



SCIENTIA
IRANICA

Sharif University of Technology
Scientia Iranica
Transactions E: Industrial Engineering
<http://scientiairanica.sharif.edu>



Influence of two different producers in a competitive location problem

B. Yousefi Yegane*, I. Nakhai Kamalabadi, and H. Farughi

Department of Industrial Engineering, Faculty of Engineering, University of Kurdistan, Sanandaj, Iran.

Received 27 January 2018; received in revised form 20 December 2018; accepted 12 January 2019

KEYWORDS

Bilevel programming;
Competitive facility
location;
Genetic algorithm;
Ant colony
optimization
algorithm.

Abstract. Facility location of two producers with preference of customers is discussed in this paper. Because of the differences between two producers in terms of their influence on the market, the problem was formulated as a bi-level integer mathematical programming model with binary variables. It was considered that both leader and follower had some facilities at first and would open new facilities, which might lead to changes in the allocation of facilities and customers. To solve the problem, two metaheuristic algorithms based on Genetic Algorithm (GA) and a hybrid of GA and Ant Colony Optimization (ACO) were proposed. In the first section of each algorithm, the location of facilities for two producers was determined; in the second section, each customer selects a facility. Upper bound of the competitive facility location problem was determined by solving the upper-level problem as an integer linear programming model without considering the follower's decision. To evaluate the efficiency of the proposed algorithms, the enumeration technique was employed to find an optimal solution. Computational results demonstrated that all of the developed algorithms were capable of achieving an optimal solution to small-sized problems and a high-quality solution within a reasonable computational time to medium- and large-scale problems.

© 2020 Sharif University of Technology. All rights reserved.

1. Introduction

The facility location problem is a branch of operations research with great significance for both practical and combinatorial optimization perspectives. The classical location problem is concerned with determining the location of a facility to optimize the allocation of facilities to customers. Competitive Facility Location (CFL) is a special case of the location problem where at least two decision-makers, simultaneously or successively, start to seek maximum market shares

to optimize their objective functions by opening new Distribution Centers (DCs), considering the strength of the competitors. The special case where only two competitors attempt to open their facilities is known in the literature as the bilevel CFL problem and can be formulated as a bi-level mathematical programming model. In general, a bi-level programming problem is a mathematical program with an optimization problem in the constraint. Each decision-maker attempts to determine the decision variables under their control to optimize their own objective function regardless of the one of the other [1]. The main objective of this paper is to develop a mathematical model for the CFL problem with the highest degree of adaptability to the real-world applications; for example, each producer can spend a limited budget on expanding the facilities or production sites, which has not been considered by many researchers up to now. As a result of budget

*. *Corresponding author.*

E-mail addresses: bys.yegane@gmail.com (B. Yousefi Yegane); i.nakhai@uok.ac.ir (I. Nakhai Kamalabadi); h.farughi@uok.ac.ir (H. Farughi)

limitation, the producer can only open a few facilities among a set of potential locations. Furthermore, this study assumes that each facility can serve more than one customer, but each customer can satisfy its demand with only one DC.

2. Literature review

Pal and Sarkar [2] developed Cournot competition by allowing the location of multiple facilities and assuming that each producer can be present in all markets. Aboolian et al. [3] studied CFL with regard to the number of facilities, their locations, and their types (expressing the product variety and capacity of the facility). He formulated the problem as a nonlinear integer programming model and obtained its solutions through two heuristic algorithms, the greedy algorithm and the steepest descent algorithm. Beresnev [4] followed a new approach by formulating the CFL problem as a bilevel programming model and then, presented a new method for determining the upper bound of the problem; in this model, both competitors were seeking to maximize their profits.

Saidani et al. [5] studied the facility location problem by considering the responses from competitors in the market and used Huff's attractiveness function to determine the market share. Ashtiani et al. [6] employed robust programming to determine the optimum solution to the CFL problem to maximize market share for competitors, under assumption of an unknown number of follower centers. Beresnev [7] continued his research on the CFL problem by developing a branch and bound method for the determination of an optimum solution. Calvet et al. [8] proposed a bilevel programming model for planning a decentralized distribution network. They suggested an evolutionary algorithm to find the location of DCs and solved the delivery problem at the upper level using exact optimization methods. Rahmani and Mirhassani [9] studied the CFL problem as a bi-level mathematical programming model and presented acceptable solutions to their mathematical model by using Lagrangian relaxation method. Their results indicated that the proposed method was highly efficient. Mirhassani et al. [10] used a modified particle swarm optimization algorithm to solve the competitive location problem and compared their results with the upper bound obtained from solving a mathematical model. Based on the conducted analysis, their results showed the capability of the proposed meta-heuristic algorithm of obtaining high-quality solutions. Zhang et al. [11] considered a CFL problem with disruption and formulated the problem as a bilevel integer problem. A variable neighborhood decomposition search includes variable fixing and cut generation developed to solve the problem. Without considering the competition to

locate DCs, Sarangwong and Likasiri [12] presented a bilevel integer linear programming model with the objective of minimum transportation cost of product transportation from plants to facilities and from facilities to customers; they proposed four algorithms to solve their bilevel mathematical formulation. Qi et al. [13] studied the CFL problem as a bi-level mathematical programming problem and a two-stage hybrid tabu search algorithm was developed to solve the problem. Kung and Liao [14] studied the CFL problem by a discrete model to maximize the profit considering network effects. Two scenarios for demand function were assumed; in the first scenario, demand function was assumed to be linear and then, the optimal solution was found; when the demand function was nonlinear, they proved that the problem was NP-hard and then, an approximation algorithm was developed to solve the problem. Capacitated CFL problem with partial demand satisfaction was considered by Nasiri et al. [15]; two bi-level mathematical programming models were developed for the problem in which in the first model, the follower had to capture whole demand of a customer or leave it intact, while this assumption is relaxed in the second model; to solve these models, two meta-heuristic algorithms based on Genetic Algorithm (GA) and particle swarm optimization were developed. A branch and bound algorithm was developed to solve the capacitated CFL problem, as proposed by Beresnev and Melnikov [16]. The problem was formulated as a bi-level mathematical programming model with binary variables. A new mathematical model for competitive location problem was proposed by Sadjadi et al. [17] to determine the location of new facilities and also the optimal product type for each of opened facilities while the behavior of the customers was based on Huff rule. In addition to the above-mentioned studies, some researchers have studied multi-objective models of CFL [18–20].

Despite the fact that the competitive location problem has been the subject of much research, analyzing the problem as a bilevel mathematical programming model is a somewhat neglected approach [21], and few studies on this subject include the works of Beresnev et al. [4,7,21,22], Rahmani and Mirhassani [9], Mirhassani et al. [10].

In this paper, the location of facilities is determined using integer bilevel mathematical programming, and several methods including branch and bound, Descent methods, Penalty function methods, and Trust region methods have been developed [23] to deal with the inherent complexity of the problem in all forms such as integer linear or nonlinear formulation; accordingly, this study proposes metaheuristics approach to solve an integer nonlinear bilevel programming; hence, the rest of literature review will be focused on papers that have used metaheuristic-

based methods to solve bilevel programming models. (GA) have been proposed by many researchers [24–26]. Tabu search-based methods have been proposed in some papers to solve linear bilevel programming models and mixed integer bilevel programming models [27,28]. On the other hand, some researchers have applied hybrid metaheuristic algorithms to solve bilevel mathematical programming models; a hybrid intelligent algorithm combined by particle swarm optimization algorithm with chaos technique was presented by Wan et al. [29]. Kou et al. [30] proposed a hybrid immune GA and particle swarm optimization for solving a bilevel linear programming problem. More information about the proposed numerous metaheuristic methods to solve bilevel mathematical problems can be found in Talbi [31]. Further information about CFL can be found in Kress and Pesch [32] and Drezner [33].

