Heuristic and Exact Methods for the Discrete $(r \mid p)$-Centroid Problem*

Ekaterina Alekseeva, Nina Kochetova,
Yury Kochetov, and Alexandr Plyasunov

Sobolev Institute of Mathematics,
Novosibirsk State University, Russia
{ekaterina2,nkochet,jkochet,apljas}@math.nsc.ru

Abstract. In the discrete $(r \mid p)$-centroid problem two decision makers, a leader and a follower, compete to attract clients from a given market. The leader opens p facilities, anticipating that the follower will react to the decision by opening his own r facilities. The decision makers try to maximize their own profits. This Stackelberg game is Σ_2^P-hard. So, we develop a hybrid memetic algorithm for it. A probabilistic tabu search heuristic is applied for improving the offsprings. To obtain an upper bound, we reformulate the problem as a mixed integer program with an exponential number of constraints and variables. Selecting some of them, we get the desired upper bound. To find optimal solutions, we iteratively modify the subset of the constraints and variables. This approach is tested on the benchmarks from the library *Discrete Location Problems*. The optimal solutions are found for $r = p = 5$, 100 clients, and 100 facilities.

1 Introduction

In this paper we study a competitive facility location model with two noncooperative decision makers: the leader and the follower. They compete to attract clients from a given market and wish to maximize their own profits. First, the leader opens p facilities. Later on, the follower opens r facilities. In fact, we have a noncooperative Stackelberg game. Following Hakimi [7], we call it the *discrete $(r \mid p)$-centroid problem.*

It is known that the problem is Σ_2^P-hard [9]. So, we deal with the more hard problem than any problem in the class NP. Polynomially solvable cases and complexity results can be found in [7], [9]. In order to find near optimal solutions, we present the game as a 0–1 linear bi-level problem and develop a hybrid memetic algorithm (HMA). The probabilistic tabu search heuristic (PTS) is used to improve each element of the population. To compute the leader profit, we solve the follower problem by commercial software.

To get an upper bound for this maximization problem, we rewrite the bi-level problem as a single level mixed integer linear program with an exponential

* This work was partly supported by the RFBR grant 08-07-00037, ADTP grant 2.1.1/3235.

P. Cowling and P. Merz (Eds.): EvoCOP 2010, LNCS 6022, pp. 11–22, 2010.

number of constraints and variables. A similar approach is suggested in [10] for partial enumeration. If we extract a small family of constraints and variables, we get an upper bound. The PTS heuristic is used for generating the family. For exact approach we apply the idea of column generation. Computational experiments for Euclidean test instances from the benchmark library *Discrete Location Problems* (http://math.nsc.ru/AP/benchmarks/Competitive/p_med_comp_eng.html) indicate that the new HMA lower bound dominates the previous ones and the exact method allows to find the global optimum for $p = r = 5$, 100 facilities, and 100 clients.

The paper is organized as follows. In Section 2, we present the mathematical model. In Section 3, the lower bounds are discussed. We describe four lower bounds; three of them are quite simple, while the last one is based on the metaheuristic approach for the bi-level mathematical formulation. In Section 4, the upper bound based on the new reformulation of the problem as a mixed integer linear program with an exponential number of constraints and variables is presented. An exact column generation method is studied in Section 5. Computational results and conclusions are discussed in Sections 6 and 7.

2 Problem Formulation

We are given a set $I = \{1, \dots, m\}$ of facilities and a set $J = \{1, \dots, n\}$ of clients. A matrix (g_{ij}) defines the distances between clients and facilities. If client j is serviced from a facility, he gives a profit $w_j > 0$. The leader and the follower open facilities. First, the leader opens p facilities. Later on, the follower opens r facilities. Each client chooses the closest open facility. We need to find p facilities for the leader to maximize his profit. Let us present this game as a linear 0–1 bi-level programming problem. We define the decision variables [2]:

$$x_i = \begin{cases} 1 & \text{if facility } i \text{ is opened by the leader,} \\ 0, & \text{otherwise,} \end{cases}$$

$$y_i = \begin{cases} 1 & \text{if facility } i \text{ is opened by the follower,} \\ 0, & \text{otherwise,} \end{cases}$$

$$z_j = \begin{cases} 1 & \text{if client } j \text{ is serviced by the leader,} \\ 0 & \text{if client } j \text{ is serviced by the follower.} \end{cases}$$

