TECHNISCHE UNIVERSITÄT WIEN
Institut für Computergraphik und Algorithmen

Models and Algorithms for Competitive Facility Location Problems with Different Customer Behavior

Benjamin Biesinger, Bin Hu, and Günther Raidl

Forschungsbericht / Technical Report

TR–186–1–14–05

August 21, 2014

Favoritenstraße 9-11 / E186, A-1040 Wien, Austria
Tel. +43 (1) 58801-18601, Fax +43 (1) 58801-18699
www.cg.tuwien.ac.at

Submitted to **Annals of Mathematics and Artificial Intelligence**

Models and Algorithms for Competitive Facility Location Problems with Different Customer Behavior*

Benjamin Biesinger · Bin Hu · Günther Raidl

Abstract Competitive facility location problems arise in the context of two non-cooperating companies, a leader and a follower, competing for market share from a given set of customers. We assume that the firms place a given number of facilities on locations taken from a discrete set of possible points. For this bi-level optimization problem we consider six different customer behavior scenarios from the literature: binary, proportional and partially binary, each combined with essential and unessential demand. The decision making for the leader and the follower depends on these scenarios. In this work we present mixed integer linear programming models for the follower problem of each scenario and use them in combination with an evolutionary algorithm to optimize the location selection for the leader. A complete solution archive is used to detect already visited candidate solutions and convert them efficiently into similar, not yet considered ones. We present numerical results of our algorithm and compare them to so far state-of-the-art approaches from the literature. Our method shows good performance in all customer behavior scenarios and is able to outperform previous solution procedures on many occasions.

Keywords competitive facility location, evolutionary algorithm, solution archive, bi-level optimization

1 Introduction

In competitive facility location problems (CFLPs) two decision makers, a leader and a follower, compete for market share. They choose given numbers of facility locations from a finite set of possible positions in order to satisfy client demands, whereas the leader starts to place all of his facilities, then the follower places his facilities. In our work we consider different scenarios which vary in the way

* This work is supported by the Austrian Science Fund (FWF) under grant P24660-N23.

Institute of Computer Graphics and Algorithms
Vienna University of Technology
Favoritenstraße 9-11/1861, 1040 Vienna, Austria
E-mail: {biesinger|hu|raidl}@ads.tuwien.ac.at

customers satisfy their demands from the set of open facilities. This classification is taken from Suárez-Vega et al. [24]:

Customer behavior

- Binary: The demand of each customer is fulfilled by the nearest facility only.
- Proportional: Each customer splits his demand over all open facilities proportional to an attractiveness value, which depends on the distances to the facilities.
- Partially binary: This is similar to the proportional behavior but the demand is split only between the nearest leader and nearest follower facility, again, proportional to an attractiveness value depending on the distance.

Demand model

- Essential demand: The customers satisfy all of their demand.
- Unessential demand: The customers do not satisfy all of their demand but only a proportion depending on the distance to the serving facility.

Combining the three customer behaviors and the two demand models results in six different scenarios. Since demand corresponds to the buying power of the customers the turnover of the competing firms increases with the amount of fulfilled demand. Therefore, in order to obtain an accurate revenue value for the leader, the subproblem of finding an optimal set of facility locations for the follower for a given set of leader locations has to be solved. This makes the problem a Σ_2^P-hard bi-level optimization problem [18]. In this work we model the decision problem of the leader who wants to maximize her turnover knowing that a follower will enter the market subsequently under a given customer behavior scenario. We propose mathematical models as well as a hybrid metaheuristic based on an evolutionary algorithm to approximately solve all variants of this problem in a practically efficient way.

Our evolutionary algorithm (EA) searches for the best possible facility locations for the leader so that her turnover is maximized. It is assumed that the follower will place his facilities optimally, i.e., aiming at maximizing his revenue or minimizing the leader's revenue. For the problem of finding the optimal locations for the follower, mixed integer linear programming (MIP) models for different customer behaviors are presented. These models can then be solved either exactly using a general purpose MIP solver like CPLEX or approximated by solving their linear programming (LP) relaxation or by a greedy algorithm. As a result, we obtain a multi-level evaluation scheme which reduces the number of accurate, hence more time-consuming, evaluations which can be applied when the LP relaxation value of the model is good enough. The EA is further enhanced with a solution archive which is a special data structure that stores all generated candidate solutions and converts duplicate solutions into guaranteed not yet considered ones. A local search procedure, combined with the archive into a tabu search variant, further improves promising solutions of the EA and thus turns it into a powerful hybrid approach. This article extends our previous work [5] by covering all customer behavior scenarios introduced in [24] and providing models as well as numerical results.

In Section 2 we define the problem under the different customer behavior scenarios more formally. Related work is presented in Section 3, which is followed by a description of the mathematical models for our considered CFLPs in Section 4. Section 5 introduces our evolutionary algorithm and its extensions. Section 6 discusses our computational results and compares our method to approaches from the literature when possible. Finally, we draw conclusions in Section 7 and give an outlook on further promising research questions.

2 Problem Definition

In the following we will formally define the competitive facility location problem with different customer behavior scenarios. Given are the numbers $p \geq 1$ and $r \geq 1$ of facilities to be opened by the leader and follower, respectively, and a weighted complete bipartite graph $G = (I, J, E)$ where $I = \{1, \ldots, m\}$ represents the set of potential facility locations, $J = \{1, \ldots, n\}$ represents the set of customers, and $E = I \times J$, is the set of edges indicating corresponding assignments. Let $w_j > 0, \forall j \in J$, be the demand of each customer, which corresponds to the (maximal) turnover to be earned by the serving facilities, and $d_{ij} \geq 0, \forall (i, j) \in E$, be the distances between customers and potential facility locations. The goal for the leader is to choose exactly p locations from I for opening facilities in order to maximize her turnover under the assumption that the follower in turn chooses r locations for his facilities optimally maximizing his turnover.

The turnover distribution of the customers differ in the six scenarios defined before and in the folowing we will give a formal description of the turnover computation of all scenarios. The definitions for the binary essential case is taken from [6] and for the proportional case from [5]. In the following let (X, Y) be a candidate solution to our competitive facility location problem, where $X \subseteq I,\ |X| = p$, is the set of locations chosen by the leader and $Y \subseteq I,\ |Y| = r$, is the associated set of follower locations. Note that X and Y do not have to be disjunct in general. Further, let $D(j, V) = \min\{d_{ji} \mid i \in V\},\ \forall j \in J,\ V \subseteq I$ be the minimum distance from customer j to all facility locations in set V. Following Kochetov et al. [13] we define the attractiveness of a facility location to a customer by $v_{ij} = \frac{a_{ij}}{(d_{ij}+1)^{\beta}}$ and define analogous to the minimum distance the maximum attractiveness from customer j to all facility locations in the set V as $A(j, V) = \max\{v_{ji} \mid i \in V\},\ \forall j \in J,\ V \subseteq I$. In this work we set $\beta = 1$ and $a_{ij} = 1\ \forall i \in I,\ j \in J$. For the attractiveness one is added to the original distances d_{ij} just to avoid numerical problems with zero distances which might occur when considering the same locations for facilities and customers.

In the next sections we follow the classification of the different customer behavior scenarios [24] and give definitions of the turnover computation of each of these scenarios.

2.1 Binary Essential

Each customer $j \in J$ chooses the closest facility, hence the owner of this closest facility gains the complete turnover w_j. The leader is preferred in case of equal distances so the follower never places a facility at a location occupied by the leader and therefore we can assume that $X \cap Y = \emptyset$ for this scenario. The set of customers which are served by one of the follower's facilities is $U^{\mathrm{f}} = \{j \in J \mid$

$D(j,Y) < D(j,X)\}$ and the customers served by the leader is given by $U^{\mathrm{l}} = J \backslash U^{\mathrm{f}}$. Consequently, the total turnover of the follower is $p^{\mathrm{f}} = \sum_{j \in U^{\mathrm{f}}} w_j$ and the total turnover of the leader $p^{\mathrm{l}} = \sum_{j \in J} w_j - p^{\mathrm{f}}$. Note that this problem is also known as $(r|p)$-*centroid problem* [9].

2.2 Proportional Essential

Each customer j splits all of his demand over all opened facilities proportional to their attractiveness. Let $x_i = 1$ if $i \in X$ and $x_i = 0$ otherwise, and $y_i = 1$ if $i \in Y$ and $y_i = 0$ otherwise, $\forall i \in I$. Then, the turnover of the follower is

$$p^{\mathrm{f}} = \sum_{j \in J} w_j \frac{\sum\limits_{i \in I} v_{ij} y_i}{\sum\limits_{i \in I} v_{ij} x_i + \sum\limits_{i \in I} v_{ij} y_i}$$

and the turnover of the leader is

$$p^{\mathrm{l}} = \sum_{j \in J} w_j - p^{\mathrm{f}}.$$

2.3 Partially Binary Essential

Each customer j splits all of his demand over the nearest leader and the nearest follower facility proportional to their attractiveness. Let $v_j^L = A(j,X)$, i.e., the highest attraction value from any leader facility to customer j and $v_j^F = A(j,Y)$. Then, the turnover of the follower is

$$p^{\mathrm{f}} = \sum_{j \in J} w_j \frac{v_j^F}{v_j^F + v_j^L}$$

and the turnover of the leader is

$$p^{\mathrm{l}} = \sum_{j \in J} w_j - p^{\mathrm{f}}.$$

2.4 Unessential Demand

In the unessential demand scenarios we need a function which describes how much the demand of a customer decreases with the distance to the nearest facility. We define this demand reduction function as $f(d) = \frac{1}{(d+1)^\gamma}$. Parameter γ controls the decrease of demand, in our work we assume $\gamma = 1$. Further, we note that when the demand is unessential $\sum\limits_{j \in J} w_j \geq p^{\mathrm{l}} + p^{\mathrm{f}}$, i.e., the total demand satisfied by the leader and the follower is no longer necessarily equal to the total demand of all customers. In the following we present the profit computation for the unessential scenarios under the different customer choice rules:

– Binary Unessential

$$p^{\mathrm{f}} = \sum_{j \in U^{\mathrm{f}}} w_j f(D(j,Y)) \quad \text{and} \quad p^{\mathrm{l}} = \sum_{j \in U^{\mathrm{l}}} w_j f(D(j,X))$$

– Proportional Unessential

$$p^{\mathrm{f}} = \sum_{j \in J} w_j \frac{\sum_{i \in I} v_{ij} f(d_{ij}) y_i}{\sum_{i \in I} v_{ij} x_i + \sum_{i \in I} v_{ij} y_i} \quad \text{and}$$

$$p^{\mathrm{l}} = \sum_{j \in J} w_j \frac{\sum_{i \in I} v_{ij} f(d_{ij}) x_i}{\sum_{i \in I} v_{ij} x_i + \sum_{i \in I} v_{ij} y_i}$$

– Partially Binary Unessential

$$p^{\mathrm{f}} = \sum_{j \in J} w_j \frac{v_j^F}{v_j^F + v_j^L} f(D(j,Y)) \quad \text{and} \quad p^{\mathrm{l}} = \sum_{j \in J} w_j \frac{v_j^L}{v_j^F + v_j^L} f(D(j,X))$$

3 Related Work

Competitive facility location problems are an old type of problem introduced by Hotelling [11] in 1929. He considered two sellers placing one facility each on a line. In the last years many variations were considered that differ in the way the competitors can open their facilities and in the behavior of the customers. Kress and Pesch give an overview of competitive location problems in networks in [14]. Vega et al. [24] outline different customer choice rules of competitive facility location problems. They consider six different scenarios of customer behavior, including binary, partially binary, proportional as well as essential and unessential goods. In their work the authors assume that the facilities can be placed anywhere on the plane and give discretization results for several customer choice rules but no concrete solution algorithms. We use the classification of these scenarios for the models we used in this work.

Most of the articles that tackle competitive facility location problems focus on one customer behavior scenario. However, Hakimi [10] extended the basic formulation to different customer behaviors and also to unessential demand. Serra and Colome [23] developed metaheuristics for the follower problem where the leader is already in the market with binary, proportional and partially binary customer choice rules and essential demand.

The literature about the binary essential customer behavior scenario is the richest and this problem is widely known as the (discrete) $(r|p)$-centroid problem which has originally been introduced by Hakimi [9]. Alekseeva et al. [1–3] present several heuristic and exact solution approaches. Laporte and Benati [16] developed a tabu search and Roboredo and Pessoa [20] describe a branch-and-cut algorithm. In Section 6.1 we compare our approach to two metaheuristics by Alekseeva et al. [2,3].

Proportional essential customer behavior is considered by Kochetov et al. [13] who developed a matheuristic for a more general problem variant that contains our problem as special case. They propose a bi-level mixed integer non-linear programming model. To solve the problem more efficiently they linearized the follower's problem. Our approach also uses this linearized model. The authors suggest an alternating heuristic to solve the leader's problem which is derived from an alternating heuristic developed for the $(r|p)$-centroid problem with continuous facility locations in [4]. In Section 6.2 we compare our approach to their algorithm.

Literature about unessential demand and partially binary customer behavior is rare. Many papers about CFLPs mention these customer behavior scenarios [10, 24, 23] but do not provide concrete algorithms. One of our contributions in this work is to give linear models for the follower's problem under unessential demands and the partially binary choice rule to be able to tackle these scenarios and find effective solution procedures.

A related work by Fernández and Hendrix [7] also considers variants of CFLPs. They study recent insights in Huff-like competitive facility location and design problems. In their survey article they compared three different articles [15, 22, 21] describing all the same basic model. In all three papers, for each facilitiy a quality level has to be determined similar to the design scenarios used in Kochetov [13] and fixed costs for opening facilities incur. Küçükaydin et al. [15] and Saidani et al. [21] assume that the competitor is already in the market and Sáiz et al. [22] focus on finding a nash equilibrium of two competitors entering a new market and opening only one facility each. However, this variants are not scope of the current article.

4 Mathematical Models

In this section we present mathematical models for CFLPs with different customer behavior scenarios. In case of binary choice we adopt the linear model from Alekseeva [2]. Finding linear models for the partially binary and proportional case is not straightforward because we have to model a ratio of demand fulfilled by the leader and the follower, respectively. In these cases we present linear transformations which are based on the transformation performed by Kochetov [13] for the proportional essential scenario.

All models use two types of binary decision variables:

$$x_i = \begin{cases} 1 & \text{if the leader opens a facility at location } i \\ 0 & \text{else} \end{cases} \qquad \forall i \in I$$

and

$$y_i = \begin{cases} 1 & \text{if the follower opens a facility at location } i \\ 0 & \text{else} \end{cases} \qquad \forall i \in I.$$

4.1 Binary Essential

The following bi-level MIP model has been introduced in [2]. It uses an additional type of binary decision variables:

$$u_j = \begin{cases} 1 & \text{if customer } j \text{ is served by the leader} \\ 0 & \text{if customer } j \text{ is served by the follower} \end{cases} \quad \forall j \in J.$$

We define the set of facilities that allow the follower to capture customer j if the leader uses solution x ($x = (x_i)_{i \in I}$):

$$I_j(x) = \{i \in I \mid d_{ij} < \min_{l \in I | x_l = 1} d_{lj}\} \quad \forall j \in J$$

Then we can define the upper level problem, denoted as leader's problem, as follows:

$$\max \sum_{j \in J} w_j u_j^* \tag{1}$$

$$\text{s.t.} \sum_{i \in I} x_i = p \tag{2}$$

$$x_i \in \{0, 1\} \qquad \forall i \in I \tag{3}$$

where $(u_1^*, \ldots, u_m^*)$ is an optimal solution to the lower level problem, denoted as follower's problem:

$$\max \sum_{j \in J} w_j (1 - u_j) \tag{4}$$

$$\text{s.t.} \sum_{i \in I} y_i = r \tag{5}$$

$$1 - u_j \leq \sum_{i \in I_j(x)} y_i \qquad \forall j \in J \tag{6}$$

$$x_i + y_i \leq 1 \qquad \forall i \in I \tag{7}$$

$$u_j \geq 0 \qquad \forall j \in J \tag{8}$$

$$y_i \in \{0, 1\} \qquad \forall i \in I, \forall j \in J \tag{9}$$

The objective function for the leader's problem (1) maximizes the leader's turnover. Equation (2) ensures that the leader places exactly p facilities. The objective function for the follower's problem (4) maximizes the follower's turnover. Similarly as in the leader's problem, (5) ensures that the follower places exactly r facilities. Inequalities (6) together with the objective function ensure the u_j variables to be set correctly, i.e., decide for each customer $j \in J$ from which competitor he is served. Inequalities (7) guarantee that the follower does not choose a location where the leader has already opened a facility. Note that all x_i variables are considered as constants here. Variables u_j are not restricted to binary values because in an optimal solution they will become 0 or 1 anyway.

4.2 Proportional Essential

For the proportional essential scenario we start with a non-linear bi-level model which is derived from Kochetov et al. [13]. The upper level problem (leader's problem) is:

$$\max \sum_{j \in J} w_j \frac{\sum_{i \in I} v_{ij} x_i}{\sum_{i \in I} v_{ij} x_i + \sum_{i \in I} v_{ij} y_i^*} \tag{10}$$

$$\text{s.t.} \sum_{i \in I} x_i = p \tag{11}$$

$$x_i \in \{0,1\} \qquad \forall i \in I \tag{12}$$

where $(y_1^*, \ldots, y_m^*)$ is an optimal solution to the lower level problem (follower's problem):

$$\max \sum_{j \in J} w_j \frac{\sum_{i \in I} v_{ij} y_i}{\sum_{i \in I} v_{ij} x_i + \sum_{i \in I} v_{ij} y_i} \tag{13}$$

$$\text{s.t.} \sum_{i \in I} y_i = r \tag{14}$$

$$y_i \in \{0,1\} \qquad \forall i \in I \tag{15}$$

The objective functions (10) and (13) maximize the sums of the fulfilled demand by the leader and the follower, respectively, considering the splitting over the facilities inversely proportional to their distances. Constraint (11) ensures that the leader opens exactly p facilities and, similarly, constraint (14) guarantees that the follower places exactly r facilities. Note that the follower in principle is allowed to open facilities at the same locations as the leader. All of the x_i variables are considered as constants in the follower's problem.

In order to be able to solve the follower's problem more efficiently Kochetov et al. [13] suggested a linear transformation of this model, which works as follows. First, two new kinds of variables are introduced:

$$z_j = \frac{1}{\sum_{i \in I} v_{ij} x_i + \sum_{i \in I} v_{ij} y_i} \qquad \forall j \in J \tag{16}$$

and

$$y_{ij} = w_j z_j v_{ij} y_i \qquad \forall i \in I, j \in J. \tag{17}$$

Variables y_{ij} have the intuitive meaning that they are the demand of customer j that is supplied by the follower facility at location i and the z_j variables are basically the denominator of the fractional objective function for a fixed j. It is obvious that if we are able to model the non-linear equation (17) in a linear way such that equation (16) is valid we get a model that is equivalent to (13–15). This is realized by the following mixed integer linear program:

$$\max \sum_{j\in J}\sum_{i\in I} y_{ij} \tag{18}$$

$$\text{s.t.} \quad (14), (15) \text{ and}$$

$$\sum_{i\in I} y_{ij} + w_j z_j \sum_{i\in I} v_{ij} x_i \le w_j \qquad \forall j \in J \tag{19}$$

$$y_{ij} \le w_j y_i \qquad \forall i \in I, j \in J \tag{20}$$

$$y_{ij} \le w_j v_{ij} z_j \le y_{ij} + W(1 - y_i) \qquad \forall i \in I, j \in J \tag{21}$$

$$y_{ij} \ge 0, z_j \ge 0 \qquad \forall i \in I, j \in J \tag{22}$$

Objective function (18) maximizes the turnover obtained by the follower. Constraints (19) set the variables y_{ij} by restricting them to not exceed the total demand of customer j minus the demand captured by the leader. The fact that a facility location i can only get some turnover from customer j when the follower opens a facility there is ensured by constraints (20). Finally, equations (17) are fulfilled because of constraints (21).