Compared to similar pieces of research, the bilevel mathematical model of this study is more compatible with the real world because new DCs are opened based on budget constraints and each facility in the model can serve more than one customer.

The rest of the paper is as follows: in Section 3, the mathematical model of the problem is defined. In Sections 4 and 5, the proposed meta-heuristic algorithms are described. The procedure of finding an upper bound is explained in Section 6, and computational results are presented in Section 7. Finally, Section 8 includes the conclusion of the research.

3. Problem formulation and description

This study considers two producers that capture the maximum market share by opening new facilities. It is assumed that both competitors currently have several facilities so as to transfer products to their customers. Each customer acquires his products through the most preferable facilities of the leader or the follower. In this study, the relation \prec_j was used to determine the preference rule by customer j to select an open facility; as a result of this definition, the relation $i \succ_j k$ means that by comparing two open facilities, facility i is more attractive than facility k for customer j . Now, suppose that the two competitors start to open new facilities to deal with the growth of market demand.

Suppose that the competitors operate with a non-cooperative behavior based on game theory approach. The main feature of the non-cooperative games is that each player looks for his own benefit. The Nash and Stackelberg equilibriums are two of the most important methods used for many non-cooperative games. The Nash equilibrium is the stable state of a system involving competitive firms that try to optimize their pay-off functions by employing a wide range of strategies. In other words, given the actions of other firms, no firm can improve its position by adopting a

different strategy. The Stackelberg equilibrium consists of two concepts: the leader and the follower. In this case, the optimal strategy of each player can be determined through the Stackelberg equilibrium. This equilibrium is applicable when one of the players can move before the other players and play as the leader. In other words, the leader has more power than the follower, and hence, in this game, the leader makes the first decisions. Afterward, the follower makes his own decisions according to the leader's decision. In the leader-follower environment, the follower chooses the best response to the decision of the leader, and the leader optimizes his objective function according to the follower's response.

Accordingly, the CFL problem will be formulated as a bilevel programming model regarding the following assumptions:

1. The decision-making of the competitors is based on the Stackelberg game;
2. Each facility can serve more than one customer;
3. The demand of each customer can be satisfied with only one facility;
4. Each producer supplies only a part of the market.

We formulate the CFL as follows. The problem has two competitors, a leader and follower that are denoted by l and f , each of whom has two groups of facilities denoted by i and s and a number of customers denoted by j (total number of customers is k). n_l and n_f denote the existing facilities of leader and follower, respectively. There are n potential points where competitors are going to open new facilities among them; to open new facilities, leaders and followers have budget constraint, the total budget of whom is denoted by B_l and B_f , respectively.

Furthermore, f_i denotes the setup cost of opening the i th DC of leader and g_i is the setup cost of the same facility for the follower. $\tilde{\phi}_{ij}$ and $\check{\phi}_{ij}$ denote net profit of delivered products of the existing leader's and follower's DC i to customer j , respectively, and the net profit of delivered products from the new i th DC to customer j is denoted by ϕ_{ij} . Given the above notations and assumptions, the problem is to determine the best location of new facilities of two competitors and assignment of customers to a DC. The following additional notations are used in the bi-level mathematical model of CFL:

$$X_j = \begin{cases} 1 & \text{if leader opens a facility} \\ 0 & \text{otherwise} \end{cases}$$

$$x_{ij} = \begin{cases} 1 & \text{if facility } i \text{ opened by leader is} \\ & \text{assigned to serve customer } j \\ 0 & \text{otherwise} \end{cases}$$

$$\hat{x}_{ij} = \begin{cases} 1 & \text{if existent facility } i \text{ opened by leader} \\ & \text{is assigned to serve customer } j \\ 0 & \text{otherwise} \end{cases}$$

$$Y_j = \begin{cases} 1 & \text{if follower opens a facility} \\ 0 & \text{otherwise} \end{cases}$$

$$y_{ij} = \begin{cases} 1 & \text{if facility } i \text{ opened by follower is} \\ & \text{assigned to serve customer } j \\ 0 & \text{otherwise} \end{cases}$$

$$\hat{y}_{ij} = \begin{cases} 1 & \text{if existent facility } i \text{ opened by follower is} \\ & \text{assigned to serve customer } j \\ 0 & \text{otherwise} \end{cases}$$

According to the definitions of the parameters, an integer bilevel programming model of the CFL problem is presented as follows:

$$\begin{aligned} \max z_l = & \sum_{j=1}^k \left(\sum_{i=1}^n \phi_{ij} x_{ij} \left[1 - \sum_{s=1}^n y_{sj}^* - \sum_{s=1}^n \hat{y}_{sj}^* \right] \right. \\ & \left. + \sum_{i=1}^{n_l} \tilde{\phi}_{ij} x_{ij} \left[1 - \sum_{s=1}^n y_{sj}^* - \sum_{s=1}^n \hat{y}_{sj}^* \right] \right) \\ & - \sum_{i=1}^n f_i X_i, \end{aligned} \tag{1}$$

s.t.

$$\sum_{i=1}^n f_i X_i \leq B_l, \tag{2}$$

$$x_{ij} \leq X_i \quad \forall i, j, \tag{3}$$

$$\sum_{s=1|s \geq j}^{n_l} \hat{x}_{si} + \sum_{s=1|s \geq j}^n x_{sj} \leq 1 \quad \forall j, i, \tag{4}$$

$$\sum_{j=1}^k \hat{x}_{ij} \leq k \quad \forall i, \tag{5}$$

$$\sum_{j=1}^k x_{ij} \leq k \quad \forall i, \tag{6}$$

$$X_j \in \{0, 1\} \quad \forall j, \tag{7}$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j, \tag{8}$$

$$\hat{x}_{ij} \in \{0, 1\} \quad \forall i, j, \tag{9}$$

where $(y_{..}^*, \hat{y}_{..}^*)$ is solved by the following problem:

$$\max z_f = \sum_{j=1}^k \left(\sum_{i=1}^n \phi_{ij} y_{ij} + \sum_{i=1}^{n_f} \tilde{\phi}_{ij} \hat{y}_{ij} \right) - \sum_{i=1}^n g_i y_i, \tag{10}$$

s. t.

$$\sum_{j=1}^n g_j \cdot Y_j \leq B_f, \tag{11}$$

$$X_j + Y_j \leq 1 \quad \forall j, \tag{12}$$

$$y_{ij} \leq Y_i \quad \forall i, j, \tag{13}$$

$$\begin{aligned} \sum_{s=1|s \geq j}^{n_f} \hat{y}_{si} + \sum_{s=1|s \geq j}^n y_{sj} + \sum_{s=1|s \geq j}^{n_l} \hat{x}_{si} \\ + \sum_{s=1|s \geq j}^n x_{sj} \leq 1 \quad \forall j, i, \end{aligned} \tag{14}$$

$$\sum_{j=1}^k \hat{y}_{ij} \leq k \quad \forall i, \tag{15}$$

$$\sum_{j=1}^k y_{ij} \leq k \quad \forall i, \tag{16}$$

$$Y_j \in \{0, 1\} \quad \forall j, \tag{17}$$

$$y_{ij} \in \{0, 1\} \quad \forall i, j, \tag{18}$$

$$\hat{y}_{ij} \in \{0, 1\} \quad \forall i, j. \tag{19}$$

Eq. (1) determines the objective function of the leader with respect to lost profit due to customer supplied by the new or existing facilities of the follower. The leader’s budget constraint on opening new centers is presented by Eq. (2). Constraint (3) ensures product distribution through opened facilities. Eq. (4) shows that each customer can satisfy his/her demand from one of the facilities of the leader. Constraints (5) and (6) indicate that both existing and new facilities of leader can supply all customers. Eqs. (7) to (9) represent the status of the upper-level decision variables. Eq. (10) states the objective function of the follower producer aimed at maximizing profit through opening new facilities and also through existing DCs. Eq. (11) represents the budget constraint of the follower. Eq. (12) ensures that at every potential point, only the leader or follower can open a new DC. Constraint (13) acts like Constraint (3), but at the upper level for each customer. The product will only be delivered by one of the two competitors considering preference rule for each customer as stated by Eq. (14).

