Discrete Applied Mathematics 161 (2013) 2624–2646

Contents lists available at ScienceDirect

Discrete Applied Mathematics

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

A specialized branch & bound & cut for Single-Allocation Ordered Median Hub Location problems

J. Puerto [a,*], A.B. Ramos [a], A.M. Rodríguez-Chía [b]

[a] Facultad de Matemáticas, Universidad de Sevilla, Spain

[b] Facultad de Ciencias, Universidad de Cádiz, Spain

ARTICLE INFO

Article history:
Received 18 March 2011
Received in revised form 28 May 2013
Accepted 31 May 2013
Available online 27 June 2013

Keywords:
Hub location
Discrete ordered median location

ABSTRACT

The Single-Allocation Ordered Median Hub Location problem is a recent hub model introduced by Puerto et al. (2011) [32] that provides a unifying analysis of the class of hub location models. Indeed, considering ordered objective functions in hub location models is a powerful tool in modeling classic and alternative location paradigms, that can be applied with success to a large variety of problems providing new distribution patterns induced by the different users' roles within the supply chain network. In this paper, we present a new formulation for the Single-Allocation Ordered Median Hub Location problem and a branch-and-bound-and-cut (B&B&Cut) based algorithm to solve optimally this model. A simple illustrative example is discussed to demonstrate the technique, and then a battery of test problems with data taken from the AP library are solved. The paper concludes that the proposed B&B&Cut approach performs well for small to medium sized problems.

1. Introduction

The importance of hub location models in the area of Supply Chain networks is shown by the number of references published in the last years using different criteria to locate hubs, as for instance, minimizing the overall transportation cost (sum) (see [4,6,10,11,13,15,21–26,34,38], among others), the largest transportation cost or the coverage cost [5,8,18–20,29,36,37], as well as the surveys [2,7,9] and the references therein.

Recently, the Single Ordered Median hub location problem, introduced by Puerto et al. [32], has been recognized as a powerful tool from a modeling point of view. The reason being that this model allows to distinguish the roles played by the different parties in a hub-type supply chain network inducing new type of distribution patterns; see [14,16,17].

This formulation incorporates flexibility through rank dependent compensation factors, and it allows one to model that the driving force in the supply chain is shared by the suppliers and the distribution system. Suppliers support the transportation costs from the origin sites to the first hub and the distribution system supports the transportation cost from the first hub to the destination sites. Actually, any origin–destination delivery path is composed of, at most, two components: (1) the subpath that goes from an origin site to the first access point (first hub) to the distribution system, and (2) the subpath that links first hubs to final destinations. In addition, this last component is itself also divided, at most, in two parts: (2.1) the inter hubs links and (2.2) the link from the last hub to the final destination.

Each one of the components of any origin–destination delivery path described above gives rise to a cost that is weighted by different compensation factors depending on the role of the party that supports the cost. The costs associated with the

* Corresponding author. Tel.: +34 954557940; fax: +34 954622800.
E-mail addresses: puerto@us.es, justo.puerto@gmail.com (J. Puerto), anabelenramos@us.es (A.B. Ramos), antonio.rodriguezchia@uca.es (A.M. Rodríguez-Chía).

0166-218X/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.dam.2013.05.035

inter-hub links have a fixed discount $0 < \mu < 1$ and the links between last-hub and the destination sites have another discount factor $0 < \delta < 1$. We assume that the commodity of each origin is transported to a single (unique) first hub. In addition, deliveries from the origin sites to the distribution system are scaled by rank dependent weights. This adds a "sorting"-problem to the underlying hub location problem, making its formulation and solution much more challenging. Hence, the objective is to minimize the total transportation cost of the flows between each origin–destination pair, routed through at most two hubs, once we have applied rank dependent compensation factors on the transportation costs of the origin–first hub links, and fixed scaling factors for the interhub and hub–final destination transportation costs.

The reader may note that apart from the classical median and center approaches to hub location problems, our analysis opens very interesting new perspectives to be considered. Some of these new paradigms are the minimization of the k-largest costs (k-centrum), the most centered costs (trimmed mean), the most extreme costs (anti-trimmed mean) and the range of costs, as well as many other models without a specific name that may fit better to some real situations. Moreover, this approach allows us to cope with actual requirements from nowadays logistics [30,35,39]. This methodology, as mentioned in [16], introduces, as part of the model, the point of view of the member of the logistics network that is the driving force of the planning process. Obviously, this gives rise to different problems that need new resolution methods because for them there are no generic or "ad hoc" algorithmic approaches available.

In this paper, we analyze in depth the above discussed model trying to obtain a better knowledge and alternative ways to solve it. More precisely, we will provide, first, a new formulation in the spirit of [28] where the number of variables has been considerably reduced with respect to the one in [32]; second, several procedures for fixing variables (based on non straightforward adaptations of the rationale of [32]), and new methods for computing lower and upper bounds, both, on some variables and on the objective function value; and third, several families of valid inequalities to be incorporated in the resolution process. From this analysis, we will develop a branch-and-bound-and-cut algorithm that allows us to solve larger instances than the ones solved in [32].

The rest of the paper is organized as follows: In Section 2 we recall the covering variable formulation of the Single Allocation Hub Location problem that provided the best computational results in [32, Section 3]. Next, we give an alternative formulation for this problem, proving the equivalence between them. Section 3 presents a new lower bound on the objective value based on a related median hub problem, some variable fixing procedures and several families of valid inequalities. In Section 4, two additional combinatorial lower bounds and four different ways to compute upper bounds are presented; this section ends with the description of the branching scheme. Section 5 reports the computational analysis of the methodology developed in previous sections. Along the paper some examples have been included to illustrate the different results and models. The paper ends with some conclusions.

2. Model and formulation

Let A denote a given set of N client sites and identify these with integers $1, \ldots, N$. Each site is collecting or gathering some commodity that must be sent to the remaining sites. Let $w_{jm} \geq 0$ be the amount of commodity to be supplied from the jth to the mth site for all $j, m \in \{1, \ldots, N\}$ and let $W_j = \sum_{m=1}^{N} w_{jm}$. In the following, we assume without loss of generality that the set of candidate sites for establishing hubs is identical to the set of sites A. Let $c_{jm} \geq 0$ denote the unit cost of sending commodity from site j to site m (not necessarily satisfying the triangular inequality). We assume that $c_{jj} = 0, \forall j = 1, \ldots, N$ (free self-service). Let $p \leq N$ be the number of hubs to be located and $X \subset A$ with $|X| = p$ denote a feasible set of candidate sites. A solution for the problem is a feasible set of candidate sites X, plus a set of paths connecting pairs (flow patterns) of sites j, m for all $j, m \in \{1, \ldots, N\}$ in such a way that each path traverses at least one and no more than two hubs from X. To be more precise, (i) if the origin site j and the destination site m are not hubs, the flow must go through one or two intermediate hubs; (ii) if either the origin or the destination sites are hubs, the flow between them can be either directly sent or sent through an additional hub; and (iii) if both origin and destination sites are hubs, the flow must go directly from the origin to the destination.