For a given solution x, we can define the set of facilities which allow to capture client j by the follower: $I_j(x) = \{i \in I \,|\, g_{ij} < \min_{l \in I}(g_{lj} \,|\, x_l = 1)\}$, $j \in J$. Note that we consider *conservative* clients. If a client has the same distances to the closest leader and the closest follower facilities, he prefers the leader facility. So, the follower never opens a facility at a site where the leader has a facility [7]. Now the model can be written as a linear 0–1 bi-level programming problem [8]:

$$\max_x \sum_{j \in J} w_j z_j^*(x) \tag{1}$$

s.t.

$$\sum_{i \in I} x_i = p, \tag{2}$$

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

where $z_j^*(x)$ is a component of the optimal solution of the follower problem:

$$\max_{y,z} \sum_{j \in J} w_j (1 - z_j) \tag{4}$$

s.t.

$$\sum_{i \in I} y_i = r, \tag{5}$$

$$1 - z_j \le \sum_{i \in I_j(x)} y_i, \quad j \in J, \tag{6}$$

$$x_i + y_i \le 1, \quad i \in I, \tag{7}$$

$$y_i, z_j \in \{0, 1\}, \quad i \in I, j \in J. \tag{8}$$

The objective function (1) defines the total profit of the leader. Equation (2) guarantees that the leader opens exactly p facilities. The objective function (4) defines the total profit of the follower. Equation (5) guarantees that the follower opens exactly r facilities. Constraints (6) determine the values of z by the decision variables y of the follower. Constraints (7) allow to open a facility by at most one decision maker. As we have mentioned, the constraints (7) are redundant. Nevertheless, we use them to reduce the feasible domain of the follower problem. Note that the optimal value for the problem does not change if we replace 0–1 variables z_j by continuous variables $0 \le z_j \le 1$. So, we deal with the mixed integer linear programming problem for the follower.

Matrix (g_{ij}) is used to define the set $I_j(x)$. In fact, we can use not only the distances, but any kind of preferences for the clients. For example, a client may prefer a facility with the minimal traveling and waiting time rather than with the minimal distance. In [12] the facility location models with general client preferences are studied. With almost no change, we can use the preferences in the definition of set $I_j(x)$. However, we preserve the distances for simplicity.

3 Lower Bounds

An arbitrary feasible solution of the problem (1)–(8) produces a lower bound. For a given solution x, we have to solve the problem (4)–(8) to get a feasible solution. It is an NP–hard problem [7]. We use the commercial CPLEX software for it. So, the rest of this section is devoted to describe various strategies for the selection of solution x.

The first and the simplest strategy is to ignore the follower [1]. The leader opens facilities to minimize the total distance between clients and his facilities. He wishes to service all clients and solves the classical p-median problem.

We use an optimal solution of this problem as solution x for the lower bound. This strategy is not so bad despite ignoring the follower. As we can see in our computational experiments, the leader loses more than half of the market, but we can improve the lower bound by a few percents only.

The second strategy is more sophisticated. The leader anticipates that the follower will react to his decision. So, $(p+r)$ facilities will be opened. According to the second strategy [1], the leader solves the $(p+r)$-median problem and opens the p most profitable facilities. As we will see below, this strategy is not perfect.

The third strategy is alternate. It is suggested for continuous locations in [5]. This heuristic is iterative. For a given solution of one decision maker, we find the optimal solution for another one. In discrete case this strategy produces a cycle. The best solution in the cycle is the result of the approach. If we use the previous strategies to create a starting solution, we can improve the profit of the leader. Surely, it is a more time consuming procedure.

The most powerful strategy is to solve the problem (1)–(8). We develop a hybrid memetic algorithm where a tabu search approach is used to improve the elements of the population [2]. Now we present the general framework of the method.

Hybrid memetic algorithm

1 Generate an initial population of the leader solutions.
2 Repeat until the stopping condition is met:
  2.1 Select two solutions x^1, x^2 from the population.
  2.2 Create a solution x by a recombination of x^1, x^2.
  2.3 Apply the random modification to x.
  2.4 Improve the solution by PTS heuristic.
  2.5 Update the population.
3 Return the best found solution.