Constraints (15) and (16) act like Constraints (5) and (6), but at the upper level. Constraints (17)–(19) define the lower-level decision variables. In this model, firstly, the upper-level decision-maker determines his strategy, followed by the strategy of the lower-level decision-maker. Awareness of this strategy determines one’s policy of optimizing the objective function and lastly, the optimum response by the leader is determined based on the best response by the follower.

4. The proposed GA

GA is one of the most popular classes of Evolutionary algorithms that was developed by J. Holland in the 1970s (University of Michigan, USA) to understand the adaptive processes of natural systems. Then, it was applied to optimization and machine learning in the 1980s. GA usually applies a crossover operator to solutions that play a major role as well as a mutation operator that randomly modifies the individual contents to promote diversity.

4.1. Chromosome representation

One of the most important factors in the successful implementation of the GA is designing a more suitable chromosome structure. Thus, in this paper, a hybrid structure was used to show the chromosome structure in order to describe the different available solutions to the problem. The first section of the chromosome structure indicates the selection of facilities by two competitors and as can be seen in Figure 1, the first section of the chromosome structure itself is composed of two sub-chromosomes of leader and follower. Length of chromosomes of the first section is considered twice the number of potential centers. The selection of facilities is performed based on leader-follower game and that in the mathematical model of the problem, at the lower level, the follower-producer facilities are the function of the leader DCs; therefore, initially, in the first chromosome structure, the leader facilities are determined by a binary representation; then, among the remaining DCs, the follower proceeds to select his/her own facilities among the remaining potential facilities, which are not very attractive for the leader. It should be noted that in each chromosome, leader and follower select a facility among the potential points

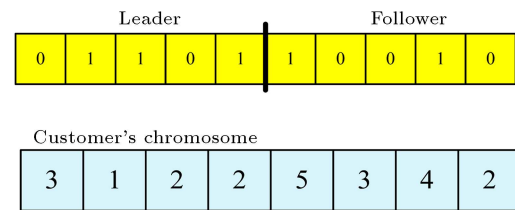


Figure 2. Sample chromosomes for five potential points and eight customers.

consecutively. After that, for each of leader-follower chromosomes, a chromosome with a length equivalent to the number of customers is produced and among the selected DCs, for each customer, a DC with the highest attractiveness will be selected. In this phase, a DC may be selected by one or several numbers of customers.

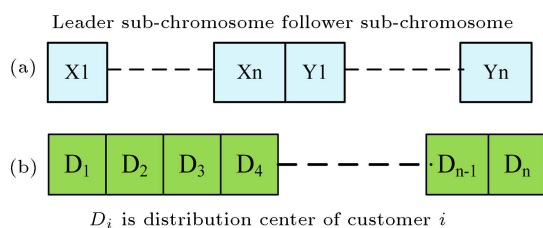
For example, suppose that there are 2 competitors and 8 customers and the former are going to open new facilities among 5 potential points (suppose that leader and follower have only one DC yet). Figure 2 shows a sample produced chromosome for this situation; this figure shows that the 2nd, 3rd, and 5th facilities belong to the leader and the first and fourth facilities to the follower. Following the selection of new facilities, the number of all facilities will be changed such that the existing facilities of leader are considered at first and start from the value of 1; then, the new facilities of leader are considered and the existing and new facilities of the follower are assumed next. Thus, in the above-mentioned example, facilities 1 to 4 belong to the leader and facilities 5 to 7 belong to the follower. Currently, the customers are going to select the best facilities by preference rules such as distance. In our example, facility 3 as one of the new facilities of the leader is used to supply customers 1 and 6 with products, and vice versa. It should be noted that the initial population is generated randomly.

4.2. Deal with infeasible chromosomes

As mentioned above, the randomly created initial population may result in the generation of infeasible chromosomes. Hence, after dealing with this situation, a local search procedure applied by Ahmadizar and Soltanpanah [34] was presented to adopt our approach. Note that, in this paper, a chromosome is considered to be feasible if the cost of sub-chromosome of the leader or follower or both of them is less than or equal to the total budget of them. The proposed local search process is shown in Figure 3.

4.3. Evaluation

After generating the initial population, the generated chromosomes are evaluated by fitness function, which is considered as the main objective function in most optimization problems. In this paper, the upper level objective function in the mathematical model is considered as the fitness function. To compute the



D_i is distribution center of customer i

Figure 1. Proposed chromosomal structures: (a) Facilities of competitors and (b) facilities of customers.

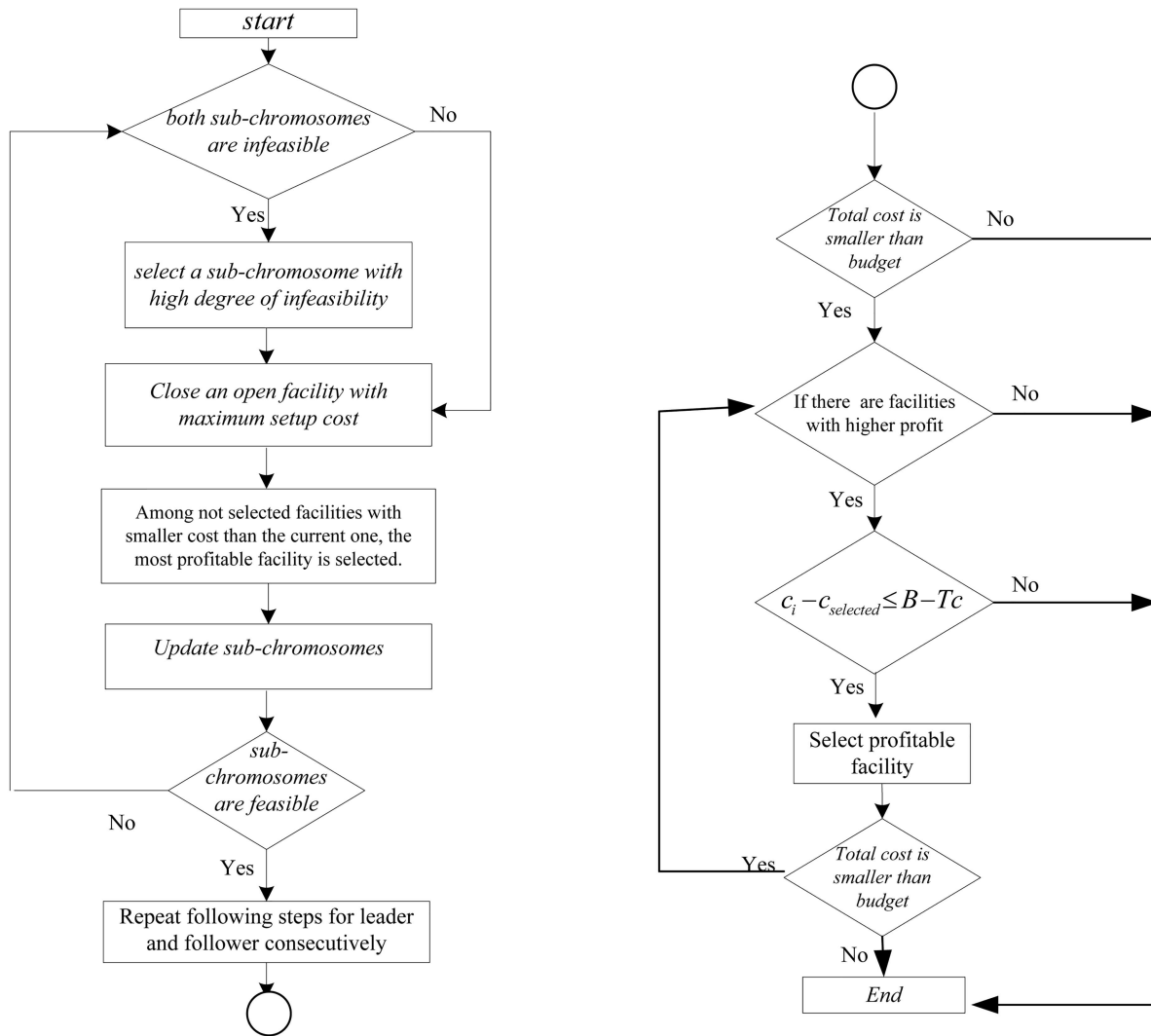


Figure 3. Local search procedure.

