European Journal of Operational Research 255 (2016) 34–42

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Discrete Optimization

p-facility Huff location problem on networks☆

Rafael Blanquero[a,*], Emilio Carrizosa[a], Boglárka G.-Tóth[b], Amaya Nogales-Gómez[c,1]

[a] Departamento de Estadística e Investigación Operativa, Facultad de Matemáticas, Universidad de Sevilla, Spain
[b] Budapest University of Technology and Economics, Hungary
[c] Mathematical and Algorithmic Sciences Lab, Huawei France R&D, Paris, France

ARTICLE INFO

Article history:
Received 21 January 2015
Accepted 19 April 2016
Available online 26 April 2016

Keywords:
Huff location problem
Location on networks
p-facility
Difference of convex
Global optimization

ABSTRACT

The p-facility Huff location problem aims at locating facilities on a competitive environment so as to maximize the market share. While it has been deeply studied in the field of continuous location, in this paper we study the p-facility Huff location problem on networks formulated as a Mixed Integer Nonlinear Programming problem that can be solved by a branch-and-bound algorithm. We propose two approaches for the initialization and division of subproblems, the first one based on the straightforward idea of enumerating every possible combination of p edges of the network as possible locations, and the second one defining sophisticated data structures that exploit the structure of the combinatorial and continuous part of the problem. Bounding rules are designed using DC (difference of convex) and Interval Analysis tools.

In our computational study we compare the two approaches on a battery of 21 networks and show that both of them can handle problems for $p \leq 4$ in reasonable computing time.

1. Introduction

Competitive location models (Eiselt, Laporte, & Thisse, 1993; Plastria, 2001) were originally introduced by Hotelling (1929), considering the location of two competing facilities on a linear market. In the seminal work of Hotelling, users patronize the facility closest to them. In contrast with this all-or-nothing assumption, it was introduced the Huff location model (Huff, 1964), in which the probability that a user patronizes a facility is proportional to its attractiveness and inversely proportional to a power of the distance to it. The Huff location problem has been extensively studied in the field of continuous location (Blanquero & Carrizosa, 2009; Drezner & Drezner, 2004; Fernández, Pelegrín, Plastria, & Tóth, 2007; Huff, 1964; 1966) and successfully applied in the marketing field, in problems such as location of petrol stations, shopping centers or restaurants (Ghosh, McLafferty, & Craig, 1995; Okabe & Kitamura, 1997; Okunuki & Okabe, 2002). The natural extension of this problem to that of locating p-facilities on the plane, has also received certain attention in the literature (Drezner, 1998; Drezner, Drezner, & Salhi, 2002; Redondo, Fernández, García, & Ortigosa, 2009a; 2009b; Tóth, Fernández, Pelegrín, & Plastria, 2009).

Network optimization models (Bertsekas, 1998) are widely used in practice due to their methodological aspects and intuitive formulations. They arise naturally in the context of assignment, flow, transportation or location problems among others. For a comprehensive introduction to location models on networks see Labbé, Peeters, and Thisse (1995).

The combination of the Huff location problem and network optimization has been already addressed in the literature (Berman, Drezner, & Krass, 2011; Blanquero, Carrizosa, Nogales-Gómez, & Plastria, 2014) and applied to market area analysis (Okabe & Kitamura, 1997) and demand estimation (Okabe & Okunuki, 2001). The single-facility case has been solved in Berman et al. (2011) by means of Interval Analysis (IA) bounds, and in Blanquero et al. (2014) using IA and difference of convex (DC) bounds. Different metaheuristics have been proposed for the p-facility case in Roksandić, Carrizosa, Urošević, and Mladenović (2012), but no attempt has been made so far to address the multifacility case with exact methods. This lack of progress in the state of the art is due to the difficulty of the problem, caused by its combinatorial component added to the continuous global optimization: one has to decide which edges are to contain facilities, and, for the

☆ Research partially supported by research grants and projects Hungarian National Research, Development and Innovation Office – NKFIH, OTKA grant PD115554, ICT COST Action TD1207 (EU), MTM2015-65915-R (Ministerio de Economía y Competitividad, Spain), P11-FQM-7603 and FQM-329 (Junta de Andalucía, Spain).