We use the total number of iterations 2.1–2.5 as the stopping condition.

3.1 Initial Population

To create a high quality initial population at Step 1 of the framework, we apply the standard local improvement algorithm with random starting points. The well–known Swap neighborhood for the p-median problem is used for the improvements. Remember that we have to solve the problem (4)–(8) in order to compute the objective function value for each element of the neighborhood. The Swap neighborhood contains $p(m-p)$ elements. It is a time–consuming procedure. To reduce the running time, we use the first improvement pivoting rule, the randomization of neighborhood, and solve the linear programming relaxations to estimate the neighboring solutions.

The efficiency and robustness of the memetic algorithm depend on the population. We need different local optima. So, a new local optimum obtained is included into the population if the Hamming distance from this solution to each solution in the population is at least a given threshold. In our computational experiments, we use the threshold $\lceil 0,6p \rceil$.

3.2 Main Operators and Parameters

The selection, recombination, random modification (mutation), and replacement operators are used in the framework. The well–known tournament selection procedure [11] is applied to pick two solutions x^1, x^2. We select k solutions from the population at random and choose the best one as a parent. In our experiments, we put $k = 5$.

The recombination or crossover operator is a variant of the well-known uniform crossover [11]. The new solution x will contain all open facilities which are common for the parents. The rest of the open facilities are chosen at random with probability 0,5 from solutions x^1, x^2.

To involve a certain diversification, we use random modification of the offspring. The common bit–flip mutations are not appropriate for the problem. We may get an unfeasible solution. Instead, we produce some random modification according to the Swap neighborhood.

To update the population at Step 2.5, we use the steady–state–no–duplicates techniques. We check that no duplicate solutions are added to the population. Moreover, we calculate the Hamming distance between the new solution and the population and update the population if the distance is at least the threshold $\lceil 0,6p \rceil$.

3.3 Tabu Search

In the well–known memetic algorithms, the standard local improvement procedure is applied to each element of the population. The algorithm finds local optima, and this feature promotes for finding global optimum or near optimal solutions. In our computational experiments for the case $w_j = 1, j \in J$, we discover a lot of local optima and plateaus. The standard local improvement procedure is not efficient for this case [1]. Therefore, we develop the PTS heuristic [6] and apply it instead of the local improvement. In [4] a tabu search algorithm is studied for the problem but a greedy procedure is applied for the follower problem. In this case we have no optimal solutions. The tabu search may produce solutions for the leader where the greedy approach has significant deviations from the optimum in the problem (4)–(8). Hence, this idea can be useful only for the instances with small p and r or particular cases of the problem. In the general case we have to apply the branch and bound method for finding optimal solution of the follower problem.

To reduce the running time at each step, we use a randomized neighborhood $N_q(x),\ q > 0$. It is the random part of the Swap neighborhood, where each element is included into the set $N_q(x)$ with probability q independently from other elements.

In order to evaluate elements of the neighborhood, we need to solve the follower problem. As mentioned above, it is an NP-hard problem. To reduce the running time, we replace the problem by its linear programming relaxation. Hence, we have a polynomial time procedure for finding the best element in the neighborhood. Below, we present the general framework of the PTS algorithm.

<u>Probabilistic Tabu Search</u>

1 Get offspring x from HMA and put $Tabu = \emptyset$.
2 Repeat until the stopping condition is met:
  2.1 Generate the neighborhood $N_q(x)$.
  2.2 If $N_q(x) \setminus Tabu \neq \emptyset$, then find the best element x' in the set $N_q(x) \setminus Tabu$; else $x' := x$.
  2.3 Put $x := x'$, update $Tabu$.
3 Return the best found solution.

The set $Tabu$ for the current solution contains some solutions from the Swap neighborhood. The pairs of swapping facilities are stored during a certain number of iterations, and the corresponding solutions are included into the set $Tabu$. We use the PTS algorithm for finding the high quality offspring at Step 2.4 of the HMA framework. Moreover, we collect the high quality solutions for the follower. As we will see in Section 4, any family of follower solutions allows us to compute an upper bound for the maximal profit (1) of the leader.

