van Oostrum (2004) considered that two players each successively place one facility into the market, until each of them has placed n facilities. Beresnev (2013) proposed a branch-and-bound algorithm for the competitive facility location problem. Ashtiani et al. (2013) provided a robust model for determining optimal locations for the leader's new facilities when the number of the follower's new facilities is unknown. Plastria and Vanhaverbeke (2008) solved the competitive location problem with foresight in which the competitor will locate a single new facility. Alekseeva, Kochetova, Kochetov, and Plyasunov (2010) worked with regard to the discrete $(r|p)$-centroid problem, based on deterministic customer behaviors. Shiode, Yeh, and Hsia (2012) investigated the optimal location policy for three competitive facilities. Unlimited to the facility location decision, some recent studies have examined the facility design aspects simultaneously; for instance, matters relating to size, product variety, and so on. This kind of problem is known as the competitive facility location-design problem; see detailed reviews in Aboolian, Berman, and Krass (2007), Redondo, Fernández, García, and Ortigosa (2010), Küçükaydin, Aras, and Kuban Altınel (2011), Sáiz, Hendrix, and Pelegrín (2011), Fernández, Salhi, and Tóth (2014), Saidani, Chu, and Chen (2012), Wang and Ouyang (2013). Table 1 lists some of the relative studies, classifying them in terms of game theoretic aspect, location space, customer behavior and location and design.

Based on the related literature, the conclusion can be drawn that few researchers have considered service distance limitations when modeling the competitive facility location problem. All previous papers have assumed that the customer can be serviced by any facility in the market; however, this assumption is not always realistic. For example, in the express service store-location problem, the facility's service distance is taken into consideration: the store will only capture the customers within the service distance.

For this paper we have taken into consideration the facility's service distance in the competitive facility location problem with foresight. The remainder of the paper is organized as follows. Section 2 presents the notations and our model. The algorithm is explained in Section 3. Numerical examples and computational results are given in Section 4. Section 5 concludes the paper and presents directions for future research.

2. Model description

A two-dimensional market region is considered in which the demand is assumed inelastic and is supposed to be concentrated in n demand points. Two competitors, both providing identical services, are referred to as the leader and the follower. There are m facilities; of these facilities, the leader owns t facilities and the follower owns the rest of the m-t facilities. The existing facilities are placed in m of n demand points and the remaining n-m points can be regarded as potential locations. The leader intends to locate p new facilities in the potential locations, given that the follower will surely respond to his action by launching r new facilities in the potential locations. It is assumed that only one new facility can be opened at each potential location.

The customer's behavior is important in the competitive facility location problem, because it is necessary to describe the demand captured by each competing facility in a precise manner. With regards to the facility's service distance limitations, this paper incorporates a new kind of customer behavior that states people will only patronize facilities within a range they feel is convenient. First, when the customer is within a facility's service distance, the customer behavior follows a random model; in other words, his demand is split by these facilities. Second, if the customer is within only one facility's service distance, the customer's behavior follows a deterministic model such that his full demand is serviced by this facility. Finally, if the customer is beyond any facility's service distance, his demand is unserved. The quality levels of all facilities are assumed to be predetermined.

Table 1
Selected researches and classification.

Authors and year	Game theoretic aspect	Location space	Customer behavior	Location and design
Ashtiani et al. (2013)	With foresight	Discrete set	Probabilistic	Location
Beresnev (2013)	With foresight	Discrete set	Deterministic	Location
Hakimi (1983)	Static	Network	Deterministic	Location
Shiode and Drezner (2003)	With foresight	Network	Deterministic	Location
Ahn et al. (2004)	Dynamic	Plane	Deterministic	Location
Plastria and Vanhaverbeke (2008)	With foresight	Discrete set	Deterministic	Location
Alekseeva et al. (2010)	With foresight	Discrete set	Deterministic	Location
Shiode et al. (2012)	With foresight	Network	Deterministic	Location
Aboolian et al. (2007)	Static	Discrete set	Probabilistic	Location and design
Redondo et al. (2010)	With foresight	Plane	Probabilistic	Location and design
Küçükaydin et al. (2011)	With foresight	Discrete set	Probabilistic	Location and design
Sáiz et al. (2011)	With foresight	Discrete set	Probabilistic	Location and design
Fernández et al. (2014)	With foresight	Plane	Deterministic	Location and design
Saidani et al. (2012)	With foresight	Plane	Probabilistic	Location and design
Wang and Ouyang (2013)	With foresight	Discrete set	Deterministic	Location and design