fitness function of each chromosome, the setup costs of facilities are calculated based on the first section of the leader-follower’s chromosome, meaning that facilities are selected by the leader. In order to determine the revenue of each DC, in the second chromosome structure which shows the facilities selected by the customers, the one which is selected by the leader in the chromosome structure of leader-follower will be considered for computation. Thus, making a profit on the selection of facilities is considered as the fitness function of every member of population.

4.4. Selection

In each iteration of the GA, some chromosomes of the current population are selected to generate a new generation by crossover and mutation operators. Although there are various methods for selecting the chromosomes for reproduction, some of them may select the chromosomes with the best fitness function and this selection method spoils the chance of

achieving good solutions by the chromosomes with an inappropriate fitness function. Hence, in this study, the Roulette Wheel will be used in order to generate a new generation, which is described as follows.

Normalize the fitness function of each chromosomes by using the following relations (fitness value of each chromosome denoted by $F(ch)$, the best value of fitness function is F_{max} , and $F'(ch)$ and $RW(ch)$ denote the normalized fitness function of each chromosome and the cumulative value of each chromosome, respectively):

$$1. F'(ch) = F_{max} - F(ch);$$

$$1. RW(ch) = \frac{\sum_{i=1}^{ch} F(i)}{\sum_{j=1}^{pop_size} F'(j)}.$$

For each chromosome, generate a random number

between 0 and 1 (RND) and then, select chromosome x based on the following rule:

if
 $RW(ch - 1) < RND_1 \leq RW(ch)$ and $RW(0) = 0$
then
 select chromosome ch
End if

In this paper, the following procedure is used in order to apply crossover and mutation.

Generate a random number (RND_2); if the random number satisfies ($RND_2 \leq \Psi_{LF}$), then apply the mutation and crossover to leader-follower chromosomes, otherwise apply them to the chromosomes of customers (Ψ_{LF} is the probability of applying crossover and mutation for leader-follower sub chromosomes).

4.5. Crossover and mutation

Crossover plays an important role in exchanging information among chromosomes. It leads to an effective combination of partial solutions in other chromosomes and it speeds up the search procedure early in the generation [35].

Two scenarios are considered to perform crossover. Based on the following rules and considering crossover probability for each scenario (SC_1 and SC_2 ; without loss of generality, we consider that $SC_1 + SC_2 = 1$), one of the two scenarios is applied to the parent chromosomes:

1. Generate a random number (RND_3) and if ($RND_3 \leq SC_1$) is satisfied, go to Step 2; otherwise, go to Step 4;
2. Select two genes randomly;
3. Change the pre- and post-genes of the two selected

genes in parent chromosomes; otherwise, go to the Step 4;

4. Switch the genes between two selected genes in the parent chromosomes.

Figure 4 shows an example of reproduction in the two mentioned scenarios for both sub-chromosomes. In the chromosomes relating to the selection of facilities by customers, the crossover is applied, similar to what was applied for leader-follower chromosomes.

The mutation takes place after performing a crossover to prevent all solutions from falling into a local optimum. Moreover, mutation can help generate those chromosomes that are not obtainable by the crossover operator. Figure 5 shows offspring which is generated by mutation of its parent chromosome.

The generated chromosomes may become infeasible after applying crossover and mutation. As an example, in the first scenario shown in Figure 4, both competitors proceed to reopen centers 2 and 3 in the first offspring and center 4 in the second offspring. Besides, in the second scenario, either competitor in the first and second offspring selects centers 4, 2 and 3, respectively. Given that it is highly probable to encounter infeasible chromosomes after applying crossover, the probability of applying crossover is low from the very beginning. Note that after applying crossover and mutation, if needed, the chromosomes associated with the selection of facilities will be updated by the customers.

It should be noted that after applying both operators and producing a new generation, some of the generated chromosomes may be infeasible; in this case, a simple local search is applied to make them feasible, as explained in the previous section.

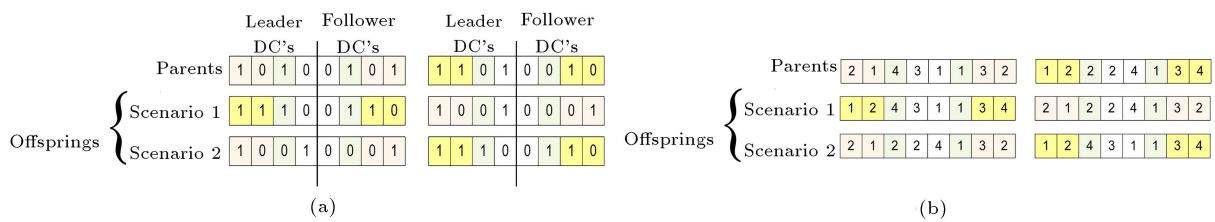


Figure 4. Crossover operator of (a) leader-follower's chromosome and (b) customer's chromosome.

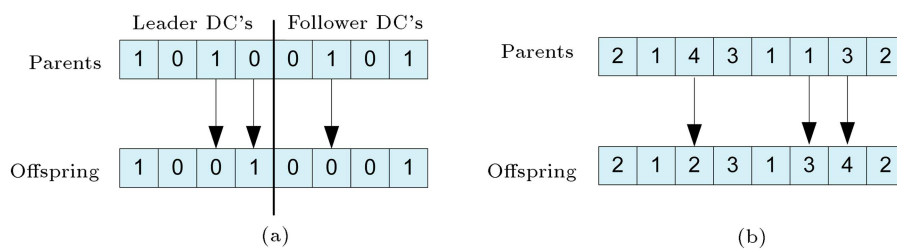


Figure 5. Mutation operator of (a) leader-follower's chromosome and (b) customer's chromosome.

<p>A Procedure</p> <p>Initialize parameter</p> <p>Leader-Follower chromosomes (LF)</p> <p><i>pop_size</i> ← population size <i>crsp1</i> ← crossover probability in first scenario <i>crsp2</i> ← crossover probability in second scenario <i>mtp1</i> ← mutation probability</p> <p>Customers Chromosomes (CC)</p> <p><i>crsp3</i> ← crossover probability in first scenario <i>crsp4</i> ← crossover probability in second scenario <i>mtp2</i> ← mutation probability <i>itter</i> ← maximum number of iteration</p> <p>Start</p> <p>Generate LF initial population <i>pop_size</i> times;</p> <p>For <i>i</i>=1 to <i>pop_size</i>:</p> <p>Generate a chromosome for customers;</p> <p>next <i>i</i></p> <p><i>z_{opt}</i> ← 0;</p> <p><i>DC</i> = ∅</p> <p><i>Allocation_{opt}</i> = ∅</p> <p><i>z_{best}</i> ← 0;</p> <p><i>Allocation_{best}</i> ← ∅</p>	<p>For <i>k</i>=1 to <i>itter</i></p> <p>For each individual:</p> <p>Calculate the fitness function;</p> <p><i>z</i> ← fitness function;</p> <p>if (<i>z</i> > <i>z_{best}</i>) then</p> <p><i>z_{best}</i> ← <i>z</i>;</p> <p><i>Allocation_{best}</i> ← CC chromosom</p> <p>end if</p> <p>Parent Selection;</p> <p>Generate offsprings based on crossover and mutation operator;</p> <p>Add offsprings to population</p> <p>Do local search</p> <p>update population</p> <p>next <i>itter</i></p> <p>print <i>DC</i>, <i>Allocation_{opt}</i>, <i>z_{opt}</i></p> <p>End</p>
--	--

Figure 6. Pseudocode of the proposed genetic algorithm.

