Lagrangian relaxation approach for capacitated p-center problem with backup center and budget constraint

Ebrahim Rezaee Nik[a] and Reza Hasanzadeh Ataei[b]

[a]Assistant Professor of Industrial Engineering, Sadjad university of technology
Tel: +985136029404, Fax: +985136029110, E-mail: rezaeenik@sadjad.ac.ir

[b]MSc Student of Industrial Engineering, Sadjad university of technology
Tel: +989158849584, E-mail: rezaataie88@gmail.com

Abstract

The *p*-center problem involves determination of locations of *p* facilities while minimizing the maximum distance between demand points and facilities. The main application areas of *p*-center problem are emergency service locations such as fire and police stations, hospitals and ambulance services. This paper deals with a generalized version of the capacitated *p*-center problem. The model takes into account the possibility that a center might suffer a disruption and assumes that every site will be covered by its closest available center. We present the mathematical model and use lagrangian relaxation for obtaining the suitable lower bound. Also, the CPLEX solver and Lagrangian relaxation method used to solve the model and high performance of the Lagrangian relaxation method is shown in numerical example.

Keywords: *P-center* problem, capacitated *p*-center problem, Lagrangian relaxation

1. Introduction

The p-center problem (pCP) is a well-known discrete optimization location problem which consists of locating p centers out of n sites and assigning the remaining n-p sites to the centers so as to minimize the maximum distance (cost) between a site and the corresponding center [1]. Facility location decisions play a prominent role in strategic planning of many firms, companies and governmental organizations. Deciding where to locate a new warehouse for a factory, where to place the fire stations, where to open a new branch of a bank and where to open a new store for chain stores are practical instances of facility location problems. These decisions are not only costly and irreversible (or at least very costly and time consuming to reverse), but they use up a lot of organization's resources also [2].

The p-center problem is one of the well-known NP-hard discrete location problems [3]. This problem can be partitioned into the uncapacitated and capacitated cases. The uncapacitated case is a basic p-center problem that does not include the demands of clients and capacities of facilities [4]. In the capacitated version of the *p*-center problem, each client is labelled with some quantity of demand, and assignment of clients to facilities is constrained with capacity restrictions of facilities (i.e., the total demands of clients assigned to a certain facility cannot exceed the facility's capacity). Namely, the capacitated *p*-center problem can be articulated as locating *p* capacitated facilities and assigning clients to them within capacity restrictions so as to minimize

maximum distance between a client and the facility it is assigned to [5].
Some of the potential applications of the p-center model would be in:

- Quick services (hospital emergency services, fire stations, police stations, …)
- Computer network services (location of the data files)
- Distribution (warehouses, garages, …)
- Military purpose
- Government and general (parks, hotels, …)
- Location-allocation for post boxes and bus stops

Since we have a minimax objective function for the p-center model, it seems that it would be most applicable to emergency cases [6].
Absolute 1-center problem is originally introduced by Hakimi in 1964. The absolute p-center problem for p>1 is initially introduced again by Hakimi in 1965. Hakimi, Schmeichel, and Pierce present improvements and generalizations for the 1-center problem proposed by Hakimi [7]. Dearing and Francis developed a formulation for the weighted absolute 1-center problem [7]. Elloumi, Labbe, and Pochet introduce a new formulation for the vertex restricted p-center problem that is based on solving a set-covering problem [7].
Facility location problems are known as hard to solve problems. Thus, this kind of hard to solve problems requires the use of computer technology. The development in computer technology brings new conveniences to the solution approaches.
Gonzalez [6] introduced a greedy heuristic for the *p*-center problem. Hochbaum and Shmoys [2] proposed a heuristic for p-center problem as follows. Initially, all distances in a graph are sorted in nondecreasing order. An edge with minimum distance was found and removed so that the number of connected graphs after removing all edges with higher distances is fewer than p. Ozsoy and Pinar [5] solve the capacitated case by labeling the quantity of demand to clients and the capacity to facility. Then, Albareda-Sambola et al. [8] improved their result by considering only clients in a given radius and improving the result using Lagrangian relaxation.
Elloumi, Labbe, and Pochet [7] present a polynomial time algorithm for computing lower and upper bounds of the optimal solution. Mirchandani and Francis [7] propose a relaxation algorithm that finds the optimal location of centers by using column generation and set covering approaches. Handler and Mirchandani [7] develop an iterative relaxation algorithm to solve the *p*-center problem for a subset of demand points.
In this paper we use the developed *p*-center problem in [9] that called *Capacitated Second p-Center Problem* (CSpCP). The paper is structured as follows: In Section 2, the proposed model is developed. In Section 3, the Lagrangian relaxation and a basic subgradient algorithm are described. Our computational results are provided in Section 4. Finally, in Section 5, the conclusions are presented and suggestions for future studies are discussed.