In addition, this model compensates origin site–first hub transportation costs by using parameters $\lambda = (\lambda_1, \ldots, \lambda_N)$. These scaling factors will be assigned to the origins depending on the order of the sequence of transportation costs of the commodity with the same origin to the first hub (see [3,27,28,31] for different ordered median location models). Indeed, if a solution sends the commodity from the origin site j via a first hub k and this delivery cost, namely $W_j c_{jk}$, were ranked in the ith position among these type of costs then this term would be scaled by λ_i, i.e. the corresponding objective function component would be $\lambda_i W_j c_{jk}$. In addition, we also consider a compensation parameter $0 < \mu < 1$ for the deliveries between hubs and another parameter $0 < \delta < 1, \mu < \delta$, for the deliveries between hubs and final destination sites. These parameters may imply that, at times, using a second hub results in a cheaper connection than going directly from the first hub to the final destination.

Observe that depending on the choices of the λ-vector we can obtain different criteria to account for the costs from the origins to their first hubs in the objective function (for instance, if $\lambda = (0, \ldots, 0, 1, \overset{k}{\ldots}, 1)$ were considered, the first component of the objective function would be the sum of the k-largest costs—k-centrum) usually providing different solutions or different allocation patterns for problems with different λ, even though the optimal solution gets the same set of open hubs, see [32] for further details. In this section we recall the formulation of the *Single Allocation Ordered Median Hub location problem* based on covering variables (see [32] for further details). In order to do that, let us denote by $\hat{c}_{jk}$ the cost of the overall flow sent from the origin site j if it were delivered via the first hub k, i.e. $\hat{c}_{jk} := c_{jk} W_j,\ j, k = 1, \ldots, N$.

Next, let G be the number of different elements of the cost sequence $(\hat{c}_{jk})$ for any $j, k = 1, \ldots, N$. Then, we can sort the different values of this sequence in increasing order:

$$\hat{c}_{(1)} := 0 < \hat{c}_{(2)} < \cdots < \hat{c}_{(G)} := \max_{1 \le j,k \le N} \{\hat{c}_{jk}\}.$$

Given a feasible solution, we use this ordering to perform the sorting process of the allocation costs with the following covering variables ($i = 1, \ldots, N$ and $h = 1, \ldots, G$):

$$\bar{u}_{ih} := \begin{cases} 1, & \text{if the } i\text{-th smallest allocation cost is at least } \hat{c}_{(h)}, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

Clearly, the i-th smallest allocation cost is equal to $\hat{c}_{(h)}$ if and only if $\bar{u}_{ih} = 1$ and $\bar{u}_{i,h+1} = 0$.

In addition, we define the following set of variables:

$$r_{jk} = \begin{cases} 1, & \text{if the commodity sent from origin site } j \text{ goes first to the hub } k, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

$$x_{k\ell m} = \text{flow that goes through a first hub } k \text{ and a second hub } \ell \text{ with destination } m,$$

with $j, k, \ell, m = 1, \ldots, N$. Since we assume free self service and non-negative costs, the above definition implies that site k is opened as a hub if the corresponding variable $r_{kk} = 1$. Hence, the formulation of the model is:

$$\min \sum_{i=1}^{N} \sum_{h=2}^{G} \lambda_i (\hat{c}_{(h)} - \hat{c}_{(h-1)}) \bar{u}_{ih} + \sum_{k=1}^{N} \sum_{\ell=1}^{N} \sum_{m=1}^{N} x_{k\ell m} (\mu c_{k\ell} + \delta c_{\ell m}) \tag{3}$$

$$\text{s.t.} \sum_{k=1}^{N} r_{jk} = 1, \quad \forall j = 1, \ldots, N \tag{4}$$

$$\sum_{j=1}^{N} r_{jk} \le N r_{kk}, \quad \forall k = 1, \ldots, N \tag{5}$$

$$\sum_{\ell=1}^{N} x_{k\ell m} = \sum_{j=1}^{N} r_{jk} w_{jm}, \quad \forall k, m = 1, \ldots, N \tag{6}$$

$$x_{k\ell m} \le (1 - r_{mm}) \sum_{j=1}^{N} w_{jm} \quad \forall k, \ell, m = 1, \ldots, N, \ell \ne m \tag{7}$$

$$\sum_{\ell=1}^{N} \sum_{m=1}^{N} x_{k\ell m} \le r_{kk} \sum_{j=1}^{N} W_j, \quad \forall k = 1, \ldots, N \tag{8}$$

$$\sum_{k=1}^{N} \sum_{m=1}^{N} x_{k\ell m} \le r_{\ell\ell} \sum_{j=1}^{N} W_j, \quad \forall \ell = 1, \ldots, N \tag{9}$$

$$\sum_{k=1}^{N} r_{kk} = p \tag{10}$$

$$\sum_{i=1}^{N} \bar{u}_{ih} = \sum_{j=1}^{N} \sum_{\substack{k=1 \\ \hat{c}_{jk} \ge \hat{c}_{(h)}}}^{N} r_{jk}, \quad \forall h = 1, \ldots, G \tag{11}$$

$$\bar{u}_{ih} \ge \bar{u}_{i-1h}, \quad \forall i = 2, \ldots, N, h = 1, \ldots, G \tag{12}$$

$$\bar{u}_{ih}, r_{jk} \in \{0, 1\}, \qquad x_{k\ell m} \ge 0, \quad \forall i, j, k, \ell, m = 1, \ldots, N, h = 1, \ldots, G. \tag{13}$$

The objective function (3) accounts for the weighted sum of the three components of the shipping cost, namely origin site to first hub, between hubs connections and last hub–final destination site. The first block of shipping costs is accounted after the compensation process using the lambda parameters, i.e. $\sum_{i=1}^{N} \sum_{h=1}^{G} \lambda_i \cdot (\hat{c}_{(h)} - \hat{c}_{(h-1)}) \cdot \bar{u}_{ih}$. In addition, the second and third blocks of delivery costs, scaled with the μ and δ parameters, respectively, can be stated as: $\sum_{k=1}^{N} \sum_{\ell=1}^{N} \sum_{m=1}^{N} x_{k\ell m} (\mu c_{k\ell} + \delta c_{\ell m})$.