4 Upper Bounds

Let $\mathcal{F}$ be a family of the follower solutions. For $y \in \mathcal{F}$, $j \in J$, we introduce a set

$$I_j(y) = \big\{ i \in I \mid g_{ij} \leq \min_{l \in I} (g_{lj} | y_l = 1) \big\}.$$

The set $I_j(y)$ shows the facilities for the leader which allow him to keep the client j if the follower uses solution y. Now we rewrite the bi-level problem as a single level problem with an exponential number of constraints and variables. Let us introduce new variables:

$$z_{jy} = \begin{cases} 1 & \text{if client } j \text{ is serviced by the leader and} \\ & \text{the follower uses a solution } y, \\ 0 & \text{if client } j \text{ is serviced by the follower and} \\ & \text{the follower uses a solution } y, \end{cases} \qquad j \in J, y \in \mathcal{F},$$

$W \geq 0$ is the total profit of the leader.

If the family $\mathcal{F}$ contains all possible solutions for the follower, then the problem (1)–(8) is equivalent to the following linear 0–1 program:

$$\max W \tag{9}$$

s.t.

$$\sum_{j \in J} w_j z_{jy} \geq W, \quad y \in \mathcal{F}, \tag{10}$$

$$z_{jy} \leq \sum_{i \in I_j(y)} x_i, \quad j \in J, y \in \mathcal{F}, \tag{11}$$

$$\sum_{i \in I} x_i = p, \tag{12}$$

$$x_i, z_{jy} \in \{0,1\}, i \in I, j \in J, y \in \mathcal{F}. \tag{13}$$

The objective function indicates the goal of the leader. The constraints (10) guarantee the best answer of the follower. The constraints (11) determine the market share for each follower solution. If the leader has no facilities in the set $I_j(y)$, then client j is serviced by a follower facility.

Note that the optimal value of the problem does not increase if we replace 0–1 variables z_{jy} by continuous variables $0 \le z_{jy} \le 1$. So, we get the mixed integer linear programming problem with m Boolean variables x_i, an exponential number of continuous variables z_{jy}, and constraints (10), (11).

In order to get an upper bound, we select a small subset of the *strong* solutions for the follower. Denote by $W(\mathcal{F})$ the optimal value of the problem (9)–(13) for the family $\mathcal{F}$. The most difficult task is finding an appropriate family. We produce it by the PTS algorithm at Step 2.4 of the HMA framework. So, we use metaheuristics to get lower and upper bounds.

5 An Exact Method

Let $x(\mathcal{F})$ be the optimal solution of the problem (9)–(13). The corresponding optimal solution of the follower problem (4)–(8) denoted by $y(\mathcal{F}) = y^*(x(\mathcal{F}))$, $z(\mathcal{F}) = z^*(x(\mathcal{F}))$. This solution defines a lower bound $LB(\mathcal{F}) = \sum_{j \in J} w_j z_j^*(\mathcal{F})$. If $LB(\mathcal{F}) = W(\mathcal{F})$, we have the optimal solution for the bi-level problem (1)–(8). Otherwise, we enlarge the family by adding $y(\mathcal{F})$ and repeat the calculations. This iterative method can be described as follows.

<u>Iterative exact method</u>

1 Choose an initial family $\mathcal{F}$.
2 Find the solution $x(\mathcal{F})$ and upper bound $W(\mathcal{F})$.
3 Solve the follower problem and find $y(\mathcal{F}), LB(\mathcal{F})$.
4 If $W(\mathcal{F}) = LB(\mathcal{F})$ then return the best found solution and STOP.
5 Include the solution $y(\mathcal{F})$ into the family $\mathcal{F}$ and go to Step 2.

Let us verify that the method is exact and finite indeed. Assume that we solve the follower problem at Step 3 and find $y(\mathcal{F})$, but $y(\mathcal{F}) \in \mathcal{F}$. From (10) we have $LB(\mathcal{F}) = \sum_{j \in J} w_j z_j^*(\mathcal{F}) = \sum_{j \in J} w_j z_{jy(\mathcal{F})} \ge W(\mathcal{F})$. So, $LB(\mathcal{F}) = W(\mathcal{F})$ and $x(\mathcal{F})$ is the optimal solution of the bi-level problem. The method is finite because $|\mathcal{F}| \le \binom{m}{r}$.