A general framework of the proposed GA based on its description is given below:

1. Generate a set of random solutions and do local search if necessary (initial population);
2. Calculate the fitness function of each individual (evaluation);
3. Select parents among the individuals (selection);
4. Apply crossover and mutation to the parents and do local search if necessary (crossover and mutation);
5. Select individuals for the next iteration (update population);
6. Apply local search to improve some individuals;
7. Repeat steps 2 to 5 until the termination criterion is met.

The pseudocode of the proposed GA is shown in Figure 6.

5. Hybrid genetic-ant colony algorithm

In this section, a hybrid procedure composed of GA and Ant Colony Optimization (ACC) algorithm is elaborated. In our proposed hybrid procedure, GA is used to locate new facilities for two competitors and the best assignment of facilities to customer is determined

by ant colony algorithm. The framework of the proposed hybrid algorithm, which will be explained in this section, is exactly similar to that of the GA described in the previous section, except that each customer selects his/her DC using ant colony algorithm. Thus, the structure of the proposed ant colony will be explained at the first step; then, the framework of the proposed hybrid genetic-ant colony algorithm will be presented.

5.1. Selection rule and pheromone updating

Artificial ants use the law of probability inspired by the behavior of natural ants for consecutive selection of facilities for each customer and construction of a solution with respect to the data obtained via a pheromone path, which is updated over time. Based on the system designed by Dorigo and Gambardella [36], ant customer *j* uses the following rule to select the DC *i* with probability of q_0 :

$$i = \arg \max \left\{ \tau_{ij} \eta_{ij}^\beta \right\}. \quad (20)$$

With the probability of $(1 - q_0)$, customer *j* selects DC *i* using the following equation:

$$p_{ij} = \frac{\tau_{ij} \eta_{ij}^\beta}{\sum_{i=1}^N \tau_{ij} \eta_{ij}^\beta}, \quad (21)$$

where τ_{ij} denotes the pheromone trail, η_{ij} represents the heuristic information of DC j , and β is a value between 0 and 1, signifying the importance of η_{ij} .

Parameter η_{ij} is determined according to the ranking of DC:

$$\eta_{ij} = \frac{1}{R_{ij}}. \tag{22}$$

R_{ij} represents the ranking of DC i from the perspective of customer j . After constructing an assignment, an ant modifies the pheromone intensity of path between DC and customer by applying the local updating rule as follows:

$$\tau_{ij} = (1 - \rho)\tau_{ij} + \rho\tau_0. \tag{23}$$

τ_0 is the initial value of the pheromone trails and ρ is the evaporation rate of the pheromone trail ($0 \leq \rho \leq 1$).

Given that ants have constructed their solution, the intensity of pheromone on each path between a DC and a customer is changed by applying the global updating rule:

$$\tau_{ij} = (1 - \rho)\tau_{ij} + \rho\tau_0 \left(\frac{pr_{ij}}{pr_{best}} \right). \tag{24}$$

pr_{best} is the maximum profit of allocation of DC to a customer so far. The pseudocode of the proposed hybrid algorithm is shown in Figure 7.

In a preliminary experiment, a variety of combinations of the parameter values were tested to set the numeric parameters of the algorithm. Experimental results showed that the best performance of the proposed GA and ACO was obtained using the parameters listed in Table 1. It should be noted that the two proposed algorithms stopped after 1000 iterations.

6. Determining an upper bound for the CFL problem

The optimality of solution of the proposed mathematical model is not clear and, consequently, for evaluating the efficiency of the proposed algorithm, an upper bound is developed and the computational results will be compared with this value.

Theorem. *The upper bound for the objective function of the competitive location problem can be determined by solving the following problem.*

Proof. Please see the Appendix (the parameters of the following problem were defined in Section 3);

$$\max \sum_{j=1}^k \left(\sum_{i=1}^n \phi_{ij} x_{ij} + \sum_{i=1}^{n_l} \tilde{\phi}_{ij} x_{ij} \right) - \sum_{i=1}^n f_i X_i, \tag{25}$$

s.t.

Table 1. Parameters of the proposed algorithms.

Parameter	Value
GA parameters	
Population size	40
<i>crsp1</i>	0.4
<i>crsp2</i>	0.2
<i>mtp1</i>	0.25
<i>crsp3</i>	0.7
<i>crsp4</i>	0.65
<i>mtp2</i>	0.2
Ψ_{LF}	0.3
SC_1	0.5
ACO parameters	
Ant	20
τ_0	0.000000001
ρ	0.1
β	1
q_0	0.9

$$\sum_{i=1}^n f_i X_i \leq B_l, \tag{26}$$

$$x_{ij} \leq X_i \quad \forall i, j, \tag{27}$$

$$\sum_{s=1|s \geq j}^{n_l} \hat{x}_{si} + \sum_{s=1|s \geq j}^n x_{sj} \leq 1 \quad \forall j, i, \tag{28}$$

$$\sum_{j=1}^k \hat{x}_{ij} \leq k \quad \forall i, \tag{29}$$

$$\sum_{j=1}^k x_{ij} \leq k \quad \forall i, \tag{30}$$

$$X_j \in \{0, 1\} \quad \forall j, \tag{31}$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j, \tag{32}$$

$$\hat{x}_{ij} \in \{0, 1\} \quad \forall i, j. \tag{33}$$

7. Computational results

In Table 2, 20 test problems were generated to evaluate the performance of the developed algorithm; then, an upper bound of each problem was determined by the mathematical programming model presented in Section 6.

The proposed algorithms were coded in VB 6.0 and run ten times for each test problem by a computer featuring Core i5 processor, 4 GB RAM, and a 64-bit

```

Hybrid GA and ACO
Initialization Step
  a. GA parameters
    op_size ← population size
    crsp1(crossover probability in first scenario)
    crsp2(crossover probability in second scenario)
    mtp1(mutation probability)

  b. ACO parameters
    ant(Number of ants)
     $\tau_0$  (initial pheromone trail)
     $\rho$ (evaporation rate)
     $q_0$  (acceptance probability)
  zbest ← 0;
  Max_iter ← maximum number of iterations;
// start
  Generate LF initial population pop_size times;
  For itter =1 to Max_iter
  For i=1 to pop_size:
    C={all customers};
  While C =  $\emptyset$  Do
    Step 1. Given the number of customers, repeat the following steps:
      Step 1-1. Use the selection rule to make an assignment;
      Step 1-2. Remove selected customer from C ;
  End While
    zbest ← upper level objective function;

    if z ≥ zopt then
      zopt ← z;
      DC ← chromosom(i);
      Allocation_opt ← ACO as si gnment;

    end if
  Alter the pheromone trail with global updating rules;

  next i

  if z ≥ zbest then
    zbest ← z;
  end if
next itter

end while
Print zbest;

```

Figure 7. Pseudocode of the proposed hybrid genetic algorithm and ant colony optimization.

Windows 7 operating system. The upper bound of each test problem was determined with Lingo 9.0 software.

In all instances, the profit of the existing and new facilities and the cost of setting up a new DC were generated randomly from the intervals [500,2500], [1000,3500], and [2000,4000] in that order.