Constraints (4) ensure that the flow from the origin site j is associated with a unique first hub. Constraints (5) ensure that one origin may be allocated to a specific first hub only if it is open. Observe that this family of constraints can be presented in a disaggregated form, i.e. $r_{jk} \le r_{kk}, \forall j, k = 1, \ldots, N$. However, the computational experience in [32] showed that using the disaggregated form provided worse computational running times than using the original constraints (5). Constraints (6) are flow conservation constraints and they ensure that the flow that enters any hub k with final destination m is the same that the flow that leaves hub k with destination m. Constraints (7) ensure that if the final destination site is a hub, then the flow goes at most through one additional hub. Note that the family of constraints (7) are redundant whenever the cost structure satisfies the triangular inequality, however they are useful in reducing solution times (see [32] for further details). Constraints (8) and (9) establish that the intermediate nodes in any origin–destination path should be open hubs. Constraint (10) fixes the number of hubs to be located. Constraints (11) state that the number of allocations with a cost at least $\hat{c}_{(h)}$ must be equal to the number of sites that support shipping costs to the first hub greater than or equal to $\hat{c}_{(h)}$. Finally, constraints (12) are a group of sorting conditions on the $\bar{u}_{ih}$-variables.

Example 2.1. To illustrate the formulation (3)–(13) we consider the following data. Let $A = \{1, \dots, 10\}$ be a set of sites and assume that we are interested in locating $p = 2$ hubs. Let the cost and flow matrices be as follows:

$$C = \begin{pmatrix} 0 & 14 & 15 & 16 & 15 & 9 & 1 & 5 & 18 & 11 \\ 5 & 0 & 7 & 2 & 19 & 16 & 20 & 1 & 2 & 17 \\ 16 & 5 & 0 & 7 & 1 & 19 & 20 & 8 & 12 & 20 \\ 12 & 1 & 10 & 0 & 13 & 1 & 15 & 16 & 4 & 19 \\ 1 & 9 & 9 & 15 & 0 & 2 & 8 & 13 & 20 & 9 \\ 8 & 10 & 16 & 8 & 4 & 0 & 2 & 2 & 2 & 13 \\ 10 & 15 & 3 & 15 & 12 & 11 & 0 & 6 & 1 & 9 \\ 6 & 5 & 13 & 16 & 6 & 12 & 17 & 0 & 12 & 12 \\ 12 & 18 & 8 & 10 & 9 & 12 & 14 & 5 & 0 & 3 \\ 8 & 19 & 17 & 3 & 14 & 16 & 20 & 19 & 8 & 0 \end{pmatrix},$$

$$W = \begin{pmatrix} 0 & 15 & 2 & 8 & 11 & 2 & 13 & 20 & 6 & 14 \\ 19 & 0 & 1 & 16 & 20 & 7 & 16 & 16 & 9 & 2 \\ 3 & 9 & 0 & 3 & 11 & 16 & 17 & 6 & 3 & 2 \\ 7 & 2 & 5 & 0 & 14 & 5 & 13 & 10 & 2 & 9 \\ 15 & 4 & 20 & 4 & 0 & 1 & 13 & 17 & 11 & 15 \\ 12 & 4 & 7 & 11 & 18 & 0 & 20 & 10 & 15 & 12 \\ 14 & 2 & 11 & 18 & 2 & 15 & 0 & 14 & 8 & 17 \\ 7 & 15 & 17 & 20 & 7 & 9 & 8 & 0 & 9 & 12 \\ 11 & 7 & 6 & 4 & 18 & 14 & 12 & 16 & 0 & 7 \\ 18 & 18 & 19 & 20 & 15 & 5 & 7 & 14 & 7 & 0 \end{pmatrix}.$$

Therefore, $\hat{c}_{(\cdot)}$, the sorted vector of $\hat{c}$, is in our case

$\hat{c}_{(\cdot)} = [0, 67, 70, 91, 100, 101, 106, 200, 212, 218, 268, 285, 303, 350, 369, 436, 455, 475, 490, 520, 530, 560, 606, 624, 670, 742, 760, 800, 804, 819, 840, 855, 871, 872, 900, 909, 950, 984, 1001, 1005, 1010, 1072, 1090, 1111, 1120, 1140, 1212, 1248, 1273, 1274, 1300, 1330, 1352, 1365, 1400, 1417, 1456, 1500, 1515, 1638, 1664, 1696, 1710, 1722, 1744, 1768, 1802, 1968, 2000, 2014, 2091, 2120, 2337, 2460]$.

Hence, $G = 74$. Let $\lambda = (0, 0, 1, 1, 0, 0, 1, 1, 1, 0)$, $\mu = 0.7$ and $\delta = 0.9$. Running XPRESS in this example the optimal solution opens hubs 4 and 6. The allocation of origin sites to first hub is given by the following values of the r-variables (see Fig. 1):

$$r_{1,6} = r_{2,4} = r_{3,4} = r_{4,4} = r_{5,6} = r_{6,6} = r_{7,6} = r_{8,4} = r_{9,4} = r_{10,4} = 1.$$

Analogously, the allocation of first hubs to final destinations are given by the values of the non null x variables. Thus, the flows considering as first hubs 4 and 6 are (see Fig. 2 for a graphical representation of the delivery paths):

$x_{4,4,2} = 51, x_{4,4,3} = 48, x_{4,4,4} = 63, x_{4,6,1} = 65, x_{4,6,5} = 85, x_{4,6,6} = 56, x_{4,6,7} = 73, x_{4,6,8} = 62, x_{4,6,9} = 30, x_{4,6,10} = 32; x_{6,4,2} = 25, x_{6,4,4} = 41, x_{6,6,1} = 41, x_{6,6,3} = 40, x_{6,6,5} = 31, x_{6,6,6} = 18, x_{6,6,7} = 46, x_{6,6,8} = 61, x_{6,6,9} = 40, x_{6,6,10} = 58$.