Constant W is chosen large enough, so that an optimal solution to this model satisfies equations (16), i.e., $W = \max_{j\in J}(w_j) \cdot \max_{i\in I, j\in J}(v_{ij}) \cdot \max_{j\in J}(z_j)$, where $\max_{j\in J}(z_j) \le \max_{j\in J}(1/\sum_{i\in I} v_{ij} x_i)$ because of constraints (19). Due to constraints (21) with its W, the LP relaxation of this model unfortunately is in general relatively weak, therefore finding an optimal follower solution by this model using a general purpose mixed integer programming solver like CPLEX is time-consuming even for small instances. Nevertheless, this model is still easier to solve than the non-linear model (13–15).

4.3 Partially Binary Essential

The model for the partially binary essential scenario is similar to the model for the proportional case. The difference is that for each customer we only have to model the ratio of the nearest leader and the nearest follower facility, which results in the following non-linear bi-level model:

$$\max \sum_{j\in J} w_j \frac{v_j^L}{v_j^L + v_j^{F^*}} \tag{23}$$

$$\text{s.t.} \sum_{i\in I} x_i = p \tag{24}$$

$$v_j^L = \max_{i\in I}(v_{ij} x_i) \qquad \forall j \in J \tag{25}$$

$$x_i \in \{0, 1\} \qquad \forall i \in I \tag{26}$$

where $(v_1^{F^*}, \dots, v_m^{F^*})$ is an optimal solution to the lower level problem:

$$\max \sum_{j \in J} w_j \frac{v_j^F}{v_j^L + v_j^{F^*}} \tag{27}$$

$$\text{s.t.} \sum_{i \in I} y_i = p \tag{28}$$

$$v_j^F = \max_{i \in I} (v_{ij} y_i) \qquad \forall j \in J \tag{29}$$

$$y_i \in \{0, 1\} \qquad \forall i \in I \tag{30}$$

The objective functions (23) and (27) maximize the sums of the fulfilled demand by the leader and the follower, respectively, considering the splitting over their nearest facilities. Constraint (24) ensures that the leader opens exactly p facilities and, similarly, constraint (28) guarantees that the follower places exactly r facilities. The highest attraction values for each customer j, expressed by variables v_j^L and v_j^F, $\forall j \in J$ are set by the non-linear constraints (25) and (29).

Again, we propose a linear transformation of the follower model similar to the proportional case. We introduce three new kinds of variables:

$$z_j = \frac{1}{v_j^L + v_j^F} \qquad \forall j \in J \tag{31}$$

$$\hat{y}_{ij} = \begin{cases} 1 & \text{if } i \text{ is the nearest follower facility to customer } j \\ 0 & \text{else} \end{cases}$$

and

$$y_{ij} = w_j z_j v_{ij} \hat{y}_{ij} \qquad \forall i \in I, j \in J. \tag{32}$$

Once more, variables y_{ij} are set to the amount of demand a (possible) follower facility at location i supplies to customer j and z_j is the denominator of the objective function Note that exactly one facility satisfies a certain amount of demand of a customer and therefore for a fixed j exactly one y_{ij} variable has a non-zero value. The linearized model is presented next.

$$\max \sum_{j \in J} \sum_{i \in I} y_{ij} \tag{33}$$

$$\text{s.t.} \sum_{i \in I} y_i = p \tag{34}$$

$$\sum_{i \in I} y_{ij} + w_j z_j v_j^L \leq w_j \qquad \forall j \in J \tag{35}$$

$$y_{ij} \leq w_j \hat{y}_{ij} \qquad \forall i \in I, j \in J \tag{36}$$

$$y_{ij} \leq w_j v_{ij} z_j \leq y_{ij} + W(1 - \hat{y}_{ij}) \qquad \forall i \in I, j \in J \tag{37}$$

$$\hat{y}_{ij} \leq y_i \qquad \forall i \in I, j \in J \tag{38}$$

$$\sum_{i \in I} \hat{y}_{ij} = 1 \qquad \forall j \in J \tag{39}$$

$$y_i \geq 0, y_{ij} \geq 0, z_j \geq 0 \qquad \forall i \in I, j \in J \tag{40}$$

$$\hat{y}_{ij} \in \{0, 1\} \qquad \forall i \in I, j \in J \tag{41}$$

Objective function (33) maximizes the turnover obtained by the follower. Constraints (35) set the variables y_{ij} by restricting them to not exceed the total demand of customer j minus the demand captured by the leader. The fact that a facility location i can only get some turnover from customer j when there is the nearest open follower facility is ensured by constraints (36). Equations (32) are fulfilled because of constraints (37). Constraints (38) and (39) guarantee that there is exactly one nearest follower facility to each customer and that this location has to be chosen by the follower.

4.4 Unessential Cases

When the demand of customers is unessential, two different goals for the follower are possible. He can either aim to minimize the leader's profit (LMIN) or to maximize his profit (FMAX). Depending on the goal the follower might choose different locations for his facilities. In this section we will discuss the changes to the models introduced before that are needed to consider unessential demand.

4.5 Binary Unessential

In the LMIN scenario only a change in the objective function is needed because the distance from each customer to the nearest leader facility is known beforehand. The new objective function for the follower's problem is the following:

$$\min \sum_{j \in J} w_j z_j f(D(j, X))$$

If the follower uses the FMAX strategy new variables have to be introduced to indicate which location i hosts a follower facility that is nearer to a customer j than any other open (leader or follower) facility. This is modelled by decision variables $\hat{y}_{ij}$ which are defined similarly as before:

$$\hat{y}_{ij} = \begin{cases} 1 & \text{if } i \text{ is the nearest follower facility to customer } j \\ & \text{and nearer than all leader facilities} \\ 0 & \text{else} \end{cases}$$

The complete model for the follower problem is as follows:

$$\max \sum_{j\in J} w_j \sum_{i\in I} \hat{y}_{ij} f(d_{ij}) \tag{42}$$

$$\text{s.t.} \sum_{i\in I} y_i = r \tag{43}$$

$$1 - z_j \leq \sum_{i\in I_j(x)} y_i \qquad \forall j \in J \tag{44}$$

$$x_i + y_i \leq 1 \qquad \forall i \in I \tag{45}$$

$$\hat{y}_{ij} \leq y_i \qquad \forall i \in I, \forall j \in J \tag{46}$$

$$\hat{y}_{ij} \leq 1 - z_j \qquad \forall i \in I, \forall j \in J \tag{47}$$

$$\sum_{i\in I} \hat{y}_{ij} \leq 1 \qquad \forall j \in J \tag{48}$$

$$z_j \geq 0 \qquad \forall j \in J \tag{49}$$

$$y_i \in \{0,1\} \qquad \forall i \in I, \forall j \in J \tag{50}$$

$$\hat{y}_{ij} \in \{0,1\} \qquad \forall i \in I, \forall j \in J \tag{51}$$

In this model there are three new types of constraints to set the $\hat{y}_{ij}$ variables correctly. Constraints (46) ensure that if one of these variables is set to one then there must be a follower facility on this location. Furthermore, a $\hat{y}_{ij}$ variable is only set to 1 iff customer j is served by the follower, which is ensured by constraints (47). Of course, only one follower facility can be the nearest to a customer, so constraints (48) are introduced. The change in the objective function models the unessential demand by reducing the turnover gained by each customer by our demand reduction function f.

4.6 Proportional Unessential

In the proportional customer behavior scenario for both LMIN and FMAX a change in the objective function is needed and for LMIN additionally a change of constraints (19):

$$\text{LMIN:} \quad \min \sum_{j\in J} w_j z_j \sum_{i\in I} v_{ij} x_i f(d_{ij})$$

$$\sum_{i\in I} y_{ij} + w_j z_j \sum_{i\in I} v_{ij} x_i = w_j \qquad \forall j \in J$$

$$\text{FMAX:} \quad \max \sum_{j\in J} \sum_{i\in I} y_{ij} f(d_{ij})$$

4.7 Partially Binary Unessential

Also for the partially binary case, the objective function changes and for LMIN the constraints (35) as well:

$$\begin{aligned}
\text{LMIN:}\quad & \min \sum_{j\in J} w_j z_j v_j^L f\left(\frac{1}{v_j^L} - 1\right) & \\
& \sum_{i\in I} y_{ij} + w_j z_j v_j^L = w_j & \forall j \in J \\
\text{FMAX:}\quad & \max \sum_{j\in J}\sum_{i\in I} y_{ij} f(d_{ij}) &
\end{aligned}$$

5 Evolutionary Algorithm

In this section we present an EA that aims to find the optimal solution to the leader's problem for each different customer behavior scenario. The algorithmic framework is also used in [6] for binary customer behavior and in [5] for proportional customer behavior, both only considered essential demand. We use an incomplete solution representation only storing the facilities of the leader indicated by the binary vector $x = (x_1, \dots, x_m)$. For augmenting the incomplete leader solution, which can also be seen as evaluating a candidate leader solution, the follower's problem has to be solved. For this purpose we derived the MIP models in the last section. As solving these MIPs exactly is time-consuming in general, a greedy evaluation procedure and, for the binary customer behavior, the LP relaxation of the MIP model is used for approximating the quality of intermediate leader solution candidates, which is different for each case. Only at the end of the EA the best approximate solution found (for the proportional and partially binary case) or each candidate solution of the final population (for the binary case) is exactly evaluated using the corresponding MIP to get optimal objective values.

First, we explain the greedy solution evaluations for the different customer behavior scenarios. Then we will show a method for how to avoid exact evaluations during the EA and still do not miss potentially good solution candidates due to the approximation of objective values. We introduced this concept in [6] and called it multi-level solution evaluation scheme (ML-ES). It is applicable to models which have a reasonably good LP relaxation bound; here we apply ML-ES only for binary customer behavior. After explaining these solution evaluation methods we will introduce the EA with its variation operators. We further want to reduce the time needed for the solution evaluation so we employ a complete solution archive, which is a data structure that stores all generated candidate solutions. It efficiently converts created duplicates into similar but not yet considered solutions to avoid unnecessary evaluations. At the end of this chapter we show how we combine the solution archive and local search to a tabu search and integrate it into the EA to obtain a powerful hybrid approach.