We define the notations of our model as follows:

i	index of existing facility, $i = 1,2,\ldots,t$ leader's facility, $i = t+1, t+2,\ldots,m$ follower's facilities
j	index of demand points, $j = 1,2,\ldots,n$
k	index of leader's new facilities, $k = 1,2,\ldots,p$
h	index of follower's new facilities, $h = 1,2,\ldots,r$
n_{pot}	number of potential locations ($n_{pot} = n$-m)
s	index of potential locations, $s = 1,2,\ldots,n_{pot}$
b_j	buying power of demand point j
d_{ij}	Euclidean distance between existing facility i and demand point j
d_{sj}	Euclidean distance between the potential location s and demand point j
q_{ij}	quality of existing facility i for demand point j
q_{Lj}	quality of the leader's new facility for demand point j
q_{Fj}	quality of the follower's new facility for demand point j
X_s	binary variable, which is equal to 1 if leader opens new facility in potential location s, and 0 otherwise
Y_s	binary variable, which is equal to 1 if follower opens new facility in potential location s, and 0 otherwise

In the competitive location problem, a gravity-based model is widely employed in which the facility attractiveness level for a customer is proportional to the quality of the facility and inversely proportional to the squared distance between the customer and the facility. According to this, the attractiveness level of facility i for customer j who is within its service distance is given by

$$A_{ij} = \frac{q_{ij}}{(\varepsilon + d_{ij}^2)} \tag{1}$$

In the equation, ε is added to d_{ij}^2 to avoid the denominator becoming 0 when the distance between the facility and the customer reduces to 0. If the facility's service distance is assumed to be β, similarly, the attractiveness level of facility i for customer j is as follows:

$$f(i,j) = \begin{cases} \frac{q_{ij}}{(\varepsilon + d_{ij}^2)}, & d_{ij} \leqslant \beta; \\ 0, & d_{ij} > \beta. \end{cases} \tag{2}$$

Then, the total attractiveness of customer j from all the facilities is expressed as

$$A_j = \sum_{i=1}^{m} f(i,j) + \sum_{s=1}^{n_{pot}} f(s,j)X_s + \sum_{s=1}^{n_{pot}} f(s,j)Y_s \tag{3}$$

The probability that customer j patronizes facility i of the leader is displayed as

$$P_{ij} = \frac{f(i,j)}{A_j + \alpha}, i = 1,2,\ldots,t. \tag{4}$$

The parameter α is added to A_j to avoid the denominator becoming zero. Consequently, the probability that customer j patronizes leader's all facilities is represented as

$$P_{Lj} = \frac{\sum_{i=1}^{t} f(i,j) + \sum_{s=1}^{n_{pot}} f(s,j)X_s}{A_j + \alpha}. \tag{5}$$

As a result, the total market share captured by the leader is illustrated as

$$M_L = \sum_{j=1}^{n} b_j P_{Lj}. \tag{6}$$

Similarly, the probability that customer j patronizes follower's facilities is described as

$$P_{Fj} = \frac{\sum_{i=t+1}^{m} f(i,j) + \sum_{s=1}^{n_{pot}} f(s,j)Y_s}{A_j + \alpha}. \tag{7}$$

Accordingly, the total market share captured by the follower is expressed as

$$M_F = \sum_{j=1}^{n} b_j P_{Fj}. \tag{8}$$

Then, the competitive facility location problem with foresight can be formulated as the following bi-level programming model:

Upper level (leader model):

$$Max\ M_L(X,Y) = \sum_{j=1}^{n} b_j P_{Lj}(X,Y) \tag{9}$$

$s.t.$

$$\sum_{s=1}^{n_{pot}} X_s = p, \tag{10}$$

$$X_s + Y_s \leqslant 1 \quad s = 1,2,\ldots,n_{pot}, \tag{11}$$

$$X_s \in \{0,1\}, Y_s \in \{0,1\} \quad s = 1,2,\ldots,n_{pot}. \tag{12}$$

Here, Y, for each value of X, is the optimal solution for the lower level problem:

Lower level (follower model):

$$Max\ M_F(Y) = \sum_{j=1}^{n} b_j P_{Fj}(X,Y) \tag{13}$$

$s.t.$

$$\sum_{s=1}^{n_{pot}} Y_s = r, \tag{14}$$

$$X_s + Y_s \leqslant 1 \quad s = 1,2,\ldots,n_{pot}, \tag{15}$$

$$X_s \in \{0,1\}, Y_s \in \{0,1\} \quad s = 1,2,\ldots,n_{pot}. \tag{16}$$

For the upper level model, the objective function (9) represents the market share collected by the leader's facilities. Constraint (10) ensures that each leader's new facility is opened in only one of the potential locations. Constraints (11) limit the number of facilities that open in each candidate's site, either by the leader or the follower. Constraints (12) state the nature of the decision variable.

For the lower level model, the goal of objective function (13) is to maximize the follower's market share. Constraints (14)–(16) can be explained analogously as constraints (10)–(12). Based on this two-stage model, for each feasible solution of the leader, the optimal solution of the follower can be acquired by solving the follower model. If each feasible solution of the leader is enumerated by solving a follower model, the leader's optimal solution will then be obtained. However, as the enumeration mechanism is not applicable for large-scale problems that widely exist in reality, for the purposes of this paper we have resorted to heuristic approaches.