Moreover, the covering variables $\bar{u}_{ih}$ are given below. Due to their structure, we only report for each i the last 1 and the first zero values since they characterize the remaining values:

$$\begin{array}{lll} i = 1 \mapsto \bar{u}_{1,1} = 1, \bar{u}_{1,2} = 0 & i = 2 \mapsto \bar{u}_{2,1} = 1, \bar{u}_{2,2} = 0 & i = 3 \mapsto \bar{u}_{3,8} = 1, \bar{u}_{3,9} = 0 \\ i = 4 \mapsto \bar{u}_{4,9} = 1, \bar{u}_{4,10} = 0 & i = 5 \mapsto \bar{u}_{5,15} = 1, \bar{u}_{5,16} = 0 & i = 6 \mapsto \bar{u}_{6,19} = 1, \bar{u}_{6,20} = 0 \\ i = 7 \mapsto \bar{u}_{7,30} = 1, \bar{u}_{7,31} = 0 & i = 8 \mapsto \bar{u}_{8,37} = 1, \bar{u}_{8,38} = 0 & i = 9 \mapsto \bar{u}_{9,44} = 1, \bar{u}_{9,45} = 0 \\ i = 10 \mapsto \bar{u}_{10,61} = 1, \bar{u}_{10,62} = 0. & & \end{array}$$

Hence, the overall cost of this solution is

$$\sum_{i=1}^{N} \sum_{h=2}^{G} \lambda_i (\hat{c}_{(h)} - \hat{c}_{(h-1)}) \bar{u}_{ih} + \sum_{k=1}^{N} \sum_{\ell=1}^{N} \sum_{m=1}^{N} x_{k\ell m} (\mu c_{k\ell} + \delta c_{\ell m}) = 3292 + 4523.5 = 7815.5.$$

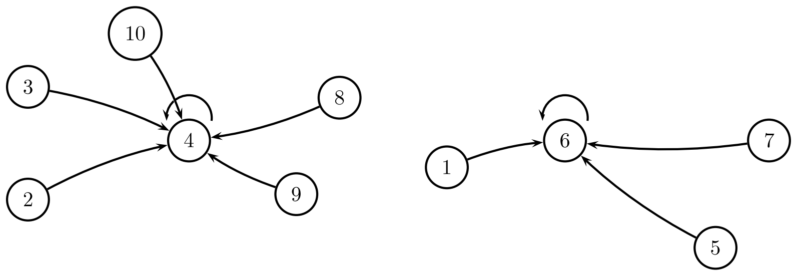

Fig. 1. Allocations of origin sites to hubs in Example 2.1.

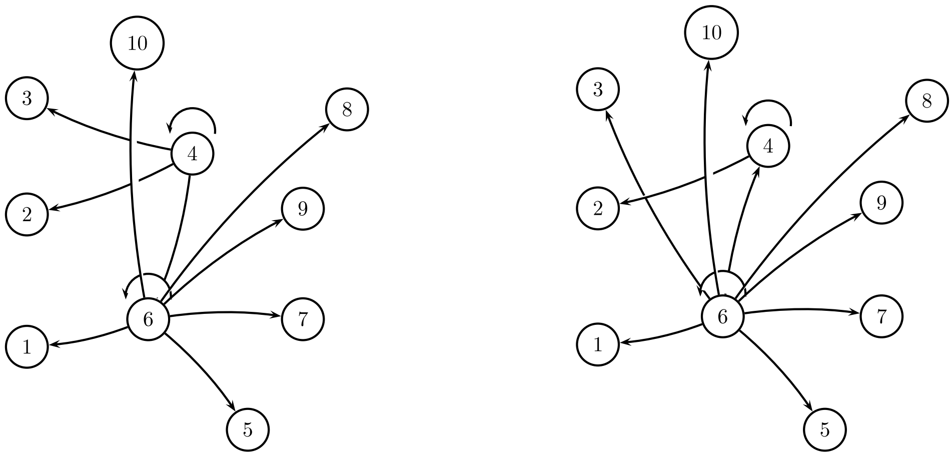

Fig. 2. Allocations from hubs 4 (left) and 6 (right) as first hubs in Example 2.1.

2.1. Improved reformulations

The rationale of the above formulation can be further strengthen for important particular cases of the discrete ordered median hub location problem. In the following, we show this reformulation that is based on taking advantage of sequences of repetitions in the λ-vector (see [28] for a similar reformulation applied to regular discrete location).

We observe that for λ-vectors with sequences of repetitions, as for instance the center, k-centrum, trimmed means or median among others, many variables used in formulation (3)–(13) are not necessary and some others can be glued together.

Since we have assumed free self-service, we have that the p smallest transportation cost from the origin to the first hubs are 0, i.e. the first p components of the λ-vector are multiplied by 0. Therefore, in order to simplify the problem we can remove these p first components. Let $\tilde{\lambda} = (\tilde{\lambda}_1, \ldots, \tilde{\lambda}_{N-p}) := (\lambda_{p+1}, \ldots, \lambda_N)$. Then, let I be the number of blocks of consecutive equal non-null elements in $\tilde{\lambda}$ and define the vectors:

1. $\gamma = (\gamma_1, \ldots, \gamma_I)$, being $\gamma_i, i = 1, \ldots, I$ the value of the elements in the i-th block of repeated elements in $\tilde{\lambda}$.
2. $\alpha = (\alpha_1, \ldots, \alpha_I, \alpha_{I+1})$, being α_i with $i = 1, \ldots, I$, the number of zero entries between the $(i-1)$-th and i-th blocks of positive elements in $\tilde{\lambda}$ and α_{I+1} the number of zeros, if any, after the I-th block of non-null elements in $\bar{\lambda}$. For notation purposes we define $\alpha_0 = 0$.
3. $\beta = (\beta_1, \ldots, \beta_I)$, being $\beta_i, i = 1, \ldots, I$ the number of elements in the i-th block of non-null elements in $\tilde{\lambda}$. For the sake of compactness, let $\beta_0 = \beta_{I+1} = 0$.

With the above definitions we are ready to define the new set of variables of our reformulation. For $i = 1, \ldots, I$ and $h = 1, \ldots, G$ we set

$$u_{ih} = \begin{cases} 1 & \text{if the } \left(\sum_{j=1}^{i} \alpha_j + \sum_{j=1}^{i-1} \beta_j + 1\right)\text{-th allocation cost is at least } \hat{c}_{(h)}, \\ 0 & \text{otherwise.} \end{cases} \tag{14}$$

$$v_{ih} = \text{number of allocations in the } i\text{-th block } \left(\text{between positions } \sum_{j=1}^{i} \alpha_j + \sum_{j=1}^{i-1} \beta_j + 1 \text{ and } \sum_{j=1}^{i} (\alpha_j + \beta_j)\right) \text{ that are at least } \hat{c}_{(h)}. \tag{15}$$

Table 1
Comparison of formulations "covering" and "improved" in Examples 2.1 and 2.2.