* Corresponding author. Tel.: +34 954557935; fax: +34 954550902.
E-mail addresses: rblanquero@us.es, rafael.blanquero@gmail.com (R. Blanquero), ecarrizosa@us.es (E. Carrizosa), bog@math.bme.hu (B. G.-Tóth), amaya.nogales.gomez@huawei.com (A. Nogales-Gómez).
[1] Most of this work was done when the author was at Departamento de Estadística e Investigación Operativa, Facultad de Matemáticas, Universidad de Sevilla.

http://dx.doi.org/10.1016/j.ejor.2016.04.039
0377-2217/

choice of edges given, the location is to be decided. Thanks to the recent results described in Blanquero, Carrizosa, and G.-Tóth (2015), in which a new data structure is introduced to address multifacility location problems on networks via branch and bound algorithms, we solve in this paper the p-facility Huff location problem on networks, formulated as a Mixed Integer Nonlinear Programming (MINLP) problem.

The remainder of this paper is organized as follows. In Section 2 we set up the notation for networks and introduce the p-facility Huff location problem. In Section 3, a branch-and-bound method with different initialization and branching rules is described. Section 4 is devoted to procedures for calculating lower and upper bounds. Computational results are reported in Section 5, where the p-facility Huff location problem is solved using the different branching and bounding rules for 12 real-life and 9 artificial networks. Finally, Section 6 contains a brief summary, final conclusions and some lines for future research.

2. The model

Let $N = (V, E)$ be a network, with node set V and edge set E. The length of the edge $e \in E$ is denoted by l_e. The distance between two nodes $a_i, a_j \in V$ is calculated as the length of the shortest path (Labbé et al., 1995) from a_i to a_j. For each $e \in E$, with end-nodes a_i, a_j, we identify each $x \in [0, l_e]$ with the point in the edge e at distance x from a_i and $l_e - x$ from a_j. In this way, we obtain that, for any vertex $a_k \in V$ and $x \in e$, the distance $d(x, a_k)$ from x to a_k, as a function of x, is a concave piecewise linear function, given by $d(x, a_k) = \min\{x + d(a_i, a_k), (l_e - x) + d(a_j, a_k)\}$.

In the p-facility Huff location model, the finite set V of vertices of the network represents users, asking for a certain service. Each user $a \in V$ has demand $\omega_a \geq 0$, that is patronized by different existing facilities, located at points $y_1, \ldots, y_r$ on the network. The demand captured by facility at y_i from user a is assumed to be inversely proportional to a positive nondecreasing function of the distance $d(a, y_i)$, namely, $\alpha_{ai}/(d(a, y_i))^2$ is used as the utility or attraction function of y_i, where $\alpha_{ai} > 0$ denotes the attraction that user a feels towards the facility at y_i. Therefore, the demand captured by the facility at y_i from the user at a is given by

$$\omega_a \frac{\alpha_{ai}/(d(a, y_i))^2}{\sum_{j=1}^{r} \alpha_{aj}/(d(a, y_j))^2}. \tag{1}$$

A new firm is entering the market, by locating p new facilities at some points $x_1, \ldots, x_p$ on the network. For simplicity, all new facilities are assumed to have the same attractiveness $\alpha_a > 0$, which is fixed. The new facilities perturb how the market is shared, since the new facilities will capture part of the demand from $a \in V$,

$$\omega_a \frac{\sum_{j=1}^{p} \alpha_a/(d(a, x_j))^2}{\sum_{j=1}^{p} \alpha_a/(d(a, x_j))^2 + \sum_{j=1}^{r} \alpha_{aj}/(d(a, y_j))^2}. \tag{2}$$