5.1 Greedy Solution Evaluation

The greedy evaluation procedures are structurally similiar for each customer behavior and they try to find a near-optimal solution to the follower's problem in short time. They perform by iteratively selecting a locally best possible position for opening a facility, until all r follower facilities are placed. A currently best possible location is determined by computing the turnover of the follower for each possible additional location depending on the specific consumer behavior using the corresponding functions defined in Section 2:

$$\text{Binary Essential:} \quad p^{\text{f}}_{\text{BE}}(y) = \sum_{j \in U^{\text{f}}(y)} w_j \tag{52}$$

$$\text{Proportional Essential:} \quad p^{\text{f}}_{\text{PE}}(y) = \sum_{j \in J} w_j \frac{\sum\limits_{i \in I} v_{ij} y_i}{\sum\limits_{i \in I} v_{ij} x_i + \sum\limits_{i \in I} v_{ij} y_i} \tag{53}$$

$$\text{Partially Binary Essential:} \quad p^{\text{f}}_{\text{PBE}}(y) = \sum_{j \in J} w_j \frac{v^F_j(y)}{v^F_j(y) + v^L_j} \tag{54}$$

Here, $y = (y_1, \ldots, y_m)$ is the partial solution vector of the follower containing all so far opened facilities plus the candidate location. A location with the highest turnover is chosen; ties are broken randomly. The final value obtained from this procedure is a lower bound to the follower's problem and therefore $\sum_{j \in J} w_j - p^{\text{f}}(y)$ is an upper bound to the objective value of the leader's solution. For the binary essential case we do not have to recompute the whole function each time we place a new facility. Whenever a new facility captures facilities from the leader, they are removed from the set of customers and therefore do no longer increase the turnover of the follower. Then we only compute the turnover gain for each placed facility separately and in the end take the sum. When the demand is unessential the greedy criteria can be adapted analogously. However, the upper bound to the leader's problem has to be computed using the functions for the turnover computation for the leader defined in Section 2.

5.2 Multi-Level Evaluation Scheme

As mentioned in the beginning of this section we can devise a multi-level evaluation scheme originally introduced in [6] which exploits relationships of different solution evaluation methods to reduce the time needed for evaluation without missing potentially new incumbent solutions. As the greedy solution evaluation procedure is an approximation of the follower's problem and returns a feasible solution candidate, the greedy evaluation yields an upper bound to the leader's problem for a fixed x. When solving the LP relaxation of a corresponding follower's model from Section 4, i.e., solve the model by omitting the integrality constraints, we get a lower bound to the leader's problem (again, for a fixed x). These relations can be exploited as follows. Whenever a new solution candidate is generated its solution value is approximated by the greedy method. When the resulting turnover of the leader is worse than than or equal to the best turnover value of the leader obtained by solving the LP relaxation so far then we do not have to evaluate this solution candidate more accurately because we know that it cannot be better than our current best solution. However, when the resulting turnover of the leader is greater than the so far best LP value, we have to evaluate it more accurately (e.g., by solving the LP relaxation), which is more time consuming, and update the best solution found so far if needed.

Our algorithm benefits from the ML-ES when we are able to omit accurate evaluations often, which is the case when the LP relaxation value of the follower's problem is good enough. Unfortunately, preliminary tests showed that for the partially binary and proportional cases the LP relaxation is in general too weak and therefore we do not use ML-ES for these customer behavior scenarios. However, for the binary case, in over 95% of the created solution candidates we are able to avoid the more time-consuming evaluation, which results in a faster algorithm and a significantly better final solution quality.

5.3 Evolutionary Algorithm Framework

The EA's initial population is created by choosing p different facility locations uniformly at random to ensure a high diversity at the beginning. We employ a steady-state genetic algorithm in which exactly one new candidate solution is derived in each iteration. It always replaces the worst individual of the population. Binary tournament selection with replacement is used to choose two candidate solutions for recombination. Offsprings further undergo mutation.

Recombination works as follows. Suppose that two parent solutions $X^1 \subset I$ and $X^2 \subset I$ are selected. Then an offspring X' of X^1 and X^2 is derived by adopting all locations from $S = X^1 \cap X^2$ and adding $p - |X^1 \cap X^2|$ further locations from $(X^1 \cup X^2) \setminus S$ chosen uniformly at random.

Mutation is based on the swap neighborhood structure, which is also known from the p-Median problem [17]. A swap move closes a facility and re-opens it at a different, so far unoccupied position. Our mutation applies μ random swap moves, where μ is determined anew at each EA-iteration by a random sample from a Poisson distribution with mean value one.

Each new candidate solution derived via recombination and mutation whose objective value lies within a certain distance from the so far best solution value further undergoes a local improvement procedure or a tabu search. It is based on a local search applying the swap neighborhood structure already used for mutation. The best improvement step function is used, so all neighbors of a solution that are reachable via one swap move are considered and evaluated and the best one is selected for the next iteration. This procedure terminates when no superior neighbor can be found, i.e., a locally optimal solution has been identified.

5.4 Solution Archive

We use a *solution archive* that stores all generated candidate solutions in a compact data structure. The archive is attached to the EA framework either after mutation is performed or in conjunction with local search. The essential idea is to avoid the reconsideration of already evaluated solutions by converting them into similar, but new solutions, i.e., performing an "intelligent mutation". This concept is able to boost the performance of evolutionary algorithms with complex solution evaluations significantly, and further reduces the danger of premature convergence. It has been successfully applied to benchmark problems with binary solution representations, including NK landscapes and Royal Road functions [19], the generalized minimum spanning tree problem [12], and our previous work on the current problem [5]. Another rather theoretical property of such an archive-enhanced EA is that in principle it is a complete optimization approach yielding a guaranteed optimal solution in bounded time after considering all solutions of the search space. In practice, however, such an EA usually will be terminated earlier, still yielding only a heuristic solution.

We combine the EA and the solution archive as follows: Each time a candidate solution is created, we check if this solution is already contained in the archive. In case it is a duplicate it is converted on-the-fly into a not yet considered solution. Then this new solution is inserted into the archive and transferred back to the EA, where it is integrated into the population. For the underlying data structure we use a trie, which is a tree data structure often applied in dictionary applications [8] and allows inserting, searching and converting a solution to be implemented in $\mathcal{O}(m)$ time, where m is the length of the solution representation, i.e., independent of the number of solutions it contains. Such a trie has a strong relationship to an explicitly stored branch-and-bound tree, as each node divides the search space into two subspaces: Each trie node at level i corresponds to the i-th bit in the solution vector and has two entries representing "0" and "1". They either contain a reference to a trie node on the next level, a complete-flag which indicates that all solutions in the subtree have been visited, or an empty-flag which indicates that none of the solutions in the subtree has been visited in the EA.

To insert a solution into the trie, we follow the solution vector and go down the trie. If we encounter a complete-flag, we know that the solution has been inserted before and thus is a duplicate. Otherwise when we reach the last level, we insert a complete-flag in the corresponding entry. To convert a duplicate solution we strive to flip a minimal number of bits in the binary vector. The decision of which bits to be flipped is based on following the solution in the trie and take alternative randomly selected branches that lead to unexplored subspaces. For a detailed description of the insertion and conversion operators, we refer to our previous work [5].

5.5 Archive-based Tabu Search

In cooperation with the solution archive the basic local improvement procedure can be extended to a tabu search variant where the solution archive acts as tabu list. When enumerating the swap neighborhood of a candidate solution, we check for each neighbor solution if it has already been visited before, i.e., is contained in the solution archive. Only so far unvisited solutions are evaluated and the best one is selected for the next iteration, even if it is worse than the original solution; ties are broken randomly. This process is repeated for α iterations without improving the objective value or until there is no more unvisited neighbor solution. Note that our approach differs from classical tabu search implementations since we do not consider move attributes to be black-listed in a tabu list of limited length, but use the solution archive instead. This tabu search is applied to the most promising solutions in the EA population.

6 Computational Results

In this section we present computational results of our algorithmic approach applied to different customer behavior scenarios and demand models. We consider separate sets of instances for the binary and for the proportional and partially binary case because the binary essential case has a significantly lower complexity and we want to maintain comparability to algorithms from the literature. We used instances generated in [6] for the binary case and instances generated as part of our previous work [5] for the proportional and partially binary case. Both instance sets are based on instances from the discrete problem library[1] and can be found online[2]. In all instances each customer location corresponds to a possible facility location, i.e., $I = J$ and the other properties are the following:

	Binary essential and unessential	**Proportional/ partially binary essential and unessential**
Number of locations	essential: 200, unessential: 100, chosen randomly on an Euclidean plane of size 7000×7000	100, chosen randomly on an Euclidean plane of size 100×100
Customer demands	chosen uniformly at random from the set $\{1, \ldots, 200\}$	chosen uniformly at random from the set $\{1, \ldots, 10\}$
r, p	$r = p \in \{10, 15, 20\}$	$r \in \{2, \ldots, 5\}, p \in \{2, \ldots, 10\}$

The parameter settings for the EA were determined in preliminary tests and are similar for all scenarios. The population size is 100 and the EA is terminated after 3000 iterations without improvement or after 300 seconds except for the binary case, where we have a fixed time limit of 600 seconds. The termination parameter α for the tabu search-based local search is set to five. Local search/tabu search is called for each candidate solution whose objective value lies within 1% (for the binary case 5%) of the best solution found so far. After the EA finishes, the final best solution is evaluated exactly by solving the corresponding MIP from Section 4 and using the best greedy solution as starting solution with CPLEX 12.5. In preliminary tests it turned out that for the binary behavior the exact evaluation of one candidate solution needs less than one second, so in these test cases we evaluated the whole population after the last iteration exactly and took the best solution candidate among them as our final solution. All tests were performed on a single core of an Intel Xeon Quadcore with 2.54 GHz. In the next sections each customer behavior scenario with essential demand is analyzed and discussed.

At the end of each of the following tables for essential demands we give a quick overview over all instances on the geometric mean, the number of instances where the corresponding configuration performed best and the number of instances where the algorithm performed best and better than all others.