	Time	Nodes	GAP	Variables		
				Cont.	Binary	Integer
Covering	1.911	107	38.52	1000	840	0
Improved	0.865	81	47.89	1000	248	148

Using the above, the reformulation of the problem is as follows:

$$\min \sum_{i=1}^{I}\sum_{h=2}^{G}\gamma_i(\hat{c}_{(h)}-\hat{c}_{(h-1)})v_{ih}+\sum_{k=1}^{N}\sum_{\ell=1}^{N}\sum_{m=1}^{N}x_{k\ell m}(\mu c_{k\ell}+\delta c_{\ell m}) \tag{16}$$

s.t. Constraints: (4)–(10), (17)

$$\sum_{i=1}^{I}\alpha_i u_{ih}+\sum_{i=1}^{I}v_{ih}+\alpha_{I+1}\geq\sum_{j=1}^{N}\sum_{\substack{k=1\\ \hat{c}_{jk}\geq\hat{c}_{(h)}}}^{N} r_{jk},\quad \forall h=1,\ldots,G \tag{18}$$

$$u_{ih}\geq u_{i-1h},\quad \forall i=2,\ldots,I, h=1,\ldots,G \tag{19}$$

$$\beta_{i-1}u_{ih}\geq v_{i-1,h},\quad \forall i=2,\ldots,I, h=1,\ldots,G \tag{20}$$

$$v_{ih}\geq\beta_i u_{ih},\quad \forall i=1,\ldots,I, h=1,\ldots,G \tag{21}$$

$$u_{ih}\in\{0,1\}, v_{ih}\in\mathbb{Z}\cap[0,\beta_i],\quad \forall i=1\ldots,I, h=1,\ldots,G \tag{22}$$

$$r_{jk}\in\{0,1\},\qquad x_{k\ell m}\geq 0,\quad \forall j,k,\ell,m=1,\ldots,N. \tag{23}$$

Clearly, the objective function (16) is a reformulation of (3) taking advantage of the u-, v-variables and the vector γ. Constraints (18) ensure that the number of sites that support a shipping cost to the first hub greater than or equal to $\hat{c}_{(h)}$ is either equal to the number of allocations with a cost at least $\hat{c}_{(h)}$ whenever $v_{Ih}>0$ or less than or equal to α_{I+1} otherwise. Constraints (19) are sorting constraints on the u-variables similar to constraints (12). Constraints (20)–(21) provide upper and lower bounds on the v-variables depending on the values of u-variables.

We observe that for those cases where $\beta_i=1$, then $v_{ih}=u_{ih}$ and this set of constraints would be added to reinforce the formulation. Moreover, the main difference of the above formulation and (3)–(13) is that all $\bar{u}_{ih}$ variables associated with blocks of zero λ-values are removed and those associated with each block of non-null λ values are replaced by $2\times G$ variables. Therefore, overall we reduce the number of variables by $(N-2I)\times G$.

Example 2.2. To illustrate the above reformulation, we consider the data in Example 2.1. Clearly, we have that $\tilde{\lambda}=(1,1,0,0,1,1,1,0)$. The number of different blocks of non-null elements in $\tilde{\lambda}$ is $I=2$. The repeated value in the first block is $\gamma_1=1$ and in the second block $\gamma_2=1$. Thus, $\gamma=(1,1)$. The number of repetitions in each block is given by $\beta=(2,3)$ being $\beta_0=\beta_3=0$. This means that the first and second blocks have two and three elements, respectively. Finally, the number of zeros preceding each block of non null elements is given by $\alpha=(0,2,1)$.

The open hubs and the allocations of sites to open hubs in the optimal solution for this reformulation coincide with those described in Example 2.1 and therefore are omitted. Table 1 shows a comparison of both formulations for Example 2.1. Columns Time, Nodes and GAP report, respectively, the CPU time, the number of the nodes of the Branch and Bound tree and the GAP at the root node when solving both formulations using XPRESS. In addition, the columns Cont., Binary, Integer give, respectively, the number of continuous, binary and integer variables of both formulations. We highlight that improved formulation provides slightly worse GAP although its number of variables is considerably less than covering formulation.

Next proposition states the equivalence between formulation (3)–(13) and (16)–(23).

Proposition 2.1. *Let $(r,x,\bar{u})$ be a solution of (3)–(13) then there exists a solution (r,x,u,v) for (16)–(23) such that their objective values are equal. Conversely, if (r,x,u,v) is a feasible solution for (16)–(23) then there exists a solution $(r,x,\bar{u})$ for (3)–(13) having the same objective value.*

Proof. Let $(r,x,\bar{u})$ be a feasible solution of (3)–(13). Then we define $u_{i_0,h}$ and $v_{i_0,h}$ for $i_0=1,\ldots,I$, $h=1,\ldots,G$, as follows:

$$\begin{cases} u_{i_0,h}=1, & \text{if } \sum_{i=1}^{N}\bar{u}_{ih}\geq\beta_{i_0}+\sum_{i=i_0+1}^{I+1}(\alpha_i+\beta_i)\\ u_{i_0,h}=0, & \text{otherwise.}\end{cases}$$

$$\begin{cases} v_{i_0,h} = \beta_{i_0}, & \text{if } \sum_{i=1}^{N} \bar{u}_{ih} \geq \beta_{i_0} + \sum_{i=i_0+1}^{I+1} (\alpha_i + \beta_i) \\ v_{i_0,h} = \sum_{i=1}^{M} \bar{u}_{ih} - \sum_{i=i_0+1}^{I+1} (\alpha_i + \beta_i), & \text{if } \sum_{i=i_0+1}^{I+1} (\alpha_i + \beta_i) < \sum_{i=1}^{N} \bar{u}_{ih} < \beta i_0 + \sum_{i_0+1}^{I+1} (\alpha_i + \beta_i) \\ v_{i_0,h} = 0, & \text{otherwise.} \end{cases}$$

These variables make constraints (18) active for any $h = 1, \dots, G$ such that $\sum_{i=1}^{N} \bar{u}_{ih} \geq \alpha_{I+1}$; otherwise this inequality is strict. Moreover, constraints (19)–(21) also hold. Therefore (r, x, u_0, v_0) is a feasible solution of (16)–(23).