Our goal is the maximization of the market share of the entering firm. Thus, the problem we need to solve can be formulated as

$$\max_{\substack{x_1 \in [0, l_{e_1}], \ldots, x_p \in [0, l_{e_p}] \\ e_1, \ldots, e_p \in E}} \times \sum_{a \in V} \omega_a \frac{\sum_{j=1}^{p} \alpha_a/(d(a, x_j))^2}{\sum_{j=1}^{p} \alpha_a/(d(a, x_j))^2 + \sum_{j=1}^{r} \alpha_{aj}/(d(a, y_j))^2}. \tag{3}$$

In order to simplify the previous expression, the following positive constant is considered for each $a \in V$:

$$\beta_a = \sum_{j=1}^{r} \frac{\alpha_{aj}/\alpha_a}{(d(a, y_j))^2}. \tag{4}$$

Problem (3) can be rewritten then as the following MINLP:

$$\max_{\substack{x_1 \in [0, l_{e_1}], \ldots, x_p \in [0, l_{e_p}] \\ e_1, \ldots, e_p \in E}} F(x_1, \ldots, x_p) \tag{5}$$

where F is defined as

$$F(x_1, \ldots, x_p) = \sum_{a \in V} \omega_a \frac{1}{1 + \frac{\beta_a}{\sum_{j=1}^{p} \frac{1}{(d(a, x_j))^2}}}. \tag{6}$$

The MINLP problem (5) is formed by a combinatorial and a continuous part. First, we need to solve the combinatorial problem of choosing a set of p edges to locate the facilities, and then solve a continuous location problem on the edges.

3. The methodology

The natural way to solve the MINLP formulation of the p-facility Huff location problem is to use a branch-and-bound method. We differentiate two main phases: the initialization phase and the branch-and-bound phase. In the initialization phase the initial exploration tree is prepared. In the branch-and-bound phase, an element of the list is selected iteratively (until the termination rule is fulfilled) according to a selection criterion, and then is divided into new elements that are included into the list if they cannot be eliminated by their bounds. In this phase, division, bounding, selection, elimination and termination rules are required.

In this paper we propose different approaches for the initialization phase, division and bounding rules. As selection, elimination and termination rules, we always apply the usual ones from the literature (Berman et al., 2011): the element to be evaluated is selected as the one with the largest upper bound, elements whose upper bound are lower than the current lower bound are eliminated, and the optimization is terminated when the relative error between the largest upper bound and the current lower bound is less than a fixed tolerance. This section is aimed at describing two types of initialization and division rules. Bounding rules will be discussed in Section 4.

The methodology proposed in Blanquero et al. (2014), where the single-facility problem is tackled, has in common with the methodology herein considered the use of a branch-and-bound algorithm with, essentially, the same upper bounds. However, the difficulty introduced by the combinatorial part of the problem leads us to use sophisticated data structures in the branch-and-bound recently introduced in Blanquero et al. (2015), so that the election of the p edges can be done in an optimal way during the algorithm running. The initialization and the division phases of the algorithm are deeply affected by the use of these structures.

3.1. Total enumeration

The straightforward way of solving Problem (5) is to separate the combinatorial and the continuous part of the problem: we first fix a set of p edges to locate the facilities, and then solve a continuous location problem on the edges. This means the branch-and-bound approach starts with a partition of the search space formed by the cartesian product of p-uples. The p-uples are formed by every possible combination of p edges, taking into account that several facilities can be located at the same edge, i.e., repetitions of the same edge are allowed in the elements of the partition. But obviously, permutations of the p-uples are not taken into account.

During the algorithm running, the edges forming the initial elements of the partition are going to be divided into small pieces (segments of edge), which will be referred to as *subedges* throughout the paper.

We denote by $\underline{s} = (s_1, \ldots, s_k)$ an element of the partition, where each component s_i is a (sub)edge that has a multiplicity $m(s_i)$, i.e., the number of facilities located at s_i is $m(s_i)$. Hence, $m(s_1) +$