3. Hybrid Tabu Search Heuristic (HTS)

As mentioned before, the problem is a bi-level nonlinear integer-programming problem. Due to the high complexity involved in solving this type of problem, exact methods are inapplicable when solving the large-scale instances that are often encountered in practice. Attention can be paid to metaheuristics, which provide frameworks to heuristic algorithm designs. Tabu search is a local search heuristic proposed by Glover (1986). According to recent literature, tabu search is extensively applied to solve the competitive location problem and has been proven to be effective (Küçükaydin et al., 2011). In this section, a hybrid tabu search heuristic was designed to solve the proposed problem in which two tabu search procedures are implemented to explore the leader's and follower's potential locations, respectively. The notations for HTS are as follows.

Notation:	
X, Y	The leader/follower's incumbent solution
X^*, Y^*	The global best leader/follower solution
f_L^*, f_F^*	The leader/follower's market share of X^*/Y^*
f_{Lt}	A temporal value of the leader's market share
$N(X)$, $N(Y)$	The neighborhood of X/Y
$\tilde{N}(X)$, $\tilde{N}(Y)$	The admissible subset of $N(X)/N(Y)$ (i.e. non-tabu or allowed by aspiration)
T_L, T_F	The leader's/follower's Tabu list

3.1. Algorithm overview

Algorithm 1 presents the pseudocode of the main algorithm, which consists of two phases, the leader phase and follower phase. In the leader phase, the leader's decision space is explored using a tabu search mechanism, while in the follower phase (line 6–12), the follower's candidate solutions are examined also using a tabu search routine.

Algorithm 1. Hybrid tabu search algorithm.

1: Construct an initial solution X_0
2: Set $X \leftarrow X_0, f_L^* \leftarrow 0, X^* \leftarrow X_0, T_L \leftarrow \varnothing$
3: While the termination criterion not satisfied do
4:   Set $f_{Lt} = 0$,
5:   For each $X' \in \tilde{N}(X), X \leftarrow X'$
6:     Construct an initial solution Y_0
7:     Set $Y \leftarrow Y_0, f_F^* \leftarrow f_F(Y_0), Y^* \leftarrow Y_0, T_F \leftarrow \varnothing$
8:     While termination criterion not met do
9:       Select Y in $\mathrm{argmax}_{Y' \in \tilde{N}(Y)}[f_F(Y')]$;[a]
10:       If $f_F(Y) > f_F^*$, then set $f_F^* \leftarrow f_F(Y), Y^* \leftarrow Y$
11:       Record tabu for the current move in T_F (delete oldest entry if necessary)
12:     End while
13:     If $f_L(X) > f_{Lt}$, then set $f_{Lt} \leftarrow f_L(X), X_t \leftarrow X$
14:   End for
15:   $X \leftarrow X_t$
16:   If $f_L(X) > f_L^*$, then set $f_L^* \leftarrow f_L(X), X^* \leftarrow X$
17:   Record tabu for the current move in T_L (delete oldest entry if necessary)
18: End while

[a] In this algorithm, argmax returns the subset of solutions in $\tilde{N}(Y)$ that maximize f_F.

3.2. Initialization

At the beginning of Algorithm 1, an initial solution of the leader is chosen (line 1). We designed four ways to construct leader's initial solution X_0: (1) the random method, that is, randomly selecting p locations from n_{pot} potential locations; (2) the demand method, selecting top p points from the potential locations according to buying power (we suppose each candidate facility is also a demand point); (3) cover-demand method, in which we select top p points by the total buying power of the covered demand points, which are within the service area; (4) greedy method, which solves the leader model assuming that the follower will not create new facility, and initializes the leader's solution with the optimal result.

For the follower problem, the initialization process (line 6) is as follows:

Algorithm 2. Follower Initialization.

1. Sequence the remainder $n_{pot} - p$ potential locations in decreasing order in terms of b_j.
2. Select the top 2*(p+r) points in the sequence as the initial candidate set I_F. If 2*(p+r) > $n_{pot} - p$, then select all the potential locations for I_F.
3. Sequence I_F in decreasing order of cover-demand (total buying power within a facility's service area). Take the top r points from I_F to initialize Y_0.

3.3. Neighborhood operations

To enable efficient neighborhood operations, each leader or follower's solution is represented by a sequence of binary numbers, with each number indicting whether or not a new facility is established at that location. For example, as Table 2 shows, given 10 candidate locations, the leader will open 2 new facilities (5 and 9) and the follower will open other 2 facilities (1 and 2).