Conversely, let (r, x, u, v) be a feasible solution of (16)–(23). Then, we define $\bar{u}_{i_0,h}$ for $i_0 = 1, \dots, N$, $h = 1, \dots, G$ as follows. For $N - i_0 \geq \alpha_{I+1}$:

$$\begin{cases} \bar{u}_{i_0,h} = 1, & \text{if } \sum_{i=1}^{I} (\alpha_i u_{ih} + v_{ih}) + \alpha_{I+1} \geq N - i_0 + 1 \\ \bar{u}_{i_0,h} = 0, & \text{otherwise.} \end{cases}$$

If $N - i_0 < \alpha_{I+1}$ then

$$\begin{cases} \bar{u}_{i_0,h} = 1, & \text{if } \sum_{j=1}^{N} \sum_{\substack{k=1 \\ \hat{c}_{jk} \geq \hat{c}_{(h)}}}^{N} r_{jk} \geq N - i_0 + 1 \\ 0, & \text{otherwise.} \end{cases}$$

Observe that the variables above defined satisfy constraints (11) and (12). Thus, $(r, x, \bar{u}_0)$ is a feasible solution of (3)–(13).

In addition, by construction, in both cases the solutions provide the same objective value. Therefore the result follows. □

3. Properties

We start with a general property of ordered median functions that allows to derive lower bounds. In order to describe that property we need to introduce some notation. For any $s = 1, \dots, N$, let j_s be the index such that

$$\sum_{j=j_s}^{I+1} (\alpha_j + \beta_j) < s \leq \sum_{j=j_s-1}^{I+1} (\alpha_j + \beta_j). \tag{24}$$

Note that the above inequality means that s is a position, with respect to the components of the $\tilde{\lambda}$-vector, that corresponds either to an element in the $(j_s - 1)$-th block or to an element between the $(j_s - 2)$-th and the $(j_s - 1)$-th blocks of non-null elements. Let $K_s = \sum_{j=j_s}^{I+1} (\alpha_j + \beta_j) + \beta_{j_s-1}$, i.e. K_s accounts for the number of components of the $\tilde{\lambda}$-vector starting from the $(j_s - 1)$-th block of non-null elements.

Our first property states a lower bound on any feasible solution of the problem based on a related median objective function, for suitable choices of weights.

Proposition 3.1. *For any $s \in \{1, \dots, N\}$ and $S \subseteq \{1, \dots, N\}$ with $|S| = s$, let $\theta \in \mathbb{R}_+^N$ be such that*

$$\begin{cases} \sum_{j=j_s}^{I} \beta_j \gamma_j + \left(s - \sum_{j=j_s}^{I+1} (\alpha_j + \beta_j) \right) \gamma_{j_s-1} \geq \sum_{j \in S} \theta_j, & \text{if } s \leq K_s \text{ and } j_s < I + 1 \\ (s - \alpha_{I+1}) \gamma_I \geq \sum_{j \in S} \theta_j, & \text{if } s \leq K_s \text{ and } j_s = I + 1 \\ \sum_{j=j_s-1}^{I} \beta_j \gamma_j \geq \sum_{j \in S} \theta_j, & \text{otherwise}, s > K_s. \end{cases} \tag{25}$$

Then $\sum_{i=1}^{I} \sum_{h=2}^{G} \gamma_i (\hat{c}_{(h)} - \hat{c}_{(h-1)}) v_{ih} \geq \sum_{j=1}^{N} \sum_{k=1}^{N} \theta_j \hat{c}_{jk} r_{jk}$, for any feasible solution of (16)–(23).

Proof. Let (r, x, u, v) be a feasible solution of (16)–(23). Then, it exists a permutation π such that:

1. $\sum_{k=1}^{N} \hat{c}_{\pi(i)k} r_{\pi(i)k} \leq \sum_{k=1}^{N} \hat{c}_{\pi(i+1)k} r_{\pi(i+1)k}$, for all $i = 1, \dots, N - 1$, and
2. $\sum_{i=1}^{I} \sum_{h=2}^{G} \gamma_i (\hat{c}_{(h)} - \hat{c}_{(h-1)}) v_{ih} = \sum_{i=1}^{N} \lambda_i \sum_{k=1}^{N} \hat{c}_{\pi(i)k} r_{\pi(i)k}$.

Next, clearly follows that condition (25) is equivalent, in the original representation of the λ-weights, to: $\sum_{i\in S}\theta_i \leq \sum_{i=N-s+1}^{N}\lambda_i, \forall s \in \{1,\ldots,N\}$ and $S \subseteq \{1,\ldots,N\}, |S| = s$. Therefore, we have

$$\begin{aligned}
\sum_{i=1}^{N}\lambda_i \sum_{k=1}^{N}\hat{c}_{\pi(i)k}r_{\pi(i)k} &= \sum_{i=1}^{N}\left(\sum_{\ell=N-i+1}^{N}\lambda_\ell\right)\left(\sum_{k=1}^{N}\hat{c}_{\pi(N-i+1)k}r_{\pi(N-i+1)k} - \sum_{k=1}^{N}\hat{c}_{\pi(N-i)k}r_{\pi(N-i)k}\right)\\
&\geq \sum_{i=1}^{N}\left(\sum_{\ell=N-i+1}^{N}\theta_{\pi(\ell)}\right)\left(\sum_{k=1}^{N}\hat{c}_{\pi(N-i+1)k}r_{\pi(N-i+1)k} - \sum_{k=1}^{N}\hat{c}_{\pi(N-i)k}r_{\pi(N-i)k}\right)\\
&= \sum_{i=1}^{N}\theta_{\pi(i)}\sum_{k=1}^{N}\hat{c}_{\pi(i)k}r_{\pi(i)k}\\
&= \sum_{j=1}^{N}\sum_{k=1}^{N}\theta_j\hat{c}_{jk}r_{jk}. \quad \square
\end{aligned}$$

The above proposition gives us a new lower bound on the solution of (16)–(23) based on solving a relaxation of a classical hub location problem with additive weights for suitable choices of θ. The reader may note that in the case of non-decreasing λ-weights one can always set $\theta_i = \frac{1}{N}\sum_{j=1}^{N}\lambda_j$ for all $i = 1,\ldots,N$ as a valid additive weight for the above lower bound on Proposition 3.1. Another suitable choice for the vector θ is $\theta = \lambda$. The bound obtained by the above methodology will be called $LB_1(\theta)$. In our case, we have used it for solving median hub location problems with $\theta_i = \frac{1}{N}\sum_{j=1}^{N}\lambda_j$ for all $i = 1,\ldots,N$. This bound has proven to be helpful in the cases of median, k-centrum and center. We have also observed that Median problems are particularly easy since, in general, these problems are solved in the root node by the lower bound LB_1. From an implementation point of view, we obtain the value of this bound by solving, with the XPRESS solver, the classical hub location problem using the formulation (16)–(23), where the constraints (18)–(21) have been removed.