[1] http://www.math.nsc.ru/AP/benchmarks/english.html

[2] www.ads.tuwien.ac.at/w/Research/Problem_Instances#Competitive_Facility_Location_Problems

6.1 Binary Essential

First we evaluate in Table 1 how our algorithm performs compared to algorithms from the literature when the behavior of the customers is binary and the demand is essential. Our EA uses the solution archive (SA) and the multi-level solution evaluation scheme (ML-ES) as described in Section 5.2 and we compare it to the tabu search approach (TS_{AL}) by Alekseeva et al. [3] and the hybrid memetic algorithm which embeds this tabu search (HMA) developed in [2]. All algorithms are executed 30 times with a total run time of 600 seconds. The average objective values over these runs and their associated standard deviations are given in columns $\overline{obj}$ and sd, respectively. For our EA, the median of the time needed for the best solution is given as well in the column $t_{\text{best}}[s]$.

Table 1 Results of binary customer behavior with essential demand.

			TS_{AL}		HMA		EA + SA + ML-ES		
Instance	r	p	$\overline{obj}$	sd	$\overline{obj}$	sd	$\overline{obj}$	sd	$t_{\text{best}}[s]$
Code1	10	10	9545,43	35,14	9505,07	57,16	**9594,00**	10,37	243,10
Code2	10	10	**9324,50**	50,20	9217,80	58,07	9321,13	26,28	130,10
Code3	10	10	9367,07	32,45	9329,37	53,93	**9374,30**	28,15	227,00
Code4	10	10	8882,03	18,31	8877,13	22,02	**8888,47**	14,39	115,60
Code5	10	10	9227,30	48,62	9240,40	52,15	**9273,10**	27,45	268,20
Code6	10	10	9825,20	35,02	9808,13	39,34	**9850,53**	5,58	197,50
Code7	10	10	9225,70	42,60	9183,77	55,95	**9270,30**	20,44	222,80
Code8	10	10	9088,17	9,62	9046,43	34,70	**9092,57**	2,37	170,60
Code9	10	10	9009,53	3,68	8950,47	59,78	**9011,40**	8,76	182,90
Code10	10	10	9382,67	25,28	9365,40	46,44	**9411,00**	0,00	151,70
Code1	15	15	10076,73	49,31	10051,83	59,42	**10095,00**	37,02	297,10
Code2	15	15	9578,77	46,03	9514,93	51,54	**9626,67**	17,34	392,00
Code3	15	15	9355,93	18,85	9310,30	44,48	**9365,97**	17,19	281,90
Code4	15	15	9169,93	18,46	9116,27	68,57	**9179,03**	32,68	241,30
Code5	15	15	9242,57	64,44	9237,70	41,65	**9252,03**	42,10	320,90
Code6	15	15	10119,03	52,39	10095,73	41,17	**10148,23**	27,71	326,70
Code7	15	15	9556,13	39,65	9496,63	59,54	**9580,30**	35,03	283,90
Code8	15	15	9047,13	47,40	8987,20	41,46	**9063,10**	41,76	357,90
Code9	15	15	9124,70	66,93	9086,47	65,56	**9168,20**	23,40	335,40
Code10	15	15	9290,80	49,24	9240,83	57,79	**9312,40**	51,91	434,30
Code1	20	20	**9837,17**	53,95	9767,93	58,96	9831,97	56,35	460,50
Code2	20	20	**9667,17**	32,12	9602,20	38,63	9666,37	52,72	421,30
Code3	20	20	9286,17	67,10	9253,50	63,57	**9296,67**	70,96	426,90
Code4	20	20	**9439,13**	34,47	9402,23	55,74	9404,70	89,41	388,50
Code5	20	20	9498,80	38,81	9422,63	52,81	**9512,10**	42,91	345,90
Code6	20	20	**10283,10**	83,09	10210,53	59,37	10261,53	91,67	452,50
Code7	20	20	9902,20	43,20	9860,03	52,13	**9943,10**	33,88	361,90
Code8	20	20	9329,67	29,32	9248,07	59,96	**9342,90**	23,35	484,30
Code9	20	20	9438,00	17,91	9404,67	42,67	**9452,57**	16,55	416,80
Code10	20	20	**9741,20**	35,77	9683,63	50,92	9688,73	74,95	460,40
geometric mean			9456,10		9411,33		**9470,05**		
#best results			6		0		**24**		
#unique best res.			6		0		**24**		

In Table 1 we can clearly see the superiority of our algorithm as we are able to outperform TS_{AL} on 24 instances. We also tested for statistical significance in our previous work [6] with a larger instance set and there we showed that our algorithm is statistically better in 38 out of 90 instances, worse in 3 instances and equal in 17 instances. We could not observe statistically significant differences on the remaining 32 instances. We refer the reader to [6] for a more detailed analysis of our computational results for the binary essential case.

6.2 Proportional Essential

For proportional customer behavior we evaluate the impact of the solution archive on the results in Table 2 as well as the performance compared to the alternating heuristic (AH) by Kochetov et al. [13]. Their AH is based on a starting solution for the leader to find the optimal facility locations for the follower which are computed using the linear MIP model for the follower. This follower solution is subsequently chosen as leader solution and the optimal follower solution is found again. This procedure is repeated until a solution is obtained which has already been generated. Since the repeated exact computation of the optimal follower's locations is very time-consuming we modified their approach by using our greedy algorithm instead of the MIP as described in Section 5.1 for finding the locations for the follower. The results are based on our previous work [5] where we analyzed following configurations:

- The EA variant where the final best solution is not evaluated with the MIP. This means that the corresponding objective values are not exact, but only approximate values from the greedy evaluation method.
- A modified version of the Alternating Heuristic (MAH) by Kochetov et al. [13], where each solution candidate is approximated by our greedy algorithm instead of evaluated exactly.
- The EA variant (EA + MIP) that does not employ the archive and utilizes the basic local search only; the final best solution is evaluated with MIP.
- The EA variant (EA + SA + MIP) that uses the solution archive and the tabu search as local improvement method; the final best solution is evaluated with MIP.

In this table, again, $\overline{obj}$ stands for the average of the objective values over 30 runs with their standard deviation in column *sd*. The time needed until termination is given in column $t[s]$. Since MAH is a deterministic algorithm only one run is performed.

In Table 2 the numerical values are given. Numbers in parenthesis mean that evaluating the best solution candidate of the EA needed more than 3600 seconds and so the objective values are determined by the greedy algorithm only. Therefore they are only approximations and not directly comparable to exact objective values. So in the summary of the EA configuration these values are not considered for comparison. The best value in each row is marked bold. When some values of a row are obtained by greedy evaluation and some other values in the same row are exact solution qualities, only the exact values are compared to each other, e.g., in the row with $r = p = 3$.

In some cases of the EA + MIP variant not all 30 runs terminated within the time limit so only the average over the finished runs is given, e.g., the row with $r = 3$ and $p = 5$. We observe that even for small p values of 4 and 5 we were not able to evaluate even one solution candidate in the given time limit. Another interesting point is that evaluating the candidate solution exactly via the MIP is the most time-consuming part of the algorithm; for $r = 3$ and $p = 8$ it needed

Table 2 Results of proportional customer behavior with essential demand.

		EA			EA + MIP			MAH		EA + SA + MIP		
r	p	$\overline{obj}$	sd	$t[s]$	$\overline{obj}$	sd	$t[s]$	obj	$t[s]$	$\overline{obj}$	sd	$t[s]$
2	2	(280,002)	0,13	106	278,671	0,15	677	277,942	667	**278,736**	0,00	600
2	3	(338,170)	0,31	157	336,587	0,19	684	334,233	535	**337,228**	0,00	625
2	4	(374,754)	0,71	154	373,455	0,48	623	373,665	503	**374,425**	0,00	674
2	5	(399,834)	0,50	200	398,642	0,90	493	399,208	260	**401,781**	0,00	505
2	6	(419,360)	0,46	241	418,779	0,67	525	419,920	275	**421,091**	0,15	586
2	7	(434,640)	0,62	223	434,388	0,74	394	431,803	272	**436,123**	0,00	440
2	8	(447,131)	0,42	202	446,710	0,46	322	446,474	158	**448,192**	0,18	440
2	9	(456,992)	0,45	300	456,615	0,63	419	455,788	166	**458,905**	0,37	529
2	10	(465,178)	0,54	300	464,620	0,53	401	463,211	173	**467,055**	0,16	416
3	2	(223,059)	0,18	144	(223,059)	0,18	144	(223,153)	<1	**(223,194)**	0,00	27
3	3	(281,283)	0,31	174	(281,283)	0,31	174	276,818	5959	**279,000**	0,00	6397
3	4	(321,185)	0,86	201	(321,185)	0,86	201	319,427	4128	**319,819**	0,00	3956
3	5	(349,644)	0,54	300	347,429*	0,48	3892	349,471	3867	**349,793**	0,00	2703
3	6	(372,924)	0,68	300	371,900*	0,98	3896	372,760	3453	**373,836**	0,12	2777
3	7	(391,264)	0,65	300	390,753*	0,74	3493	391,314	2086	**391,894**	0,39	2658
3	8	(406,302)	0,52	300	405,907*	0,77	3124	407,623	1721	**407,765**	0,08	3148
3	9	(418,553)	0,37	300	418,051*	0,53	2795	419,985	1709	**420,305**	0,18	2424
3	10	(429,040)	0,56	300	428,357	0,53	2370	430,465	1299	**431,578**	0,33	2670
4	2	(183,188)	0,11	246	(183,188)	0,11	246	**(183,223)**	<1	**(183,223)**	0,00	38
4	3	(238,953)	0,33	226	(238,953)	0,33	226	(239,527)	<1	**(239,628)**	0,00	83
4	4	(279,021)	0,56	298	(279,021)	0,56	298	(280,336)	<1	**(280,549)**	0,08	126
4	5	(310,562)	0,70	300	(310,562)	0,70	300	**(313,041)**	<1	**(313,041)**	0,00	157
4	6	(335,415)	0,65	300	(335,415)	0,65	300	(337,158)	<1	**(337,540)**	0,12	242
4	7	(355,659)	0,52	300	(355,659)	0,52	300	(356,575)	<1	**(358,233)**	0,18	267
4	8	(372,334)	0,67	300	(372,334)	0,67	300	(374,436)	<1	**(375,031)**	0,04	300
4	9	(386,207)	0,81	300	(386,207)	0,81	300	(387,975)	<1	**(389,837)**	0,12	300
4	10	(398,011)	0,74	300	(398,011)	0,74	300	(400,421)	1	**(401,428)**	0,13	300
5	2	(156,357)	0,15	199	(156,357)	0,15	199	**(156,538)**	<1	**(156,538)**	0,00	44
5	3	(207,548)	0,18	293	(207,548)	0,18	293	(207,682)	<1	**(208,025)**	0,00	112
5	4	(247,295)	0,76	300	(247,295)	0,76	300	(244,959)	<1	**(248,663)**	0,06	212
5	5	(278,806)	0,60	300	(278,806)	0,60	300	(279,889)	<1	**(281,522)**	0,00	194
5	6	(304,283)	0,54	300	(304,283)	0,54	300	(305,488)	<1	**(307,129)**	0,13	300
5	7	(325,520)	0,81	300	(325,520)	0,81	300	(327,357)	<1	**(328,314)**	0,05	300
5	8	(343,534)	0,61	300	(343,534)	0,61	300	(345,947)	<1	**(346,254)**	0,12	300
5	9	(358,373)	1,02	300	(358,373)	1,02	300	(360,572)	<1	**(362,159)**	0,31	300
5	10	(371,213)	0,74	300	(371,213)	0,74	300	**(374,737)**	1	(374,556)	0,24	300
geo. mean		(330,38)			330,06			330,55		**331,67**		
#best res.		-			0			1		**35**		
#u. best res.		-			0			1		**32**		