The neighborhood operation involves the 1-Swap move that shifts an open facility from one site to another candidate site. Take the follower's solution of Table 2 as an example, its neighborhood is shown in Table 3 when potential location 1 and 6 are selected for the 1-Swap move.

For the leader, neighborhoods are created by swapping each component X_{s_k} of the leader's current solution $X = (X_{s_1}, X_{s_2}, \ldots, X_{s_p})$ with each point in the leader's candidate swap list LC, represented by the point pair (X_{s_k}, LC). Additionally, the follower's neighborhoods are created by swapping each member Y_{s_h} of the follower's current solution $Y = (Y_{s_1}, Y_{s_2}, \ldots, Y_{s_r})$ with each point in the follower's candidate swap list FC, represented by point pair (Y_{s_h}, FC).

Given X and Y, the list FC is created by selecting a points from the rest $n_{pot} - p - r$ potential locations; five methods were designed to generate the list: (1) random method, select a points randomly; (2) cover-demand method, select a points which have the most cover-demand (i.e., total demand of the points that exist within the service distance of a facility); (3) uncover-demand method, which means the total buying power of locations that are within the service area of the facility under consideration but beyond that of any other facilities; (4) nearby method, select nearby a points for exchange; (5) hybrid method, divide the list into quarters with each part generated by one of the above methods.

Given X, the leader's swap list LC is generated by first solving a follower's solution Y, taking Y as subset of LC (being sure to include the most promising locations), and then selecting a points from the rest $n_{pot} - p - r$ potential locations according to the same methods that were adopted to the follower's swap list. Note that the length of LC is $a + r$ now.

3.4. Tabu moves and respiration rules

In the research, the tabu lists T_L and T_F recorded the last few transformations performed on the leader's solution and follower's solution, respectively, and therefore prohibits reverse transforma-

Table 2
Solution encoding.

	1	2	3	4	5	6	7	8	9	10
X	0	0	0	0	1	0	0	0	1	0
Y	1	1	0	0	0	0	0	0	0	0

Table 3
An example of the follower's neighborhood.

	1	2	3	4	5	6	7	8	9	10
X	0	0	0	0	1	0	0	0	1	0
Y	**0**	1	0	0	0	**1**	0	0	0	0

Table 4
Results of small instances.

No.	n	m	t	p	r	Optimal			HTS				
						Leader locations		Leader market	Leader locations		Leader market		Time (s)
						1	2		1	2	Value	Gap	
1	16	5	3	2	1	3	7	64.9406	3	7	64.9406	0	0.067
					2	6	8	57.5923	6	8	57.5923	0	0.109
					3	6	8	52.1362	6	8	52.1362	0	0.152
2	17	5	3	2	1	14	10	71.7112	14	10	71.7112	0	0.092
					2	7	10	59.7015	7	10	59.7015	0	0.202
					3	7	10	53.8131	7	10	53.8131	0	0.272
3	18	5	3	2	1	17	9	51.4842	17	9	51.4842	0	0.124
					2	12	9	46.4614	12	9	46.4614	0	0.166
					3	12	9	41.8539	12	9	41.8539	0	0.235
4	19	5	3	2	1	7	6	58.4885	7	6	58.4885	0	0.118
					2	7	9	52.9384	7	9	52.9384	0	0.224
					3	6	5	48.4352	6	5	48.4352	0	0.449
5	20	5	3	2	1	3	5	67.1893	3	5	67.1893	0	0.157
					2	10	3	58.7955	10	3	58.7955	0	0.3
					3	8	15	53.8922	8	15	53.8922	0	0.41
6	21	5	3	2	1	5	14	82.2994	5	14	82.2994	0	0.193
					2	10	6	71.0303	10	6	71.0303	0	0.421
					3	6	10	61.3944	6	10	61.3944	0	0.668
7	22	5	3	2	1	6	18	90.1852	6	18	90.1852	0	0.232
					2	14	16	78.6613	14	16	78.6613	0	0.426
					3	14	16	66.3405	14	16	66.3405	0	0.769
8	23	5	3	2	1	17	4	83.0274	17	4	83.0274	0	0.261
					2	4	18	70.0555	4	18	70.0555	0	0.607
9	24	5	3	2	1	6	10	93.0572	6	10	93.0572	0	0.322
					2	10	7	81.8097	10	7	81.8097	0	0.719
10	25	5	3	2	1	10	22	83.4099	10	22	83.4099	0	0.351
					2	10	22	73.3504	10	22	73.3504	0	0.732
11	26	5	3	2	1	10	7	71.7428	10	7	71.7428	0	0.424
					2	10	19	66.9933	10	19	66.9933	0	0.868
12	27	5	3	2	1	11	21	94.1337	11	21	94.1337	0	0.407
					2	11	2	83.4679	11	2	83.4679	0	0.948
13	28	5	3	2	1	11	27	95.6134	11	27	95.6134	0	0.679

tions. The aspiration criteria allow a tabu move when it results in a solution with better objective value than the current best-known solution.