To evaluate the efficiency of the proposed algorithms, the results should be compared with an optimum solution; for this reason, similar to the studies of Qi et al. [13] and Nasiri et al. [15], the

enumeration method was used to find an optimum solution to small-sized problems (Nos. 1 to 5), which is reported in Table 3. The results showed that both meta-heuristics were able to find an optimal solution; the optimal solution was found, but the procedure is time consuming and not affordable; therefore, the remaining test problems were solved by two meta-heuristic algorithms, the results of which are shown in Table 4.

Figures 8–10 clearly show that while the proposed

Table 2. Characteristics of randomly generated problems.

Problem no.	Size	Number of existing DCs		Number of potential DCs	Number of customers
		Leader DCs	Follower DCs		
1	Small	2	2	4	8
2		2	2	4	10
3		2	2	5	12
4		2	2	5	14
5		4	4	5	16
1	Medium	4	4	6	18
2		5	5	6	20
3		5	5	6	22
4		6	6	6	25
5		6	6	6	30
6		15	15	15	70
7		20	20	15	100
8		25	25	15	200
9		30	30	15	300
10		35	35	15	400
1	Large	70	70	20	750
2		150	150	25	1500
3		175	175	30	2000
4		200	200	50	4000
5		300	300	75	5000

Table 3. Results of small-sized problems.

Problem no.	Optimal solution (Leader objective function)	Proposed algorithms				Gap (%)	
		Leader objective function		Time (second)		GA	GA-ACO
		GA	GA-ACO	GA	GA-ACO		
1	7341	7341	7341	< 1 S	< 1 S	0	0
2	7896	7896	7896	< 1 S	< 1 S	0	0
3	9652	9407	9407	< 1 S	< 1 S	0	0
4	12427	12243	12243	< 1 S	< 1 S	1.48	1.48
5	17710	17603	17710	< 1 S	< 1 S	0.6	0

algorithm was capable of achieving good solutions within a reasonable computational time, a better solution was obtained by applying the proposed hybrid algorithm.

The comparison of the results of GA and GA-ACO (hybrid algorithm) showed that employing GA and ACO as a hybrid algorithm would improve the value of the objective function, while the CPU time

would be prolonged. Since the results of meta-heuristics cannot be interpreted as an optimum solution, an efficient upper and lower bound was calculated for the CFL problem, as discussed in this study. Thus, these upper and lower bounds could be used to solve the problem using such methods as branch and bound; in addition, the computational time of such a method may be shortened using these bounds.

Table 4. Computational results of test problems.

Problem no.	Upper level problem				GA		ACO		GAP (%)		
	Upper bound	Integer	Constraints	CPU time (s)	Objective function	CPU time (s)	Objective function	CPU time (s)	GA	GA-ACO	
Medium-sized problem	1	45386	187	138	2	22234	< 1 S	22367	1.34	0.51	0.51
	2	49349	227	153	3	28512	< 1 S	28548	1.76	0.42	0.42
	3	54585	249	167	3	36327	1.23	36463	2.31	0.33	0.33
	4	64654	307	189	4	47970	1.6	48519	2.76	0.26	0.25
	5	75261	367	224	5	53601	1.87	55213	4.2	0.29	0.27
	6	194610	2116	1152	7	99876	4.26	105732	6.1	0.49	0.46
	7	267441	3516	1637	10	163467	4.49	183819	8	0.39	0.31
	8	519107	8016	3242	12	368235	4.94	438967	9.3	0.29	0.15
	9	747837	13516	4847	26	473478	5.54	595361	10.5	0.37	0.20
	10	859444	20016	6452	24	594905	8.62	634605	12.4	0.31	0.26
Medium-sized problem	1	1855966	67521	15842	46	884670	13.45	1197219	15.3	0.52	0.35
	2	3730113	262526	39177	606	1813270	33.98	2197645	41.2	0.51	0.41
	3	5094872	410031	62207	> 3600	2123641	41.5	2625864	49.71	0.58	0.48
	4	10272496	1000051	204252	> 3600	3212916	73.65	3934209	99	0.69	0.62
	5	29441430	1875076	380377	> 3600	11256000	135.28	15263208	180.17	0.62	0.48

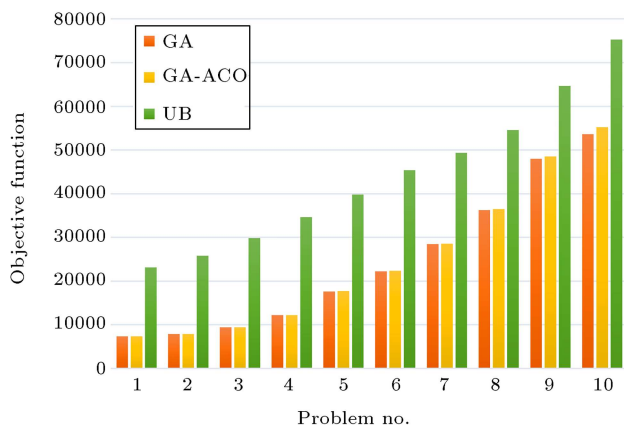


Figure 8. Objective function of the proposed algorithms for small-sized problems.

8. Conclusion

This paper investigated the competitive facility location problem of two competitors. Because of the imbalance of power between the two, a leader-follower game was developed for them to open new facilities in order to maximize their profit. The leader established new facilities to exploit the new high market demand and the follower reacted by opening his/her own new facilities. These two competitors chose new facilities among several potential locations based on the rules of the Stackelberg game. The problem was described as a bilevel mathematical model and then, because of its high complexity, two metaheuristic algorithms were used to solve the problem. To verify

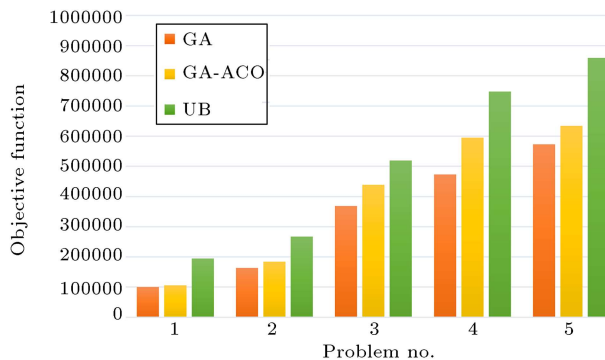


Figure 9. Objective function of the proposed algorithms for medium-sized problems.

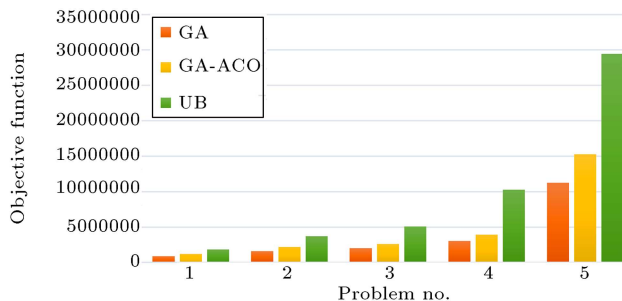


Figure 10. Objective function of the proposed algorithms for large-sized problems.

these algorithms, an enumeration technique applied to small-sized problems was used to find an optimal solution, while an upper bound was developed for the problem, as well. The comparison showed the

impressive performance of the proposed approaches in obtaining optimal solutions to small-sized problems. Then, anyone can assure that the solution to medium- and large-scale problems is the best result for each test problem. Future research directions include examination of other subjects such as relocating one or more than one distribution center, developing better upper and lower bounds with different approaches, providing exact methods to solve the problem, and studying the problem by considering multi-follower and other game disciplines. Moreover, given that the obtained results of the proposed algorithm can be interpreted as a good lower bound, designing a branch and bound algorithm can lead to finding an optimal solution to medium- and large-scale problems.

Acknowledgements

The authors would like to thank the anonymous referees for their constructive comments and suggestions which have been very helpful in improving the presentation of this paper.