*Not all of the 30 runs completed in the time limit

over 90% of the overall time but it decreases when p increases. The run-time of all configurations that incorporate the exact evaluation increases steadily with r because of the growing complexity of the MIP.

On some instances MAH finds a solution in less time than our algorithms and especially when the exact evaluation is too time consuming it is very fast. The quality of the solutions is similar to our EA approach when we do not use the SA, but by incorporating the solution archive we boosted the performance of our algorithm so that the final solution quality is in all but 4 of the tested instances better than the quality of the solutions produced by MAH and in 3 of the 4 cases it is equal. For some of the smaller instances EA + SA + MIP has a very small standard deviation, which underlines the robustness of our algorithm.

Compared to binary customer behavior, the proportional scenario is much harder to solve and we can only approximate the value of solution candidates for instances that are only half the size.

6.3 Partially Binary Essential

In the next computational tests we analyzed the partially binary essential customer behavior. Since there is, to the best of our knowledge, no algorithm with numerical results described in the literature we only compare different configurations of our EA. Similarly to the proportional case we compare our EA without exact evaluation, the EA with exact evaluation in the end (EA + MIP) and the EA with solution archive and exact evaluation (EA + SA + MIP). Table 3 shows our numerical results, column names have the same meaning as before.

Table 3 Results of partially binary customer behavior with essential demand.

		EA			EA + MIP			EA + SA + MIP		
r	p	$\overline{obj}$	sd	$t[s]$	$\overline{obj}$	sd	$t[s]$	$\overline{obj}$	sd	$t[s]$
2	2	(283,753)	0,29	26	278,450	1,15	549	**278,931**	0,00	529
2	3	(315,105)	0,47	33	309,243	0,49	539	**310,013**	0,00	515
2	4	(337,476)	1,13	32	330,013	1,24	432	**332,359**	0,00	376
2	5	(349,361)	0,35	38	343,743	0,46	417	**345,116**	0,32	444
2	6	(359,113)	0,53	36	354,270	0,76	436	**357,640**	0,45	437
2	7	(368,140)	0,64	40	363,248	0,70	404	**366,883**	2,55	412
2	8	(376,035)	0,71	33	370,994	1,18	378	**376,136**	2,12	382
2	9	(383,784)	0,84	47	378,761	1,91	382	**385,354**	1,08	316
2	10	(391,553)	1,31	44	384,370	1,72	378	**388,068**	0,47	303
3	2	(259,461)	0,25	38	247,791	0,34	590	**247,946**	0,00	606
3	3	(289,450)	0,74	42	277,505	1,26	451	**279,000**	0,00	432
3	4	(311,032)	1,50	43	299,228	1,47	380	**302,217**	0,00	354
3	5	(323,333)	0,93	39	312,901	1,92	362	**313,582**	0,43	362
3	6	(334,559)	0,65	42	324,425	0,88	393	**325,250**	0,97	386
3	7	(343,815)	0,66	49	333,255	1,39	354	**335,827**	1,37	348
3	8	(352,919)	0,83	57	341,766	0,91	331	**347,421**	2,07	344
3	9	(360,388)	1,14	72	349,304	1,94	333	**356,983**	2,23	320
3	10	(367,969)	1,45	64	355,705	1,97	305	**363,047**	1,97	348
4	2	(239,204)	0,54	58	225,354	0,57	559	**225,640**	0,00	560
4	3	(269,482)	0,64	52	253,806	1,15	429	**255,072**	0,00	410
4	4	(290,283)	1,68	45	274,913	1,94	331	**279,000**	0,00	330
4	5	(303,248)	1,61	51	288,922	1,95	349	**291,000**	0,62	330
4	6	(315,374)	0,65	56	301,074	0,58	331	**303,139**	0,78	343
4	7	(324,823)	0,61	83	310,542	0,83	325	**315,167**	0,33	298
4	8	(333,640)	0,86	78	319,463	1,15	317	**327,670**	0,00	302
4	9	(341,007)	0,87	80	327,074	3,05	299	**335,919**	0,13	318
4	10	(348,310)	1,02	97	335,461	1,64	299	**343,982**	0,58	304
5	2	(220,928)	0,72	58	211,955	0,94	667	**212,604**	0,00	626
5	3	(250,491)	0,99	52	240,746	1,07	515	**242,035**	0,00	427
5	4	(272,251)	2,35	45	262,368	2,43	379	**265,917**	0,00	365
5	5	(285,997)	1,32	51	276,552	1,71	403	**278,193**	0,00	401
5	6	(297,032)	0,58	56	287,690	0,77	402	**290,754**	0,88	400
5	7	(306,395)	0,75	83	297,093	0,92	356	**301,843**	0,49	340
5	8	(315,239)	0,74	78	306,201	1,21	370	**314,168**	0,00	340
5	9	(323,263)	0,98	80	314,983	1,40	353	**323,154**	0,00	364
5	10	(330,717)	2,00	99	322,066	2,12	315	**330,684**	0,00	311
geo. mean		(316,00)			305,65			**309,24**		
#best res.		-			0			**36**		
# u. best res.		-			0			**36**		

First, we observe that for all our tested instance we were able to evaluate the best solution candidate exactly, even for the cases which were not possible for the proportional customer behavior. Also, the time needed for this evaluation is much less and at most about 10 minutes for the hardest instance (in the case of $r = 5$ and $p = 2$). The deviation of the greedy objective value and the exact objective value is around 3% on average which shows that our greedy solution evaluation method is relatively accurate. In this customer behavior scenario the benefits of using a solution archive are even more obvious than in the other scenarios as EA + SA + MIP performed better in all our tested instances. Second, we see again that for a fixed r the time needed for solving the model decreases with increasing p because the solution space is getting smaller. For many of the instances we obtained a very low standard deviation which, again, shows the robustness of our approach.

Compared to the other customer behavior scenarios the complexity of partially binary behavior lies in between the binary and the proportional choice rule, where binary is the easiest to solve and proportional by far the hardest. We also see that the leader is preferred in proportional scenarios as for a fixed r and p the turnover is higher than in the partially binary case in most of the instances but especially for a large p and small r, i.e., when he is able to place more facilities than the follower. For example, the turnover for the leader when $r = 3$ and $p = 10$ is in the proportional case nearly 16% higher than when the customers use the partially binary choice.

6.4 Unessential Demands

We performed computational tests for all customer behavior scenarios with unessential demands. Like in the partially binary customer behavior also for unessential demands there are no numerical results available in the literature. We tested the two different follower strategies LMIN and FMAX and compared them to each other. In the following tables in addition to the average leader objective value (obj^{l}) over 30 runs we also present the average turnover obtained by the follower for the corresponding best leader solution found (obj^{f}). For both values the standard deviations are given as well (sd). Usually only a fraction of the total demand of all customers can be satisfied when the demand is unessential and these (average) fractions are the values in column market saturation (sat.).

The first interesting observation is that the turnover of the follower and the market saturation in the FMAX strategy is higher than in the LMIN strategy in all our test cases, which corresponds to our intuition. According to this observation in the proportional and the partially binary case the turnover of the leader is always lower when the follower uses the LMIN strategy. However, a rather surprising result is that this is not always the case for the binary customer behavior, see Table 4. In some of the instances with $r = p = 10$ and most of the instances with $r = p = 20$ the leader objective value is higher for the LMIN strategy. This can be explained by two factors: On the one hand the model for the follower is easier to solve in the LMIN case and therefore the algorithm is able to perform more iterations so that the leader can possibly obtain better facility locations. On the other hand when the follower uses the LMIN strategy the leader wants to open the facilities in such a way that it is difficult for the follower to capture facilities of her. In some instances this can be more easily achieved and therefore the leader's turnover can be higher.

Table 4 Results of binary customer behavior with unessential demand.