4. Computational results and analysis

4.1. The performance of HTS

In this section, both small instances and large instances are generated randomly to evaluate the performance of the designed algorithm, as no benchmark problem currently exists in the literature relating to the problem. The buying power is randomly generated for different demand points in a range of 1–10. Quality values are randomly generated for all facilities in a range of 1–5. All the instances have been generated similarly to those in Ashtiani et al. (2013). The proposed HTC was implemented in the C# language and was run on a PC with an Intel Core i5 Processor (2.3 GHZ) and 4 GB memory.

In order to evaluate the effectiveness of the algorithm, we compared the HTS with the optimal solver on small instances. The optimal solution is derived by first enumerating all possible leader solutions, and then using the program Lingo to solve a follower model to optimality. The parameters of the HTS algorithm are set as follows: the maximal number of follower and leader iterations is 300; the non-updating times when exploring the solution space of the leader and follower (termination criterion) are both set to 30; the tabu list length is 7, $\varepsilon = 1$, $\beta = 1$, $\alpha = 0.000001$; and a is set to be $n/3$. After comparing the four methods in the leader's initial solution construction, the greedy method is adopted because it can generate higher quality solutions in most cases. The performance of all these methods will be elaborated on after this section. As for generating the candidate swap list, the hybrid method outperformed other methods on the objective value and the rate of convergence, which will also be illustrated later, in Section 4.3. The solutions for all small instances calculated by HTS are depicted and compared with optimal solutions in Table 4. As shown, for all small instances with 16–28 demand points (these are also the candidates' points for facilities, except for those facilities that already existed), the proposed HTS can obtain the same solution as the exact solver did in a very short time (less than 1 s), which demonstrates that the algorithm is effective and efficient for small instances.

Table 5 demonstrates the computational results of large instances with up to 100 demand points obtained by HTS, which could not be worked out using Lingo. According to Table 5, we observed that the HTS is efficient and can solve large instances with an average computational time of less than 75 s. Take the eighth instance as an example; the leader will open two new facilities. When the follower plans to open one new facility, the leader's optimal locations are 18 and 74. On the other hand, when the follower determines to launch two new facilities, the leader's optimal locations are 14 and 18. However, when the follower intends to open four new facilities, the leader's optimal locations are 3 and 18. This proves that the follower's reaction plays a critical role in the leader's location decision regarding the competitive facility location problem with foresight.

4.2. Evaluation of initialization methods

We have designed four methods (see Section 3.3) to initialize the leader's solution for the algorithm and here we will compare the results in order to choose the best method. In the target instance, there are 100 demand points, in which the leader will open four new facilities and follower will open two new facilities. Next, we calculate the instance 10 times for each method and obtain the average value. Fig. 1 illustrates how the objective value (the leader's market share) grows with the iteration number. The results show that the greedy method can generate high quality solutions, which can converge to the optimal solution quickly (after two iterations). Compared to the greedy method, the most cover-demand method is slightly inferior, as it can also converge to the optimal solution but more slowly (four iterations). The demand method and random method have inferior performance when compared to the two counterparts, as they converge slowly and cannot reach the optimal solution. Note that the difference between the objective values is only visible when we zoom in the curves. As we have observed, the greedy method is the best; therefore, we used it in our algorithm. Note that although the HTS algorithm converges within only a couple of iterations, the computational workload is sufficient, because for each iteration, all the neighborhood solutions of current X (see Section 3.4) are evaluated, and for each time of such evaluation, a follower model has to be solved.

Table 5
Results of large instances.

No.	n	M	t	p	r	Leader market	Leader locations		Average time (s)
							1	2	
1	30	5	3	2	1	108.99	13	17	2
					2	101.59	13	17	4
					3	101.59	13	17	7
					4	82.46	13	12	11
2	40	5	3	2	1	131.51	36	32	7
					2	118.01	32	38	14
					3	118.33	14	19	23
					4	112.55	3	24	37
3	50	5	3	2	1	127.49	39	20	5
					2	123.16	14	20	10
					3	119.91	42	20	19
					4	115.25	39	20	75
4	60	5	3	2	1	129.99	48	18	5
					2	129.99	18	48	13
					3	125.99	18	48	24
					4	118.61	18	48	42
5	70	5	3	2	1	143.99	41	19	5
					2	141.77	19	41	14
					3	134.99	25	37	30
					4	134.97	41	19	43
6	80	5	3	2	1	151.42	12	63	6
					2	143.42	12	63	15
					3	141.54	12	63	26
					4	137.15	63	38	49
7	90	5	3	2	1	145.99	45	17	7
					2	144.49	17	5	22
					3	141.99	45	17	38
					4	139.6	17	12	55
8	100	5	3	2	1	152.99	18	74	9
					2	151.99	14	18	22
					3	151.99	18	14	42
					4	148.99	18	3	70