References

- Amirtaheri, O., Zandieh, M., and Dorri, B. "A bi-level programming model for decentralized manufacturer-distributor supply chain considering cooperative advertising", *Scientia Iranica, Transactions E: Industrial Engineering*, **25**(2), pp. 891–910 (2018).
- Pal, D. and Sarkar, J. "Spatial competition among multi-store firms", *International Journal of Industrial Organization*, **20**(2), pp. 163–190 (2002).
- Aboolian, R., Berman, O., and Krass, D. "Competitive facility location model with concave demand", *European Journal of Operational Research*, **181**(2), pp. 598–619 (2007).
- Beresnev, V.L. "Upper bounds for objective functions of discrete competitive facility location problems", *Journal of Applied and Industrial Mathematics*, **3**, pp. 419–432 (2009).
- Saidani, N., Chu, F., and Chen, H. "Competitive facility location and design with reactions of competitors already in the market", *European Journal of Operational Research*, **219**(1), pp. 9–17 (2012).
- Ashtiani, M.G., Makui, A., and Ramezani, R. "A robust model for a leader-follower competitive facility location problem in a discrete space", *Applied Mathematical Modelling*, **37**(1–2), pp. 62–71 (2013).
- Beresnev, V. "Branch-and-bound algorithm for a competitive facility location problem", *Computers & Operations Research*, **40**(8), pp. 2062–2070 (2013).
- Calvete, H.I., Gale, C., and Iranzo, J.A. "Planning of a decentralized distribution network using bilevel optimization", *Omega*, **49**, pp. 30–41 (2014).
- Rahmani, A. and Mirhassani, S.A. "Lagrangean relaxation-based algorithm for bi-level problems", *Optimization Methods and Software*, **30**(1), pp. 1–14 (2015).
- Mirhassani, S.A., Raeisi, S., and Rahmani, A. "Quantum binary particle swarm optimization-based algorithm for solving a class of bi-level competitive facility location problems", *Optimization Methods Software*, **30**(4), pp. 756–768 (2015).
- Zhang, Y., Synder, L.V., Ralphs, T.K., and Zhaojie, X. "The competitive facility location problem under disruption risks", *Transportation Research Part E*, **93**, pp. 453–473 (2016).
- Saranwong, S. and Likasiri, C. "Bi-level programming model for solving distribution center problem: A case study in northern Thailand's sugarcane management", *Computers & Industrial Engineering*, **103**, pp. 26–39 (2017).
- Qi, M., Xia, M., Zhang, Y., and Miao, L. "Competitive facility location problem with foresight considering service distance limitations", *Computers and Industrial Engineering*, **112**, pp. 483–491 (2017).
- Kung, L-C. and Liao, W-H. "An approximation algorithm for a competitive facility location problem with network effects", *European Journal of Operational Research*, **267**(1), pp. 176–186 (2017).
- Nasiri, M.M., Mahmoodian, V., Rahbari, A., and Farahmand, S. "A modified genetic algorithm for the capacitated competitive facility location problem with the partial demand satisfaction", *Computers and Industrial Engineering*, **124**, pp. 435–448 (2018).
- Beresnev, V. and Melnikov, A. "Exact method for the capacitated competitive facility location problem", *Computers & Operation Research*, **95**, pp. 73–82 (2018).
- Sadjadi, S.J., Gorji Ashtiani, M., Makui, A., and Ramezani, R. "A mathematical model for competitive location problem with product selection", *Scientia Iranica, Transaction E: Industrial Engineering*, **27**(4), pp. 2157–2176 (2020). DOI: 10.24200/sci.2018.51736.2337
- Bilir, C., Ekici, S.O., and Ulengin, F. "An integrated multi-objective supply chain network and competitive facility location model", *Computers and Industrial Engineering*, **108**, pp. 136–148 (2017).
- Wang, S.C. and Chen, T.C. "Multi objective competitive facility location problem with distance based attractiveness and its best non-dominated solution", *Applied Mathematical Modeling*, **47**, pp. 785–795 (2017).
- Konak, A., Konak, S.A., and Synder, L. "A multi objective approach to the competitive facility location problem", *Procedia Computers Science*, **108**, pp. 1434–1442 (2017).

21. Beresnev, V.L. and Mel'nikov, A.A. "Approximate algorithms for the competitive facility location problem", *Journal of Applied and Industrial Mathematics*, **5**(2), pp. 180–190 (2011).
22. Beresnev, V.L. and Mel'nikov, A.A. "The branch-and-bound algorithm for a competitive facility location problem with the prescribed choice of suppliers", *Journal of Applied and Industrial Mathematics*, **8**(2), pp. 177–189 (2014).
23. Wang, G., Ziyou, G., and Zhongping, W. "A global optimization algorithm for solving the bi-level linear fractional programming problem", *Computers & Industrial Engineering*, **63**(2), pp. 428–432 (2012).
24. Oduguwa, V. and Roy, R. "Bi-level optimisation using genetic algorithm", In *IEEE International Conference on Artificial Intelligence Systems Divnomorskoe, Russia*, pp. 322–327 (2002).
25. Yin, Y. "Genetic-algorithms-based approach for bilevel programming models", *J. Transp. Eng.*, **126**, pp. 115–120 (2000).
26. Hejazi, S.R., Memariani, A., Jahanshahloo, G., and Sepehri, M.M. "Linear bilevel programming solution by genetic algorithm", *Comput. Oper. Res.*, **29**(13), pp. 1913–1925 (2002).
27. Wen, U.P. and Huang, A.D. "A simple Tabu search method to solve the mixed-integer linear bilevel programming problem", *Europ. J. Oper. Res.*, **88**(3), pp. 563–571 (1996).
28. Rajesh, J., Gupta, K., Kusumakar, H.S., Jayaraman, V.K., and Kulkarni, B.D. "A Tabu search based approach for solving a class of bilevel programming problems in chemical engineering", *J. Heuristics*, **9**(4), pp. 307–319 (2003).
29. Wan, Z., Wang, G., and Sun, B. "A hybrid intelligent algorithm by combining particle swarm optimization with chaos searching technique for solving nonlinear bilevel programming problems", *Swarm and Evolutionary Computation*, **8**, pp. 26–32 (2013).
30. Kou, R.J., Lee, Y.H., Zulvia, F.E., and Tien, F.C. "Solving bi-level linear programming problem through hybrid of immune genetic algorithm and particle swarm optimization algorithm", *Applied Mathematics and Computation*, **266**, pp. 1013–1026 (2015).
31. Talbi, E.-G., *Metaheuristics for Bi-level Optimization*, Springer Publishing Company, Incorporated (2013).
32. Kress, D. and Pesch, E. "Sequential competitive location on networks", *European Journal of Operational Research*, **217**(3), pp. 483–499 (2012).
33. Drezner, T. "A review of competitive facility location in the plane", *Logistics Research*, **7**, pp. 1–12 (2014).
34. Ahmadizar, F. and Soltanpanah, H. "Reliability optimization of a series system with multiple-choice and budget constraints using an efficient ant colony approach", *Expert Systems with Applications*, **38**(4), pp. 3640–3646 (2011).
35. Farughi, H., Yegane, B.Y., and Fathian, M. "A new critical path method and a memetic algorithm for flexible job shop scheduling with overlapping operations", *Simulation: Transaction of the Society for Modeling and Simulation International*, **89**(3), pp. 264–277 (2012).
36. Dorigo, M. and Gambardella, L.M. "Ant colony system: a cooperative learning approach to the traveling salesman problem", *IEEE Transactions on Evolutionary Computation*, **1**(1), pp. 53–66 (1997).

Appendix

Determination of the upper bound for the problem

The upper bounds of the problem instances were determined by a method similar to the one proposed by Beresnev and Mel'nikov [21]. Each customer could be supplied by one of the new or existing facilities of one of the two competitors.