2. Problem formulation

Let $N = \{1, \dots, n\}$ be the given set of sites and set of candidate sites for centers is identical to N. We developed model [9] taking into account budget constraint and three types of centers with different capacity. Therefore, the notations which variable, parameter, and indexes are using in the model are given bellow:

N: set of sits
K: type of centers *k*=A, B, C
h_i: denote the demand of site $i \in N$
b_k: the capacity of a center of type *k*
W_i: weight of site *i*
d_{ij}: distance (cost, travel time) from *i* to $j \ \forall\, i, j \in N$
P: number of centers to be located
f_k: facility setup cost of type *k*
B: total budget
Decision variables;

$$x_{jjk} = \begin{cases} 1, & \textit{if a center type k is located at site j,} \\ 0, & \textit{otherwise,} \end{cases}$$

$$x_{ijk} = \begin{cases} 1, & \textit{if the closest center type k to site i is j,} \\ 0, & \textit{otherwise,} \end{cases}$$

Note that x_{jjk}= 1 implies self-allocation of site *j*

13th International Conference of Industrial Engineering**Mazandaran University of Science and Technology**Mizban International Hotel, 22nd and 23rd February 2017

$$v_{ijk} = \begin{cases} 1, if\ the\ second\ closest\ center\ k\ to\ site\ i\ is\ j \\ 0, otherwise, \end{cases}$$

Thus, the model of the problem was developed in the order given below:

$$\text{Min } \boldsymbol{Z} \tag{1}$$

$$\text{subject to } \sum_j \sum_k x_{jjk} = P \tag{2}$$

$$\sum_j \sum_k x_{ijk} = 1 \quad \forall\, i \in N \tag{3}$$

$$\sum_{\substack{j \\ i \neq j}} \sum_k v_{ijk} = 1 \quad \forall\, i \in N \tag{4}$$

$$x_{ijk} + v_{ijk} \le x_{jjk} \quad \forall\, k\ \&\ i,j \in N \tag{5}$$

$$\sum_{\substack{l \in N \\ d_{il} \le d_{ij}}} \sum_k x_{jlk} + \sum_{\substack{l \in N \\ d_{il} > d_{ij}}} \sum_k x_{ilk} \le 1 \quad \forall\, i,j \in N \tag{6}$$

$$x_{jjk} + \sum_{\substack{l \in N \\ d_{il} > d_{ij}}} v_{ilk} \le 1 + x_{ijk} \quad \forall\, k\ \&\ i,j \in N : i \neq j \tag{7}$$

$$\sum_{\substack{l \in N \\ d_{lj} < d_{li}}} h_l\left(v_{lik} + x_{jjk} - 1\right) + \sum_{\substack{l \in N \\ d_{lj} > d_{li}}} h_l(v_{ljk} + x_{iik} - 1) \le b_k x_{jjk} \quad \forall\, k\ \&\ i,j \in N : i \neq j \tag{8}$$

$$z \ge \sum_{l \in N} \sum_k W_i d_{il} v_{ilk} \quad \forall\, i \in N \tag{9}$$

$$\sum_j \sum_k f_k x_{jjk} \le B \tag{10}$$

$$x_{jjk}, x_{ijk}, v_{ijk} \in \{0,1\}$$

The objective function (1) together with constraints (9) are used to get the maximum distance of any site with respect to its second closest center by means of an additional decision variable **z** which is minimized in the objective. Constraints (2) fixes the number of centers to p. Constraints (3) force each site to be allocated to one center. Constraints (4) force each site (center) to be allocated to one second closest center (backup). Constraints (5) ensures that assignments can only be made to open facilities for first and second closest center. (6) and (7) are closest assignment constraints so that, (6) force a variable x_{ijk} to take value one if j is the closest center with respect to site i and (7) force allocation (by means of v-variables) to the second closest center. Constraints (8) are the capacity constraints, each constraint in this family takes care of the capacity of center j if center i fails. Budget constraint (10) states that it should not exceed the amount of total budget.

3. Lagrangian Relaxation approach

There have been a number of successful applications of Lagrangian relaxation to combinatorial optimization problems. The first notable success was the work of Held and Karp on the Traveling Salesman Problem. Fisher highlights a number of the early Lagrangian relaxation applications. The first successful applications of Lagrangian relaxation to location planning models were reported by Cornuejols *et al.* [10].