3.1. Variable fixing

This section addresses the description of some preprocessing steps that we propose to reduce the size of our improved formulation. Due to the definition of the variables in this formulation, one can expect that many u-and-v-variables in the right hand part of the matrix of u-and-v-variables will take value 0 in the optimal solution. Indeed, $u_{ih} = 0$ means that the $\sum_{j=1}^{i-1}(\alpha_j+\beta_j)+1$-th sorted allocation cost is less than $\hat{c}_{(h)}$ which is very likely to be true if h is sufficiently large and i is perhaps not that large. The same type of arguments also suggests that one may expect that $u_{ih} = 1$ whenever i is large and h is small to medium size because this would mean that the $\sum_{j=1}^{i-1}(\alpha_j+\beta_j)+1$-th sorted allocation cost would not have been done at cost less than $\hat{c}_{(h)}$. Note that an analogous strategy, that we will explain in detail later, applies to the v-variables since their interpretation is similar. With these strategies, the size of the formulation could be reduced if some (hopefully many) of these u-and-v variables were fixed beforehand. In this subsection we describe a number of variable fixing possibilities for the set of u-and-v-variables which are useful in the overall solution process. These variable fixing procedures are based on non straightforward adaptations of some arguments already used in [32] for a different formulation. The reader should note that the preprocessing phase developed in this paper also obtains new upper bounds on the v variables.

First of all, since $c_{jj} = 0\ \forall j = 1,\ldots,N$, it is clear that $u_{i1} = 1, v_{i1} = \beta_i, \forall i = 1,\ldots,I$. Moreover, whenever $\hat{c}_{jk} \neq 0$ if and only if $j \neq k$ then we can also fix $u_{i2} = 1, v_{i2} = \beta_i, \forall i = 1,\ldots,I$.

3.1.1. Fixing u-and-v-variables to their upper bound

In order to fix u_{ih}-and-v_{ih}-variables to their upper bound for $i = 1,\ldots,I, h = 1,\ldots,G$, we will deal with an auxiliary problem that maximizes the number of origin–first hub allocations satisfying $\hat{c}_{jk} \leq \hat{c}_{(h-1)}$. Let

$$z_{jk} = \begin{cases} 1, & \text{if origin site } j \text{ is assigned to hub } k\\ 0, & \text{otherwise.}\end{cases}$$

Using these variables, the formulation of this problem is:

$$\begin{aligned}
&\max\ H1_h := \sum_{j=1}^{N}\sum_{k=1}^{N} z_{jk}\\
&\text{s.t. } z_{jk}\hat{c}_{jk} \leq \hat{c}_{(h-1)}, \quad \forall j,k = 1,\ldots,N\\
&\sum_{k=1}^{N} z_{jk} \leq 1, \quad \forall j = 1,\ldots,N\\
&z_{jk} \leq y_k, \quad \forall j,k = 1,\ldots,N
\end{aligned}$$

$$\sum_{k=1}^{N} y_k \le p,$$
$$z_{jk}, y_k \in \{0,1\}, \quad j,k = 1,\ldots,N.$$

If $H1_h$ is the optimal value of problem above, since there are N origin–first hub allocations, the number of allocations satisfying $\hat{c}_{jk} \ge \hat{c}_{(h)}$ must be necessarily greater than or equal to $N - H1_h + 1$. Thus, let us denote by $i_0 \in \{1,\ldots,I\}$ the index such that

$$p + \sum_{j=1}^{i_0-1}(\alpha_j+\beta_j) < H1_h \le p + \sum_{j=1}^{i_0}(\alpha_j+\beta_j).$$

Then, we have that in any feasible solution of the problem:

$$\begin{cases} u_{ih}=1, \quad v_{ih}=\beta_i, \quad i=i_0,\ldots,I, & \text{if } H1_h \le p+\alpha_{i_0}+\sum_{j=1}^{i_0-1}(\alpha_j+\beta_j) \\ \left\{\begin{array}{l} v_{i_0h} \ge p+\sum_{j=1}^{i_0}(\alpha_j+\beta_j) - H1_h, \\ u_{ih}=1, \quad v_{ih}=\beta_i, \quad i=i_0+1,\ldots,I, \end{array}\right\} & \text{otherwise.} \end{cases}$$

It is worth noting that for any column h in the two cases above, we fix u variables for $i = i_0+1,\ldots,I$. Therefore, the greater the value of I the larger the number of fixed variables.

Another procedure to fix u-variables to 1 is the following. Assume that $u_{i_1,\ell} = 0$ for a fixed $i_1 \in \{1,\ldots,I\}$ and $\ell \in \{1,\ldots,G\}$. Let $S(\ell) = \{j : \min_{k(\neq j)=1,\ldots,N} \hat{c}_{jk} \ge \hat{c}_{(\ell)}\}$ be the set of sites such that their smallest allocation costs to a first hub is at least $\hat{c}_{(\ell)}$. Then we have

$$N - p - \sum_{i=1}^{i_1}\alpha_i - \sum_{i=1}^{i_1-1}\beta_i \ge \sum_{i=1}^{I}\alpha_i u_{i\ell} + \sum_{i=1}^{I} v_{i\ell} + \alpha_{I+1} \ge \sum_{j=1}^{N}\sum_{\substack{k=1\\ \hat{c}_{jk}\ge\hat{c}_{(\ell)}}}^{N} r_{jk} \ge |S(\ell)| - p. \tag{26}$$

Note that the first inequality follows from (19) to (20) together with the information that $u_{i_1\ell} = 0$. The middle inequality is (18). Finally, the last inequality says that the number of allocations that are at least $\hat{c}_{(\ell)}$ must be greater than or equal to the number of sites whose smallest allocation cost is at least $\hat{c}_{(\ell)}$ minus the at most p potential self-allocations. From (26) it follows that a solution with $u_{i_1,\ell} = 0$ can be feasible only if

$$N - \sum_{i=1}^{i_1}\alpha_i - \sum_{i=1}^{i_1-1}\beta_i \ge |S(\ell)|.$$

If this inequality fails to hold then $u_{i_1\ell} = 1$.