			LMIN					FMAX				
Instance	r	p	obj^l	sd	obj^f	sd	sat.	obj^l	sd	obj^f	sd	sat.
Code1	10	10	1846,37	0,00	699,52	0,00	29,30%	1849,32	0,00	1539,73	0,00	39,00%
Code2	10	10	1927,16	0,00	988,90	0,00	27,72%	1929,70	0,00	1793,24	0,00	35,39%
Code3	10	10	1850,92	0,00	1164,79	0,00	32,25%	1855,38	0,01	1564,22	0,01	36,57%
Code4	10	10	1914,31	0,00	1313,22	0,00	32,51%	1918,90	0,00	1674,72	0,00	36,20%
Code5	10	10	1909,94	0,00	1189,45	0,00	30,26%	1912,81	0,00	1718,66	0,00	35,45%
Code6	10	10	1892,39	0,00	1000,04	0,00	30,39%	1898,67	0,00	1753,45	0,00	38,37%
Code7	10	10	1928,64	0,00	895,01	0,00	25,21%	1937,19	0,00	1819,43	0,00	33,54%
Code8	10	10	1889,22	0,00	995,05	0,00	30,18%	1893,17	0,00	1698,59	0,00	37,58%
Code9	10	10	1937,16	0,00	1001,99	0,00	28,27%	1943,06	0,00	1756,90	0,00	35,59%
Code10	10	10	1926,28	0,00	1165,26	0,00	30,23%	1931,94	0,00	1793,41	0,00	36,43%
Code1	15	15	2626,28	0,00	1270,15	0,00	44,84%	2630,42	0,41	2065,77	0,41	54,05%
Code2	15	15	2835,00	0,00	1148,96	0,00	37,87%	2835,93	4,31	2492,44	4,31	50,65%
Code3	15	15	2673,87	0,00	1602,77	0,00	45,73%	2675,32	5,63	2118,62	5,63	51,27%
Code4	15	15	2786,36	0,00	1576,55	0,00	43,95%	2791,15	0,00	2231,67	0,00	50,60%
Code5	15	15	2788,79	0,00	1433,94	0,00	41,23%	2790,91	3,26	2346,67	3,26	50,16%
Code6	15	15	2780,75	0,00	1635,72	0,00	46,40%	2786,19	0,26	2398,07	0,26	54,47%
Code7	15	15	2850,97	0,89	1453,86	10,88	38,44%	2857,61	4,30	2563,34	4,30	48,41%
Code8	15	15	2766,82	0,00	1429,97	0,00	43,91%	2769,97	2,24	2239,73	2,24	52,42%
Code9	15	15	2827,96	0,00	1553,82	0,00	42,15%	2832,91	1,53	2418,83	1,54	50,52%
Code10	15	15	2835,45	0,00	1686,00	0,00	44,22%	2841,64	1,69	2466,41	1,69	51,91%
Code1	20	20	3156,98	64,77	1549,71	132,88	54,17%	3153,85	51,35	2654,00	40,05	66,84%
Code2	20	20	3476,13	65,08	1813,75	212,00	50,28%	3515,71	54,99	3224,91	47,04	64,07%
Code3	20	20	3267,67	48,30	1806,11	131,34	54,26%	3227,89	64,64	2754,22	57,30	63,97%
Code4	20	20	3386,16	85,40	1985,39	132,71	54,11%	3375,96	62,21	2817,20	60,38	62,39%
Code5	20	20	3418,97	59,38	2011,84	134,34	53,02%	3402,50	60,47	2987,49	56,47	62,38%
Code6	20	20	3436,56	82,77	1807,69	152,24	55,10%	3356,40	80,46	2963,24	66,61	66,40%
Code7	20	20	3537,77	61,17	1978,53	155,31	49,26%	3548,15	63,04	3316,42	45,49	61,30%
Code8	20	20	3378,65	60,82	1714,22	133,31	53,29%	3355,12	58,21	2851,08	54,63	64,94%
Code9	20	20	3540,28	59,59	1981,12	148,48	53,11%	3494,47	51,85	3124,28	36,54	63,67%
Code10	20	20	3482,95	78,33	2137,71	134,22	54,96%	3463,84	81,00	3135,54	65,65	64,54%

The model for the follower's problem for proportional unessential customer behavior, especially FMAX, is still hard to solve so we again only compute greedy values for some of the instances, denoted by parantheses in Table 5. For these instances we do not state the market saturation and the objective values and standard deviations of the follower because we do not have exact results. However, compared to essential demands we were able to solve the follower's model for more instances and therefore get accurate results in more cases.

In Table 6 we see the results of partially binary customer behavior with unessential demands. The results show that for this scenario the results are very stable because the objective values have a very low standard deviation in many instances. The most interesting observation in this table is that, in contrast to the essential cases, in most instances the leader objective value (and the market saturation) is higher than in the proportional scenario. The reason for this behavior is that in the partially binary scenario more demand is satisfied by nearer facilities and therefore the total satisfied demand also increases.

From the results we conclude that in general the FMAX strategy is better because significantly more demand can be satisfied and the follower increases his

Table 5 Results of proportional customer behavior with unessential demand.

		LMIN					FMAX				
r	p	obj^{l}	sd	obj^{f}	sd	sat.	obj^{l}	sd	obj^{f}	sd	sat.
2	2	19,818	0,00	19,818	0,00	7,10%	30,460	0,00	29,331	0,00	10,72%
2	3	32,045	0,00	18,100	0,00	8,99%	42,091	0,00	26,735	0,00	12,33%
2	4	42,586	0,00	16,047	0,00	10,51%	52,266	0,00	25,160	0,00	13,88%
2	5	52,538	0,00	15,569	0,00	12,21%	61,266	0,00	23,528	0,00	15,20%
2	6	61,468	0,00	14,990	0,00	13,70%	70,285	0,00	22,218	0,00	16,58%
2	7	69,757	0,00	14,551	0,00	15,11%	78,613	0,00	20,634	0,00	17,79%
2	8	77,767	0,00	14,143	0,00	16,47%	85,893	0,00	19,610	0,00	18,91%
2	9	84,962	0,00	13,604	0,00	17,66%	91,963	0,00	19,118	0,00	19,91%
2	10	91,552	0,00	13,027	0,00	18,74%	98,666	0,03	18,557	0,07	21,01%
3	2	17,133	0,00	23,343	0,00	7,25%	28,049	0,00	40,476	0,00	12,28%
3	3	25,262	0,00	25,262	0,00	9,05%	39,386	0,00	38,039	0,00	13,88%
3	4	35,786	0,02	23,610	0,32	10,64%	49,724	0,03	35,434	0,03	15,26%
3	5	45,888	0,00	22,267	0,00	12,21%	59,172	0,00	33,332	0,00	16,58%
3	6	54,939	0,00	21,571	0,00	13,71%	67,564	0,00	31,682	0,00	17,79%
3	7	63,250	0,00	21,026	0,00	15,10%	74,954	0,00	29,902	0,00	18,79%
3	8	71,341	0,08	20,260	0,10	16,42%	82,241	0,00	28,839	0,00	19,91%
3	9	78,551	0,20	19,595	0,08	17,59%	89,195	0,07	28,040	0,13	21,01%
3	10	85,244	0,19	18,998	0,08	18,68%	95,494	0,39	27,384	0,21	22,02%
4	2	15,120	0,00	33,323	0,00	8,68%	(26,501)	0,00	–	–	–
4	3	22,935	0,00	28,380	0,00	9,20%	(37,586)	0,00	–	–	–
4	4	29,895	0,00	29,895	0,00	10,72%	47,448	0,00	45,056	0,00	16,58%
4	5	39,870	0,00	28,417	0,00	12,24%	56,606	0,00	42,641	0,00	17,79%
4	6	49,085	0,00	27,466	0,00	13,72%	64,863	0,13	40,647	0,14	18,91%
4	7	57,464	0,03	26,822	0,03	15,10%	72,384	0,03	39,126	0,16	19,98%
4	8	65,234	0,34	25,972	0,25	16,35%	79,469	0,25	37,534	0,30	20,97%
4	9	72,388	0,54	25,180	0,23	17,49%	86,197	0,53	36,634	0,27	22,01%
4	10	78,244	0,84	24,138	0,38	18,35%	91,158	1,06	35,784	0,33	22,75%
5	2	(13,856)	0,00	–	–	–	(25,474)	0,00	–	–	–
5	3	20,973	0,00	35,673	0,00	10,15%	(36,173)	0,00	–	–	–
5	4	27,804	0,00	32,685	0,00	10,84%	(45,780)	0,00	–	–	–
5	5	34,449	0,00	34,449	0,00	12,35%	(54,150)	0,00	–	–	–
5	6	43,715	0,01	33,071	0,02	13,76%	62,393*	0,05	49,228*	0,11	20,00%
5	7	52,077	0,13	32,331	0,17	15,13%	70,206	0,25	47,418	0,26	21,08%
5	8	59,373	0,77	31,206	0,36	16,23%	76,770	0,51	46,073	0,41	22,01%
5	9	65,956	1,14	30,357	0,30	17,26%	82,630	0,68	44,856	0,36	22,85%
5	10	70,972	2,02	29,086	0,38	17,93%	86,950	1,85	43,642	0,38	23,40%

*Not all of the 30 runs completed in the time limit

profit, too. However, if the follower wants to lower the turnover of the leader by all means the LMIN strategy might be useful but we could show that this is only valid for proportional and partially binary behavior and not for binary behavior. Compared to the essential demand cases we showed that while the complexity of the models for the follower's problem of the binary behavior increases, the complexity of the other two scenarios decreases and we got accurate results for more instances.

7 Conclusions and Future Work

In this work we presented bi-level mixed integer programming models for competitive facility location problems with different customer behavior as described in the literature. We used an evolutionary algorithm incorporating a complete solution archive for finding the best locations for the leader, which has already been successfully applied to binary and proportional behavior in our previous work.

Table 6 Results of partially binary customer behavior with unessential demand.