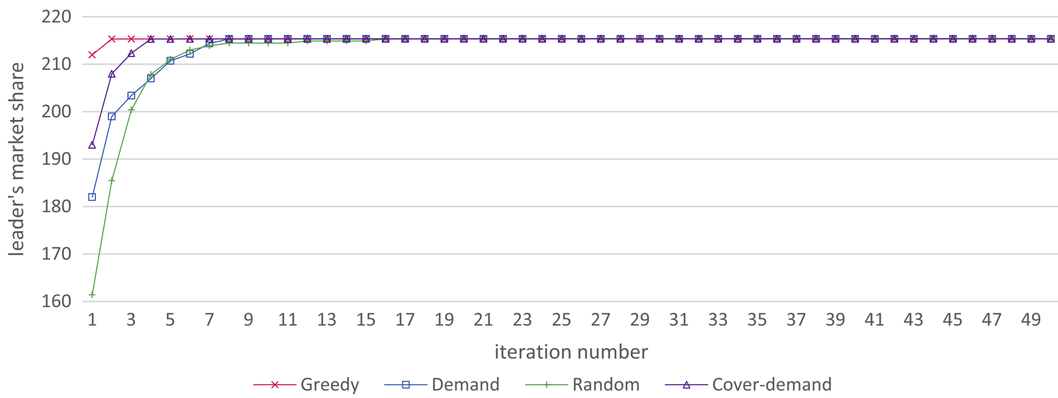

Fig. 1. Comparison of the four methods to initialize the leader's solution.

4.3. Evaluation of neighborhood methods

The candidate list determines how best to generate the neighborhoods of an incumbent solution, and therefore has critical influence on the solution quality and the convergence rate of the algorithm, especially for large instances. In the following section, we will focus on the performance of the proposed five methods (see Section 3.4). For this test we have adopted the instance with 100 demand points. The leader will open four new facilities and the follower will launch 10. To make it comparable, we fixed the leader's location plan and solved the follower model by using the five methods, respectively. The program was run 10 times and the average market share achieved by the follower after each iteration is shown in Fig. 2. According to the results, the hybrid, cover-demand and uncover-demand methods all performed well, which can quickly converge to the optimal solution, as their curves coincide with each other. The random method can also reach the optimal solution, but this takes longer. The nearby method is the worst in comparison to the others. In this paper we have selected the hybrid method for the algorithm.

4.4. Effects of service distance

In this research we explicitly considered the limitations of service distance in a competitive market; the distance that a facility can cover is expected to have critical effects on the decision-making process. To observe this, we tried different values of β (distance) on a set of scenarios, in terms of different p and r values. To simplify the computational process without any loss of generality, we took the medium-size instances for our experiments, which contained 50 demand points and five existing facilities, with three of those belonging to the leader. Three kinds of distance values ($\beta = 1, 2, 3$) were tested according to different new facilities configurations (r = 1, 2, 3 while fixing p to 2). The computational results are listed in Table 6.

It can be seen in Table 6 that, as the service distance increases from 1 to 2, the leader and follower's solutions exhibit significant differences. For instance, when p = 2, r = 3, the leader's choice for new facilities is (20, 40) for $\beta = 1$, while it is (32, 38) for $\beta = 2$. This coincides with our assumption that the service distance really matters in terms of decision-making. This also provides insight regard-

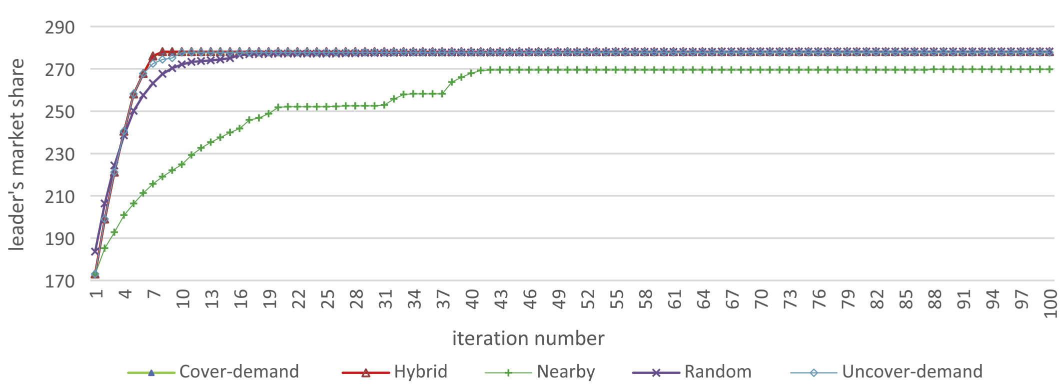

Fig. 2. Comparison of the five methods to generate candidate list.