The main concept of Lagrangian relaxation is to identify the set of complicating constraints of a general integer program and to introduce them into the objective function in a Lagrangian fashion by attaching unit penalties to them so to guide the search toward reducing the amount of constraints violation. This transformation should be constructed to render the new problem easier to solve optimally and hence produce upper/lower bounds. The penalties are adjusted based on the violation and the process is repeated until a suitable stopping criterion (for instance, when the gap between the best lower and upper bound is small, a negligible change in the solution configuration is detected, the maximum computing time is reached, among others) is met[11]. A detailed explanation of Lagrangian relaxation and its implementation can be found in [12, 13].

In this section we relax Constraints (3), (5) and (8) that make the original problem difficult to solve in a reasonable amount of time.
The Lagrangian relaxation problem can be expressed as follows:

$$L(\alpha_{ijk}, \beta_{ij}) = \text{Min}\, Z + \sum_i \alpha_i \left(\sum_j \sum_k x_{ijk} - 1 \right) + \sum_i \sum_j \sum_k \beta_{ijk} \left(x_{ijk} + v_{ijk} - x_{jjk} \right) + \sum_i \sum_j \sum_k \mu_{ijk} \left(\sum_{\substack{l \in N \\ d_{lj} < d_{li}}} h_l (v_{lik} + x_{jjk} - 1) + \sum_{\substack{l \in N \\ d_{lj} > d_{li}}} h_l (v_{ljk} + x_{iik} - 1) - b_k x_{jjk} \right)$$

$s.t \quad (2), (4), (6), (7), (9), (10)$

$x_{jjk}, x_{ijk}, v_{ijk} \in \{0,1\}$

Where α, β and μ stands for the array of Lagrange multipliers

3.1. Subgradient method

Subgradient optimization is a commonly used method to update the Lagrange multipliers. In fact, subgradient optimization can be considered as an adapted version of the gradient method. The algorithm usually stops when a maximum number of iterations is reached. We describe subgradient algorithm as follows:

Algorithm 1: Subgradient method for improving the Lagrangian multipliers

Step 1: Initialize $\boldsymbol{\alpha_i^0 = 0}$, $\boldsymbol{\beta_{ijk}^0 = 0}$, $\boldsymbol{\mu_{ijk}^0 = 0, UB = \overline{Z}}$ $\boldsymbol{and\ LB^* = -\infty}$, $\boldsymbol{\theta = 2}$

Step 2: solve relaxation problem and calculate LB

Step 3: *if LB>LB** then *LB=LB**

Step 4:

$$\gamma(\alpha_i^t) = \sum_j \sum_k x_{ijk} - 1$$

$$\gamma(\beta_{ijk}^t) = x_{ijk} + v_{ijk} - x_{jjk}$$

$$\gamma(\mu_{ijk}^t) = \sum_{\substack{l \in N \\ d_{lj} < d_{li}}} h_l (v_{lik} + x_{jjk} - 1) + \sum_{\substack{l \in N \\ d_{lj} > d_{li}}} h_l (v_{ljk} + x_{iik} - 1) - b_k x_{jjk}$$

$$\alpha_i^t = \max\left(0, \alpha_i^{t-1} + k^t \left(\gamma(\alpha_i^t)\right)\right)$$

$$\beta_{ijk}^t = \max\left(0, \beta_{ijk}^{t-1} + k^t \left(\gamma(\beta_{ijk}^t)\right)\right)$$

$$\mu_{ijk}^t = \max\left(0, \mu_{ijk}^{t-1} + k^t \left(\gamma(\mu_{ijk}^t)\right)\right)$$

While

$$\boldsymbol{stepsize} \qquad k^t = \theta^t \frac{UB - LB^*}{\sum_i \sum_j \sum_k \left(\gamma(\alpha_i^t) + \gamma(\beta_{ijk}^t) + \gamma(\mu_{ijk}^t) \right)^2}$$

Step 5: If there is no improvement of lower bound on *m* consecutive iterations, then $\boldsymbol{\theta = \frac{\theta}{2}}$

Step 6: return to step2

Where $\overline{Z}$ is the known feasible solution value, and θ^t is a scalar satisfying the relation $0 \le \theta^t \le 2$. This scalar is set to 2 at the start of the procedure and is halved whenever the bound does not improve in *m* consecutive iterations.

4. Computational results

In this section, we test the performance of the Lagrangian relaxation. For our computational experiments we used random data sets by uniform distribution for each cases, so that:

$d_{ij} = U(1, 20)$
$h_l = round(U(1, 10))$
$w_i = U(0, 1)$
$b_A = 8$, $b_B = 10$, $b_c = 12$

$f_A = 15$, $f_B = 20$, $f_C = 30$

The proposed model and Lagrangian relaxation algorithm were coded in GAMS 24.1.2 and the programs were run on a Core i7 2 GHz Notebook with 6 GB RAM. The results of the experiments are reported in table 1 for problem. In this table, for different values of ***N*** and ***p***, the best lower bound that obtained from the Lagrangian relaxation and optimal value is shown. Columns Opt and LB respectively represent the solution obtained from solving the main problem and the lower bound is obtained from Lagrangain relaxation. CPU column shows the runtime (in second).