		LMIN					FMAX				
r	p	obj^{l}	sd	obj^{f}	sd	sat.	obj^{l}	sd	obj^{f}	sd	sat.
2	2	21,288	0,00	21,288	0,00	7,63%	34,168	0,00	31,450	0,00	11,76%
2	3	35,324	0,00	20,960	0,00	10,09%	47,015	0,00	30,165	0,00	13,83%
2	4	48,015	0,00	21,055	0,00	12,38%	59,276	0,00	29,247	0,00	15,86%
2	5	59,988	0,00	19,512	0,00	14,25%	71,004	0,00	27,864	0,00	17,72%
2	6	71,623	0,00	19,022	0,00	16,24%	82,540	0,00	27,079	0,00	19,64%
2	7	82,551	0,00	19,783	0,00	18,34%	93,484	0,00	25,939	0,00	21,40%
2	8	93,423	0,00	19,487	0,00	20,23%	103,987	0,30	25,514	0,65	23,21%
2	9	103,254	0,00	20,031	0,00	22,09%	114,234	0,00	25,246	0,00	25,00%
2	10	113,116	0,00	20,735	0,00	23,99%	122,132	1,36	24,473	0,38	26,27%
3	2	20,501	0,00	22,075	0,00	7,63%	32,744	0,00	44,798	0,00	13,90%
3	3	28,658	0,00	28,658	0,00	10,27%	45,339	0,00	43,072	0,00	15,84%
3	4	41,349	0,00	28,753	0,00	12,56%	57,582	0,00	41,117	0,00	17,69%
3	5	53,268	0,00	29,170	0,00	14,77%	69,171	0,00	39,802	0,00	19,53%
3	6	64,903	0,00	28,679	0,00	16,77%	80,790	0,00	38,398	0,00	21,36%
3	7	75,591	0,00	29,186	0,00	18,78%	91,440	0,00	37,178	0,00	23,05%
3	8	86,463	0,00	28,890	0,00	20,67%	101,820	0,00	36,985	0,00	24,88%
3	9	96,482	0,00	27,898	0,00	22,29%	111,967	0,02	36,416	0,20	26,59%
3	10	106,419	0,00	28,221	0,00	24,13%	122,981	0,31	35,902	0,04	28,47%
4	2	19,856	0,00	22,720	0,00	7,63%	31,007	0,00	57,367	0,00	15,84%
4	3	27,934	0,00	29,378	0,00	10,27%	44,102	0,00	55,005	0,00	17,76%
4	4	35,294	0,00	35,294	0,00	12,65%	56,457	0,00	53,602	0,00	19,72%
4	5	47,213	0,00	35,711	0,00	14,86%	67,858	0,00	51,352	0,00	21,36%
4	6	58,848	0,00	35,220	0,00	16,86%	79,037	0,00	49,204	0,00	22,98%
4	7	69,629	0,00	35,515	0,00	18,84%	89,714	0,00	48,228	0,00	24,72%
4	8	79,507	0,00	35,948	0,00	20,69%	100,461	0,00	47,720	0,00	26,56%
4	9	90,337	0,00	34,987	0,00	22,46%	110,602	0,03	46,858	0,16	28,22%
4	10	100,274	0,00	35,311	0,00	24,30%	120,672	0,07	46,275	0,33	29,92%
5	2	19,241	0,00	20,779	0,00	7,17%	30,831	0,00	68,490	0,00	17,80%
5	3	27,318	0,00	27,420	0,00	9,81%	43,493	0,00	66,877	0,00	19,78%
5	4	34,570	0,00	36,014	0,00	12,65%	55,232	0,00	64,410	0,00	21,44%
5	5	41,462	0,00	41,462	0,00	14,86%	66,663	0,00	61,991	0,00	23,06%
5	6	53,097	0,00	40,971	0,00	16,86%	77,940	0,00	59,831	0,00	24,69%
5	7	63,878	0,00	41,266	0,00	18,84%	88,616	0,00	58,855	0,00	26,43%
5	8	74,554	0,00	40,796	0,00	20,67%	98,942	0,03	58,530	0,12	28,22%
5	9	84,363	0,00	41,222	0,00	22,51%	109,581	0,00	57,526	0,00	29,95%
5	10	94,297	0,00	41,528	0,00	24,34%	119,869	0,00	56,054	0,00	31,53%

The solution representation is based on the leader facilities only and we used our developed lower level MIP models as well as a greedy method for the evaluation of candidate solutions. We showed that the model for the binary customer behavior scenario can be solved much more easily than the others and that the proportional case is by far the hardest to solve. The observation that the leader benefits more from proportional customer behavior than from partially binary behavior is also interesting. Even more interesting is that this changes when the demand is unessential. For the unessential demand model binary behavior is still the easiest and the proportional behavior still the hardest to solve but the former is getting harder while partially binary and proportional models are getting easier.

Our tests showed that our algorithmic approach is practically effective for CFLPs. Even for the more complex customer behavior scenarios our EA is able to find good solutions relatively fast. We compared our algorithm with previous state-of-the-art algorithms on the binary and proportional essential customer behavior scenarios and were able to outperform them in many cases.

Future research directions could be the development of a better approximation of the leader's objective value, e.g., by extending our greedy algorithms with a local search. When using a more elaborate solution evaluation we have obviously a tradeoff between accuracy and run-time. It would also be interesting to extend our models for different customer behavior to more realistic scenarios by taking opening costs of facilities into account to be able to maximize not only the turnover of the leader but also the profit. We also want to study the impact of different customer behaviors, i.e., compare the solutions obtained by the corresponding models in a more detail. This enables us to gain insights into the impact of different models on scenarios in practice.

References

1. Alekseeva, E., Kochetov, Y.: Matheuristics and Exact Methods for the Discrete $(r|p)$-Centroid Problem. In: E.G. Talbi (ed.) Metaheuristics for Bi-level Optimization, *Studies in Computational Intelligence*, vol. 482, pp. 189–219. Springer Berlin Heidelberg (2013)
2. Alekseeva, E., Kochetova, N., Kochetov, Y., Plyasunov, A.: A Hybrid Memetic Algorithm for the Competitive P-Median Problem. In: N. Bakhtadze, A. Dolgui (eds.) Information Control Problems in Manufacturing, vol. 13, pp. 1533–1537. International Federation of Automatic Control (2009)
3. Alekseeva, E., Kochetova, N., Kochetov, Y., Plyasunov, A.: Heuristic and Exact Methods for the Discrete $(r|p)$-Centroid Problem. In: P. Cowling, P. Merz (eds.) Evolutionary Computation in Combinatorial Optimization, *Lecture Notes in Computer Science*, vol. 6022, pp. 11–22. Springer Berlin Heidelberg (2010)
4. Bhadury, J., Eiselt, H., Jaramillo, J.: An alternating heuristic for medianoid and centroid problems in the plane. Computers & Operations Research **30**(4), 553–565 (2003)
5. Biesinger, B., Hu, B., Raidl, G.: An evolutionary algorithm for the leader-follower facility location problem with proportional customer behavior. In: P.M. Pardalos, M.G. Resende, C. Vogiatzis, J.L. Walteros (eds.) Learning and Intelligent Optimization, Lecture Notes in Computer Science, pp. 203–217. Springer International Publishing (2014)
6. Biesinger, B., Hu, B., Raidl, G.: A hybrid genetic algorithm with solution archive for the discrete $(r|p)$-centroid problem. Tech. Rep. TR 186–1–14–03, Vienna University of Technology, Vienna, Austria (2014). Submitted to Journal of Heuristic
7. Fernández, J., Hendrix, E.M.: Recent insights in huff-like competitive facility location and design. European Journal of Operational Research **227**(3), 581–584 (2013)
8. Gusfield, D.: Algorithms on strings, trees, and sequences: computer science and computational biology. Cambridge University Press, New York, NY, USA (1997)
9. Hakimi, S.: On locating new facilities in a competitive environment. European Journal of Operational Research **12**(1), 29–35 (1983)
10. Hakimi, S.: Locations with spatial interactions: Competitive locations and games. In: P.B. Mirchandani, R.L. Francis (eds.) Discrete Location Theory, pp. 439–478. Wiley, New York (1990)
11. Hotelling, H.: Stability in competition. The Economic Journal **39**(153), 41–57 (1929)
12. Hu, B., Raidl, G.: An evolutionary algorithm with solution archives and bounding extension for the generalized minimum spanning tree problem. In: T. Soule (ed.) Proceedings of the 14th Annual Conference on Genetic and Evolutionary Computation (GECCO 2012), pp. 393–400. ACM Press, Philadelphia, PA, USA (2012)
13. Kochetov, Y., Kochetova, N., Plyasunov, A.: A matheuristic for the leader-follower facility location and design problem. In: H. Lau, P. Van Hentenryck, G. Raidl (eds.) Proceedings of the 10th Metaheuristics International Conference (MIC 2013), pp. 32/1–32/3. Singapore (2013)
14. Kress, D., Pesch, E.: $(r|p)$-centroid problems on networks with vertex and edge demand. Computers & Operations Research **39**, 2954–2967 (2012)
15. Küçükaydin, H., Aras, N., Altınel, I.K.: 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 (2011)

16. Laporte, G., Benati, S.: Tabu Search Algorithms for the $(r|X_p)$-medianoid and $(r|p)$-centroid Problems. Location Science **2**, 193–204 (1994)
17. Mladenović, N., Brimberg, J., Hansen, P., Moreno-Pérez, J.A.: The p-median problem: A survey of metaheuristic approaches. European Journal of Operational Research **179**(3), 927 – 939 (2007)
18. Noltemeier, H., Spoerhase, J., Wirth, H.C.: Multiple voting location and single voting location on trees. European Journal of Operational Research **181**(2), 654–667 (2007)
19. Raidl, G., Hu, B.: Enhancing genetic algorithms by a trie-based complete solution archive. In: P. Cowling, P. Merz (eds.) Evolutionary Computation in Combinatorial Optimization, *Lecture Notes in Computer Science*, vol. 6022, pp. 239–251. Springer Berlin Heidelberg (2010)
20. Roboredo, M., Pessoa, A.: A branch-and-cut algorithm for the discrete $(r|p)$-centroid problem. European Journal of Operational Research **224**(1), 101–109 (2013)
21. Saidani, N., Chu, F., Chen, H.: Competitive facility location and design with reactions of competitors already in the market. European Journal of Operational Research **219**(1), 9–17 (2012)
22. Sáiz, M.E., Hendrix, E.M., Pelegrín, B.: On nash equilibria of a competitive location-design problem. European Journal of Operational Research **210**(3), 588–593 (2011)
23. Serra, D., Colomé, R.: Consumer choice and optimal locations models: Formulations and heuristics. Papers in Regional Science **80**(4), 439–464 (2001)
24. Suárez-Vega, R., Santos-Peñate, D., Pablo, D.G.: Competitive Multifacility Location on Networks: the $(r|X_p)$-Medianoid Problem. Journal of Regional Science **44**(3), 569–588 (2004)