Table 6
Effects of service distance.

β	n = 50, m = 5, t = 3											
	p = 2 r = 1				p = 2 r = 2				p = 2 r = 3			
	Leader		Follower		Leader		Follower		Leader		Follower	
	Market	Solution	Market	Solution	Market	Solution	Market	Solution	Market	Solution	Market	Solution
1	127.49	(20,39)	65.49	(42)	123.16	(14,20)	86.82	(39,42)	119.91	(20,40)	105.07	(34,47,48)
2	156.32	(19,32)	96.66	(38)	142.07	(19,32)	102.91	(12,38)	125.16	(32,38)	125.82	(12,19,39)
3	156.32	(19,32)	79.66	(38)	142.06	(19,32)	102.91	(12,38)	125.16	(32,38)	125.82	(12,19,39)

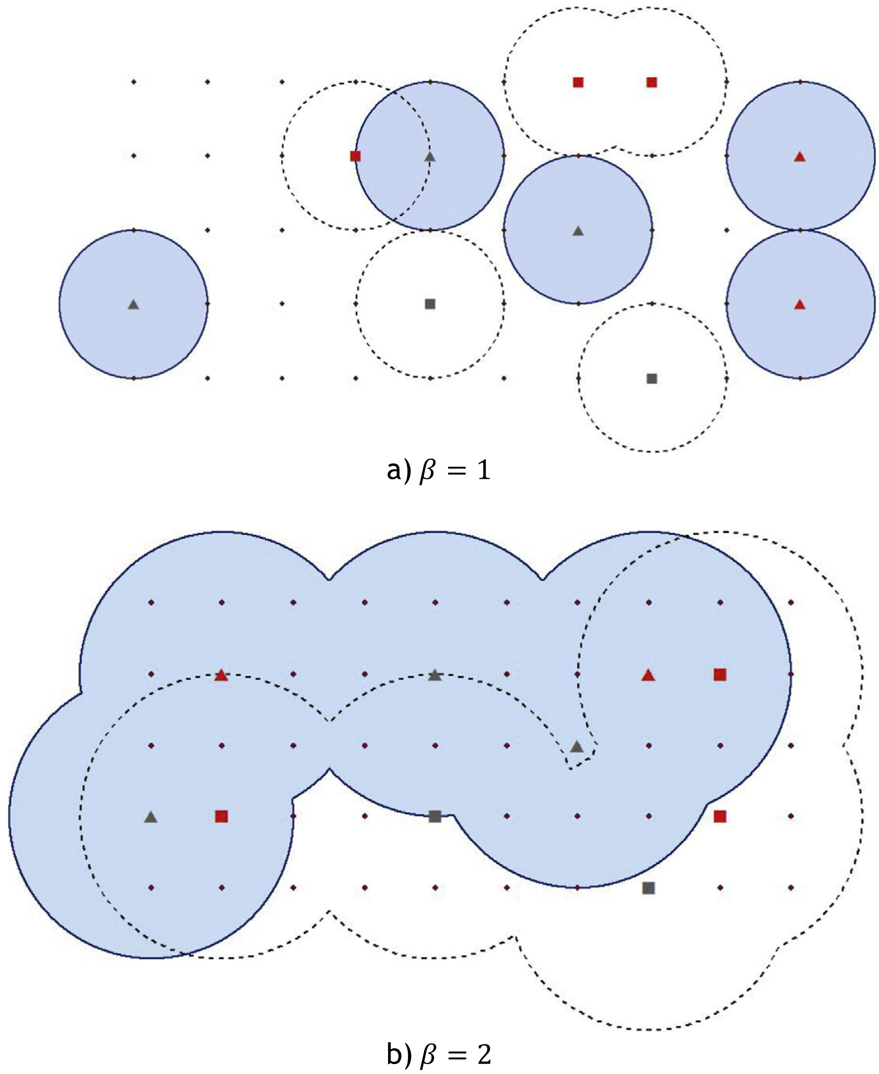

Fig. 3. Visualization of the solutions for different service distance.

ing how to better location plans according to a specific area's accessibility. For rural areas, for example, it is acceptable for local residents to visit more remote facilities; therefore, the choosing of new facilities should be different from those in downtown areas.

However, Table 6 also tells that when β further increases to 3, the solutions for both leader and follower remain the same. To examine this phenomenon, we used geographical information system (GIS) software, ESRI ArcGIS Ver10.2, to represent the locations and their coverage areas in a more visualized and intuitive way, as shown in Fig. 3.