Table 1 - Results of Opt solution & Lagrangian Relaxation

N	*P*	Opt	LB	CPU(in second)	
				opt	LB
40	3	11.53	7.41	807	640
	4	9.45	7.86	562	720
	5	9.15	7.53	379	420
	6	8.26	6.92	150	620
45	3	9.88	6.43	5150	480
	4	7.12	6.40	3155	544
	5	6.72	4.26	1425	797
	6	6.72	4.35	668	925
	7	7.19	4.88	270	507
50	3	11.49	7.23	8897	623
	4	14.30	7.95	3399	669
	5	8.83	6.28	7592	666
	6	7.48	6.22	7019	790
55	4	8.52	4.13	7440	916
	5	6.93	3.98	5122	911
	6	6.91	4.06	2969	2100
60	4	10.46(non optimal)	6.22	40089	1206
	5	12.09(non optimal)	6.12	36510	1260
	8	11.23(non optimal)	6.16	30231	2704

The computational results show that the lower bound obtained from the Lagrangian relaxation is close to the optimal solution and this approach have significantly reduced the runtime.

5. Conclusions and suggestions

In this paper, we have developed a model by considering budget constraints, three types of centers and a solution procedure for the capacitated p-center problem with backup center. A solution algorithm is developed based on the technique of Lagrangian Relaxation. In our algorithm, the Lagrangian relaxation technique is applied for lower bound computation. According to the results, the solution algorithm can generate an approximate solution that is very close to the optimal solution and the lower bound based on Lagrangian relaxation problem and time solution is suitable.

Future research includes stochastic demand for model and using real test instances. To improve its efficiency of lagrangian relaxation, we can consider better choices of the stepsize parameter for the subgradient optimization and also can use the heuristic algorithms.

References

[1] Albareda-Sambola, Maria, Luisa I. Martínez-Merino, and Antonio M. Rodríguez-Chía. "The probabilistic p-center problem." *European Working Group on Location Analysis Meeting 2015*.

[2] Kariv, Oded, and S. Louis Hakimi. "An algorithmic approach to network location problems. II: The p-medians." *SIAM Journal on Applied Mathematics*37.3 (1979): 539-560.

[3] Albareda-Sambola, Maria, Juan A. Díaz, and Elena Fernández. "Lagrangean duals and exact solution to the capacitated p-center problem." *European Journal of Operational Research* 201.1 (2010): 71-81.

[4] Dantrakul, Sittipong, and Chulin Likasiri. "A maximal client coverage algorithm for the p-center problem." *Thai Journal of Mathematics* 10.2 (2012): 423-432.

[5] Hekmatfar, Masoud, and Reza Zanjirani Farahani, eds. *Facility Location: Concepts, Models, Algorithms and Case Studies*. Physica, 2009.

[6] Taghavi, Majid, and Hassan Shavandi. "The P-Center Problem under Uncertainty." *Journal of Industrial and Systems Engineering* 6.1 (2012): 48-57.

[7] Özsoy, F. Aykut, and Mustafa Ç. Pınar. "An exact algorithm for the capacitated vertex p-center problem." *Computers & Operations Research*33.5 (2006): 1420-1436.

[8] Espejo, Inmaculada, Alfredo Marín, and Antonio M. Rodríguez-Chía. "Capacitated p-center problem with failure foresight." *European Journal of Operational Research* 247.1 (2015): 229-244.

[9] Saç, Yiğit. *Implementation of new and classical set covering based algorithms for solving the absolute p-center problem*. Diss. Bilkent University, 2011.
[10] Nezhad, Ali Mohammad, Hasan Manzour, and Said Salhi. "Lagrangian relaxation heuristics for the uncapacitated single-source multi-product facility location problem." *International Journal of Production Economics* 145.2 (2013): 713-723.
[11] Murray, Alan T., and Richard L. Church. "Solving the anti-covering location problem using Lagrangian relaxation." *Computers & operations research* 24.2 (1997): 127-140.
[12] Fisher, Marshall L. "An applications oriented guide to Lagrangian relaxation."*Interfaces* 15.2 (1985): 10-21.
[13] Fisher, Marshall L. "The Lagrangian relaxation method for solving integer programming problems." *Management science* 27.1 (1981): 1-18.