We show each acceptable solution of the upper-level problem with an ordered triple of $X = (x_i, x_{ij}, \hat{x}_{ij})$, for which the lower-level problem will have an optimum solution in the form of $Y^* = (y_i^*, y_{ij}^*, \hat{y}_{ij}^*)$. In addition, we denote each acceptable solution of the follower problem with $\tilde{Y} = (y_i, y_{ij}, \hat{y}_{ij})$. An admissible solution to the above leader-follower problem will be expressed as (X, \tilde{Y}) , the value of which can be determined by replacement in Eq. (1) and is represented by $\Omega(X, \tilde{Y})$. Likewise, the optimum solution of the problem is denoted by $\Omega(X^*, \tilde{Y})$ and for each acceptable solution $\Omega(X, \tilde{Y})$, the relationship $\Omega(X^*, \tilde{Y}) \geq \Omega(X, \tilde{Y})$ holds.

To maximize the profit, it is assumed that the follower and producer select locations of lower importance to the leader, supposing that the two competitors do not have an equal strength; therefore, a non-cooperative game will be played between the two competitors. Thus, for each arbitrary solution X , there is an optimum non-cooperative solution in the form of \tilde{Y} , which is applied to the relationship $\Omega(X, \tilde{Y}) \geq \Omega(X, \tilde{Y})$. According to the above definition, an acceptable non-cooperative solution is defined in the form of (X, \tilde{Y}) and an optimum non-cooperative solution is defined in the form of (X^*, \tilde{Y}^*) . We show the optimal value of the objective function with $\Omega(X^*, \tilde{Y}^*)$.

Lemma 1. For each possible solution to the problem, the following relationship is true.

$$\sum_{j=1}^k \left(\sum_{i=1}^n \phi_{ij} x_{ij} \left[1 - \sum_{s=1}^n y_{sj}^* - \sum_{s=1}^{n_f} \hat{y}_{sj}^* \right] + \sum_{i=1}^{n_l} \tilde{\phi}_{ij} \hat{x}_{ij} \left[1 - \sum_{s=1}^n y_{sj}^* - \sum_{s=1}^n \hat{y}_{sj}^* \right] \right) - \sum_{i=1}^n f_i X_i \leq k \left(\max_i \{ \phi_{ij} x_{ij} \} + \max_s \{ \tilde{\phi}_{ij} \hat{x}_{ij} \} \right). \tag{A.1}$$

Proof. If customer j is served by one of the follower's centers, i.e., $\phi_{ij} x_{ij} = 0$ and $\tilde{\phi}_{ij} \hat{x}_{ij} = 0$, the proof is completed, because one of the two equations $\sum_{s=1}^n y_{sj}^* = 1$ or $\sum_{s=1}^{n_f} \hat{y}_{sj}^* = 1$ will be true; suppose that for a set of potential locations I_{new} and a set of existent locations I_{exist} , one of the equations $\phi_{ij} x_{ij} > 0$ (for all $i \in I_{new}$) or $\tilde{\phi}_{i'j} \hat{x}_{i'j} > 0$ (for all $i' \in I_{exist}$) is true; in this case, for new and existing distribution centers i and p , we will have:

$$\sum_{j=1}^k \left(\sum_{i=1}^n \phi_{ij} x_{ij} \left[1 - \sum_{s=1}^n y_{sj}^* - \sum_{s=1}^{n_f} \hat{y}_{sj}^* \right] \right) \leq k \left(\max_j \{ \phi_{ij} x_{ij} |_{x_{ij}=1} \} \right) \leq k \left(\max_j \{ \phi_{ij} x_{ij}, \tilde{\phi}_{ij} \hat{x}_{ij} |_{\hat{x}_{pj} + x_{ij}=1} \} \right) \leq k \left(\left(\max_j \{ \phi_{ij} x_{ij} \} + \max_j \{ \tilde{\phi}_{ij} \hat{x}_{ij} \} \right) |_{\hat{x}_{pj} + x_{ij}=1} \right), \tag{A.2}$$

and :

$$\sum_{j=1}^k \left(\sum_{i=1}^n \tilde{\phi}_{pj} \hat{x}_{pj} \left[1 - \sum_{s=1}^n y_{sj}^* - \sum_{s=1}^{n_f} \hat{y}_{sj}^* \right] \right) \leq k \left(\max_j \{ \tilde{\phi}_{pj} \hat{x}_{pj} |_{\hat{x}_{pj}=1} \} \right) \leq k \left(\max_j \{ \phi_{ij} x_{ij}, \tilde{\phi}_{pj} \hat{x}_{pj} |_{\hat{x}_{pj} + x_{ij}=1} \} \right) \leq k \left(\left(\max_j \{ \phi_{ij} x_{ij} \} + \max_j \{ \tilde{\phi}_{ij} \hat{x}_{ij} \} \right) |_{\hat{x}_{pj} + x_{ij}=1} \right). \tag{A.3}$$

Because each customer j is covered by only one of the existing or new distribution centers, the proof is complete. Thus, the quantity of:

$$\max_{j=1}^k \left(\max_j \{ \phi_{ij} x_{ij} \} + \max_s \{ \tilde{\phi}_{sj} \hat{x}_{sj} \} \right) - \sum_{j=1}^n f_j \cdot X_i$$

is an upper bound for the optimal value of the objective function of CFL which is described by Eq. (1). To calculate the upper bound, the classical facility location problem can be solved as follows:

$$\max \sum_{j=1}^k \left(\sum_{i=1}^n \phi_{ij} x_{ij} + \sum_{i=1}^{n_l} \tilde{\phi}_{ij} \hat{x}_{ij} \right) - \sum_{i=1}^n f_i X_i, \tag{A.4}$$

s. t.

$$\sum_{i=1}^n f_i \cdot X_i \leq B_L, \tag{A.5}$$

$$x_{ik} \leq X_i \quad \forall i, k, \tag{A.6}$$

$$\sum_{s=1|}^{n_l} \hat{x}_{sk} + \sum_{s=1|}^n x_{sk} \leq 1 \quad \forall k, \tag{A.7}$$

$$\sum_{j=1}^k \hat{x}_{ij} \leq k \quad \forall i, \tag{A.8}$$

$$\sum_{j=1}^k x_{ij} \leq k \quad \forall i, \tag{A.9}$$

$$X_j \in \{0, 1\} \quad \forall j, \tag{A.10}$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j, \tag{A.11}$$

$$\hat{x}_{ij} \in \{0, 1\} \quad \forall i, j. \tag{A.12}$$

Biographies

Babak Yousefi Yegane received his BS in Industrial Engineering from the Engineering Faculty of Kurdistan University in 2003 and his MS in the same field from the Faculty of Industrial Engineering of Tafresh University (a branch of Industrial Engineering Faculty of Amir Kabir University of Technology) in 2010. He is currently, a PhD candidate of Industrial Engineering at the Industrial Engineering Department of University of Kurdistan. His research interests include supply chain management, scheduling, and game theory.

Isa Nakhai Kamalabadi is a Professor of Industrial Engineering at University of Kurdistan. He received his BSc degree in Computer Science and its applications from Shahid Beheshti University, Iran in 1979, the MSc degree in Industrial Engineering from Tarbiat Modares University, Iran in 1988, and PhD degree in Industrial Engineering from University of Toronto in 1995. He is currently working as Professor of Industrial Engineering at University of Kurdistan and as a Joint Professor at Tarbiat Modares University, Iran. His

research interests include optimization, supply chain management, scheduling, and information technology.

Hiwa Farughi is an Assistant Professor at the University of Kurdistan. He was born in 1976. He received BS and MS degrees in Industrial Engineering

from Amir Kabir University of Technology in 1998 and 2000, respectively. He received his PhD degree in Industrial Engineering from Iran University of Science and Technology in 2012. His research interest topics include operations research applications in health care, production planning, reliability, and quality control.