Step 2 is the most time consuming. We have to solve a large scale optimization problem. If we use the branch and bound method, we get $W(\mathcal{F})$ and $x(\mathcal{F})$. But the method spends a lot of time proving optimality. Actually, we need the solution only. Therefore, we may reduce the running time if replace the problem (9)–(13) by the following feasibility problem. By W^* denote the optimum for the bi-level problem (1)–(8) and consider the following system:

$$\sum_{j \in J} w_j z_{jy} > W^*, \quad y \in \mathcal{F}, \tag{14}$$

$$z_{jy} \leq \sum_{i \in I_j(y)} x_i, \quad y \in \mathcal{F}, j \in J, \tag{15}$$

$$\sum_{i \in I} x_i = p, \tag{16}$$

$$x_i \in \{0,1\}, 0 \leq z_{jy} \leq 1, \quad i \in I, j \in J, y \in \mathcal{F}. \tag{17}$$

If we have a feasible solution $x(\mathcal{F})$ for it, we include $y(\mathcal{F})$ into family $\mathcal{F}$ and repeat the calculations. Otherwise, we can stop the search with the appropriate family. The feasibility problem is easier. We do not need to prove optimality. We may apply the feasibility pump [3], the branch and bound method with convenient objective function, or metaheuristics again.

Of course, we do not know the optimal value W^*. So, we use the best found value $W' \leq W^*$ by the HMA and update it during the search. Now the framework of the modified exact method is the following.

Modified exact method

1 Apply HMA to create an initial family $\mathcal{F}$ and W'.
2 Find feasible solution $x(\mathcal{F})$ for the system (14)–(17).
  If it is infeasible then return the best found solution and STOP.
3 Solve the follower problem and find $y(\mathcal{F}), LB(\mathcal{F})$.
4 If $W' < LB(\mathcal{F})$ then $W' := LB(\mathcal{F})$.
5 Include $y(\mathcal{F})$ into family $\mathcal{F}$ and go to Step 2.

Further improvements of the method can deal with decreasing the family from time to time or generating several feasible solutions at Step 2.

6 Computational Results

The developed memetic algorithm and the exact method were coded in GAMS (General Algebraic Modeling System) and tested on the instances from the benchmark library *Discrete Location Problems.* For all instances, clients and facilities are in the same sites, $I = J$. The elements of matrix (g_{ij}) are Euclidean distances between points i, j in the two dimensional Euclidean plane. The points are chosen at random uniformly in the square 7000×7000. All experiments are carried out at the PC Pentium Intel Core 2, 1.87 GHz, RAM 2 Gb, running under the Windows XP Professional operating system.

In the first series of experiments we compare the lower and upper bounds. Table 1 shows computational results for the instances with $n = m = 100$, $p = r = 10$, $w_j = 1, j \in J$. The first column indicates the names of the instances. Columns $LB(p)$, $LB(p + r)$, AH, HMA present the lower bounds for the strategies based on the classical p-median problem, the $(p+r)$-median problem [1], the alternating heuristic [5], and the hybrid memetic algorithm, respectively. Note that the HMA lower bound dominates the other ones. Nevertheless, the difference between HMA and $LB(p)$ bounds is small. For the instance 511, these bounds coincide. We may conclude that $LB(p)$ is a good approximation for the leader behavior. Moreover, the running time for the $LB(p)$ is a few seconds only. For the HMA bound, we need about 7 hours of running time if we terminate calculations after 100 iterations with the population of size 25 and 100 iterations of the PTS algorithm for the local improvement to each individual. The alternating heuristic produces good approximations as well. It is more time consuming than $LB(p)$ strategy but can get a better lower bound. In brackets, we show the length of the cycle for the heuristic. Columns UB_0 and W show