In Fig. 3, the leader's existing and newly opened facilities are represented using gray and red triangles, respectively, while the follower's existing and new facilities are indicated with gray and red quadrates. Meanwhile, the service areas of the leader and follower's facilities are also depicted by regions, which merge from several service circles. Regions with solid borders are for the leaders, and those with dashed borders represent the follower. Fig. 3a implies that both the leader and the follower tend to locate facilities in isolated sub-areas in order to avoid competition. In Fig. 3b, as the circle radius increases, the uncovered areas are limited; therefore, most of the service areas of the rival parties overlap. If given a very large service distance (for example, $\beta = 3$ for this scenario), the problem is equivalent to a competitive location problem without service distance limitations. Therefore, the solution will remain stable when β increases even further, which explains the results shown in Table 6 that state the leader's decision will remain the same for $\beta = 2$ and $\beta = 3$.

5. Conclusions

In this paper, we studied a discrete competitive facility location problem with foresight by taking into account the service distance of the facilities. A new kind of customer behavior was proposed in this work that assumes a facility's attraction to a customer decreases with distance and falls to zero if the distance is beyond a limitation. This work proposed a bi-level, nonlinear, integer-programming model to solve this problem. In order to solve large-scale problems, a two-stage hybrid tabu search algorithm was developed. The performance of the designed algorithm was first evaluated by comparing it against optimal solutions on small instances. Next, the results on large-scale problems were reported and analyzed. The effect of parameter β was also analyzed in order to provide insight for other applications. We found that, on one hand, the service distance matters greatly in decision-making; on the other hand, the solution will remain stable when β increases further. Future work can be conducted by extending the problem to consider a situation in which the follower plans to build an unknown number of facilities.

Acknowledgement

National Key Technologies Research and Development Program (2016YFB0502601) and The National Natural Science Foundation of China (71272030) supported this work.

References

Aboolian, R., Berman, O., & Krass, D. (2007). Competitive facility location and design problem. *European Journal of Operational Research, 182*(1), 40–62.

Ahn, H. K., Cheng, S. W., Cheong, O., Golin, M., & Van Oostrum, R. (2004). Competitive facility location: The Voronoi game. *Theoretical Computer Science, 310*(1), 457–467.

Alekseeva, E., Kochetova, N., Kochetov, Y., & Plyasunov, A. (2010). Heuristic and exact methods for the discrete (r|p)-centroid problem. In *Evolutionary computation in combinatorial optimization* (pp. 11–22). Berlin, Heidelberg: Springer.

Ashtiani, M. G., Makui, A., & Ramezanian, R. (2013). A robust model for a leader–follower competitive facility location problem in a discrete space. *Applied Mathematical Modelling, 37*, 62–71.

Beresnev, V. (2013). Branch-and-bound algorithm for a competitive facility location problem. *Computers and Operations Research, 40*(8), 2062–2070.

Eiselt, H. A., & Laporte, G. (1997). Sequential location problems. *European Journal of Operational Research, 96*(2), 217–231.

Fernández, J., Salhi, S., & Tóth, B. G. (2014). Location equilibria for a continuous competitive facility location problem under delivered pricing. *Computers & Operations Research, 41*, 185–195.

Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research, 13*(5), 533–549.

Hakimi, S. L. (1983). On locating new facilities in a competitive environment. *European Journal of Operational Research, 12*(1), 29–35.

Kress, D., & Pesch, E. (2012). Sequential competitive location on networks. *European Journal of Operational Research, 217*(3), 483–499.

Küçükaydin, H., Aras, N., & Kuban Altınel, I. (2011). Competitive facility location problem with attractiveness adjustment of the follower: A bilevel programming model and its solution. *European Journal of Operational Research, 208*(3), 206–220.

Plastria, F. (2001). Static competitive facility location: An overview of optimisation approaches. *European Journal of Operational Research, 129*(3), 461–470.

Plastria, F., & Vanhaverbeke, L. (2008). Discrete models for competitive location with foresight. *Computers and Operations Research, 35*(3), 683–700.

Redondo, J. L., Fernández, J., García, I., & Ortigosa, P. M. (2010). Heuristics for the facility location and design (1|1)-centroid problem on the plane. *Computational Optimization & Applications, 45*, 111–141.

Saidani, N., Chu, F., & Chen, H. (2012). Competitive facility location and design with reactions of competitors already in the market. *European Journal of Operational Research, 219*(1), 9–17.

Sáiz, M. E., Hendrix, E. M. T., & Pelegrín, B. (2011). On Nash equilibria of a competitive location-design problem. *European Journal of Operational Research, 210*(3), 588–593.

Shiode, S., & Drezner, Z. (2003). A competitive facility location problem on a tree network with stochastic weights. *European Journal of Operational Research, 149*(1), 47–52.

Shiode, S., Yeh, K., & Hsia, H. (2012). Optimal location policy for three competitive facilities. *Computers and Industrial Engineering, 62*(3), 703–707.

Wang, X., & Ouyang, Y. (2013). A continuum approximation approach to competitive facility location design under facility disruption risks. *Transportation Research: Part B: Methodological, 50*, 90–103.