Table 1. Lower and upper bounds

Instance	$LB(p)$	$LB(p+r)$	AH	HMA	UB_0	W
111	41	31	49 (2974)	50	74	54
211	41	36	45 (287)	49	70	57
311	46	41	44 (444)	48	73	55
411	41	39	45 (813)	49	72	58
511	48	40	47 (138)	48	70	55
611	42	39	46 (3607)	47	67	57
711	49	37	48 (675)	51	73	55
811	42	37	44 (255)	48	74	55
911	47	35	46 (2486)	49	68	54
1011	46	33	47 (4963)	49	70	54

two upper bounds. To compute UB_0, we need to rank the facilities for the follower [8]. We suppose that the follower uses this ranking instead of the optimal strategy. In this case the bi-level problem can be presented as a mixed integer linear program. An optimal value of the program is the upper bound UB_0. In our experiments, we use the ranking obtained by the Lagrangian relaxations [1]. In order to compute W, we need family $\mathcal{F}$. As mentioned above, we collect optimal solutions of the follower by the PTS algorithm at Step 2.4 of the HMA framework. The cardinality of the family is 400. Table 1 shows that the upper bound W dominates UB_0 substantially. Nevertheless, we cannot prove the optimality of the HMA solutions by substantially increasing the families. We believe that the HMA solutions are optimal. So, we need more intelligent search strategies for creating the families.

The second series of experiments is devoted to the modified exact method. Our goal is to investigate the influence of parameters p and r on the optimal families

Table 2. Optimal solutions

	$w_j = 1$			$w_j \in (0, 200)$		
Instance	*Opt*	*Iter*	*Time*	*Opt*	*Iter*	*Time*
111	47	123	120	4139 (47%)	98	65
211	48	69	60	4822 (45%)	127	37
311	45	231	3600	4215 (45%)	262	5460
411	47	111	150	4678 (47%)	128	900
511	47	106	120	4594 (44%)	190	720
611	47	102	90	4483 (47%)	121	660
711	47	115	180	5153 (46%)	167	2550
811	48	67	42	4404 (46%)	190	720
911	47	108	160	4700 (45%)	247	2520
1011	47	124	165	4923 (48%)	83	30

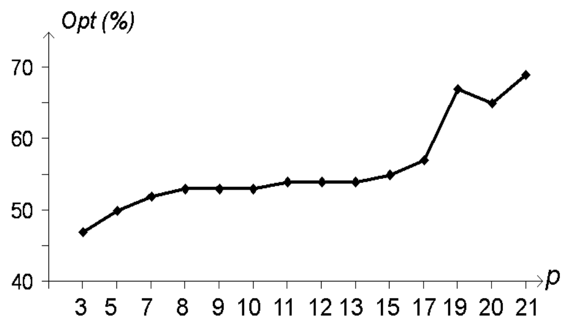

Fig. 1. The segment of the leader in percents, $p = r$

or the number of iterations, and study the market share. Table 2 presents computational results for Euclidean instances from the same library. For all instances, we have $n = m = 100$, $p = r = 5$ and two classes of weights: $w_j = 1$ and $w_j \in (0, 200), j \in J$. Note that up to now there are no reported results for $m > 70$. Columns *Opt* show optimal values for the instances and the leader profits in percents (market share). Columns *Iter* present the total number of iterations of the modified exact method. Columns *Time* indicate the running time, in minutes, for the method excluding the time for the HMA heuristic. We can see that the follower gets more than 50% of the market. He has some advantages for small p and r. For large values of p and r the leader has some advantages. Figure 1 indicates the segment of the leader, in percents, for $p = r$, $n = m = 50$, $w_j \in (0, 200), j \in J$.

The segment increases when p and r grow. The decision makers obtain a half of the market for $p = r = 7$. So, the leader should open many facilities to control the most of the market. Of course, his segment decreases when r grows and vice versa.

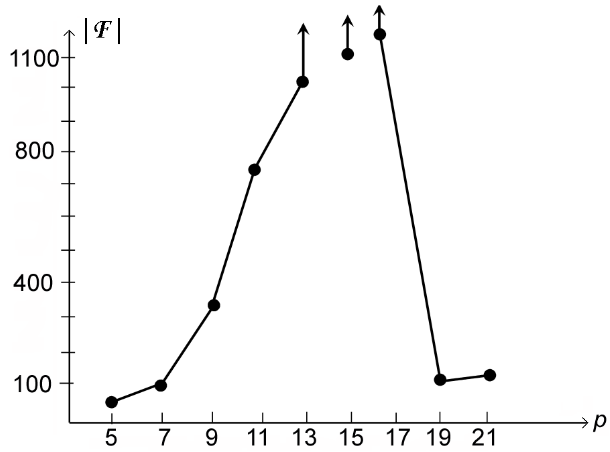

Fig. 2. The number of iterations for the modified exact method, $p = r$

In the third series of experiments we study the families of the follower solutions. Figure 2 shows the cardinalities of the families (the total number of iterations) for $n = m = 50$, $p = r$, $w_j \in (0, 200), j \in J$. The problem is easy when p and r are small or large. The total number of iterations is about one hundred. The problem becomes hard when $11 \leq p \leq 17$. We need more than a thousand follower solutions in the family. We guess that the case $p = r = [m/3]$ is the most difficult for the method even if we have the optimal value W^*. Huge family is a reason why we cannot prove optimality of the HMA solutions for $p = r = 10$.

7 Conclusions

We consider the well-known discrete $(r \mid p)$-centroid problem and develop a hybrid memetic algorithm for finding near optimal solutions and the exact column generation method. The problem is Σ_2^P–hard. We use commercial software to compute the objective function values for the feasible solutions of the leader. These solutions are used in the evolutionary algorithm as population members. A probabilistic tabu search heuristic is applied for improving the offspring at each step of the evolution. In order to get an upper bound, we reformulate the bi-level problem as a single level mixed integer programming problem with an exponential number of constraints and variables. Metaheuristics are used to collect an appropriate subset of the constraints and variables. To find the global optimum, we develop the modified iterative method. The feasibility subproblem is solved at each iteration. It seems interesting to study metaheuristics for the subproblem later on.

References

1. Alekseeva, E., Kochetova, N.: Upper and lower bounds for the competitive p-median problem. In: Proceedings of XIV Baikal International School–Seminar Optimization Methods and Their Applications, Irkutsk, vol. 1, pp. 563–569 (2008) (in Russian)
2. Alekseeva, E., Kochetova, N., Kochetov, Y., Plyasunov, A.: A hybrid memetic algorithm for the competitive p-median problem. In: Preprints of the 13th IFAC Symposium on Information Control Problems in Manufacturing, Moscow, pp. 1516–1520 (2009)
3. Bertacco, L., Fischetti, M., Lodi, A.: A feasibility pump heuristic for general mixed–integer problems. Discrete Optimization 4(1), 63–76 (2007)
4. Benati, S., Laporte, G.: Tabu search algorithms for the $(r|X_p)$–medianoid and $(r|p)$–centroid problems. Location Science 2(4), 193–204 (1994)
5. Bhadury, J., Eiselt, H., Jaramillo, J.: An alternating heuristic for medianoid and centroid problems in the plane. Computers and Operations Research 30, 553–565 (2003)
6. Glover, F., Laguna, M.: Tabu Search. Kluwer Acad. Publ., Boston (1997)
7. Hakimi, S.L.: Locations with spatial interactions: competitive locations and games. In: Mirchandani, P.B., Francis, R.L. (eds.) Discrete Location Theory, pp. 439–478. Wiley & Sons, Chichester (1990)
8. Kochetov, Y.A., Kononov, A.V., Plyasunov, A.V.: Competitive facility location models. Computational Mathematics and Mathematical Physics 49, 994–1009 (2009)
9. Noltermeier, H., Spoerhose, J., Wirth, H.C.: Multiple voting location and single voting location on trees. European J. Oper. Res. 181, 654–667 (2007)
10. Rodriguez, C.M.C., Perez, J.A.M.: Multiple voting location problems. European J. of Oper. Res. 191(2), 437–453 (2008)
11. Sastry, K., Goldberg, D., Kendall, G.: Genetic algorithms. In: Burke, E.K., Kendall, G. (eds.) Search Methodologies. Introductory Tutorials in Optimization and Decision Support Techniques, pp. 97–126. Springer, Heidelberg (2005)
12. Vasil'ev, I.L., Klimentova, K.B., Kochetov, Y.A.: New lower bounds for the facility location problem with clients' preferences. Computational Mathematics and Mathematical Physics 49, 1010–1020 (2009)