3.1.2. Fixing u-and-v-variables to 0

Under a similar rationale to the one used in the previous section, we try to fix as many u_{ih}-and-v_{ih}-variables to 0 as possible, for $i = 1,\ldots,I$ and $h = 1,\ldots,G$. In this case, we deal with an auxiliary problem that maximizes the number of origin–first hub allocations satisfying $\hat{c}_{jk} \ge \hat{c}_{(h)}$.

$$\max\ H2_h := \sum_{j=1}^{N}\sum_{k=1}^{N} z_{jk}$$
$$\text{s.t. } \hat{c}_{jk} \ge z_{jk}\hat{c}_{(h)}, \quad \forall j,k = 1,\ldots,N$$
$$\sum_{k=1}^{N} z_{jk} \le 1, \quad \forall j = 1,\ldots,N$$
$$z_{jk} \le y_k, \quad \forall j,k = 1,\ldots,N$$
$$\sum_{k=1}^{N} y_k \le p,$$
$$y_k, z_{jk} \in \{0,1\}, \quad \forall j,k = 1,\ldots,N.$$

Therefore, if $H2_h$ is the optimal value of problem above, it means that there are no feasible solutions of the problem with less than $N - H2_h$ allocations done at a cost at most $\hat{c}_{(h)}$. Let $1 \le i_2 \le I$ be the index such that

$$p + \sum_{j=1}^{i_2-1}(\alpha_j+\beta_j) < N - H2_h \le p + \sum_{j=1}^{i_2}(\alpha_j+\beta_j).$$

Thus, in any feasible solution of the problem we have that:

$$\begin{cases} u_{ih}=0, \quad v_{ih}=0, \quad i=1,\ldots,i_2-1, & \text{if } N-H2_h \le p+\sum_{j=1}^{i_2}\alpha_j+\sum_{j=1}^{i_2-1}\beta_j \\ \left\{\begin{matrix} u_{i_2,h}=0, \quad v_{i_2h}\le p+\sum_{j=1}^{i_2}(\alpha_j+\beta_j)-(N-H2_h), \\ u_{ih}=0, \quad v_{ih}=0, \quad i=1,\ldots,i_2-1, \end{matrix}\right\} & \text{otherwise.} \end{cases}$$

Note that whenever $N-H2_h=p$ then there is nothing to fix and therefore no variables are set to zero in column h. We also point out, that analogously to the case of fixing variables to their upper bound, the greater the value of I the larger the number of variable fixed to zero.

3.2. Valid inequalities

This section is devoted to describe several families of valid inequalities of this problem that help in solving the problems and shed light on the structure of its polyhedral description. It is worth mentioning that with the exception of the families of inequalities (27), (28), (33) and (34) the rest are new and they have not been used before.

First, we present a family of valid inequalities that are a straightforward consequence of the definition of the u-and-v-variables, but that help a lot in solving the problem. This first family is:

$$u_{ih}\ge u_{i,h+1}, \quad i=1,\ldots,I, h=1,\ldots,G-1, \tag{27}$$

$$v_{ih}\ge v_{i,h+1}, \quad i=1,\ldots,I, h=1,\ldots,G-1. \tag{28}$$

Let us consider two u-variables, namely $u_{i_0h_0}$ and $u_{i_1h_1}$ with $i_0\le i_1$ and $h_0<h_1$. Then, it is clear that the number of allocations that are made with allocation costs $\hat{c}_{(h_0)}\le\hat{c}_{jk}\le\hat{c}_{(h_1)}$ must be less than or equal to the sum of the allocation variables whose costs are in the range of those values. This can be written for the u-variables as:

$$\left(\beta_{i_0}+\sum_{i=i_0+1}^{i_1-1}(\alpha_i+\beta_i)+\alpha_{i_1}\right)(u_{i_0h_0}-u_{i_1,h_1+1})\le\sum_{\hat{c}_{(h_0)}\le\hat{c}_{jk}\le\hat{c}_{(h_1)}} r_{jk}, \quad \forall i_0\le i_1, h_0<h_1, \tag{29}$$

and for the entire range of u and v variables as:

$$\sum_{i=1}^{I}(v_{ih_0}-v_{i,h_1+1})+\alpha_{i+1}(u_{ih_0}-u_{i+1,h_1+1})\le\sum_{\hat{c}_{(h_0)}\le\hat{c}_{jk}\le\hat{c}_{(h_1)}} r_{jk}, \quad \forall i_0\le i_1, h_0<h_1. \tag{30}$$

The third set of valid inequalities states that for any subset S of s origin sites, the number of assignments whose allocation costs are done at a costs greater than or equal to $\hat{c}_{(h)}$ cannot be greater than the number of unresolved allocations among those assigned at the s largest costs. In order to represent these inequalities, we use the indexes j_s and K_s defined in (24). Now, we distinguish two cases depending on the value of K_s:

1. If $j_s<I+1$:

$$\sum_{j\in S}\sum_{\substack{k=1\\ \hat{c}_{jk}\ge\hat{c}_{(h)}}}^{N} r_{jk}\le\begin{cases}\sum_{j=j_s}^{I}(\alpha_j u_{jh}+v_{jh})+\left(s-\sum_{j=j_s}^{I}(\alpha_j+\beta_j)\right), & \text{if } s<K_s,\\ \alpha_{I+1}+\sum_{j=j_s}^{I}(\alpha_j u_{jh}+v_{jh})+v_{j_s-1,h}, & \text{if } s=K_s,\\ \alpha_{I+1}+\sum_{j=j_s}^{I}(\alpha_j u_{jh}+v_{jh})+v_{j_s-1,h}+(s-K_s)u_{j_s-1,h}, & \text{otherwise, } s>K_s.\end{cases} \tag{31}$$

2. If $j_s=I+1$:

$$\sum_{j\in S}\sum_{\substack{k=1\\ \hat{c}_{jk}\ge\hat{c}_{(h)}}}^{N} r_{jk}\le\begin{cases} s, & \text{if } s<K_s,\\ \alpha_{I+1}+v_{Ih}, & \text{if } s=K_s,\\ \alpha_{I+1}+v_{Ih}+(s-K_s)u_{Ih}, & \text{otherwise, } s>K_s.\end{cases} \tag{32}$$

To separate the above inequalities for a given fractional solution (r^*, x^*, u^*, v^*), a column h and a size s we solve the following subproblem:

$$\max \sum_{j,k=1|\hat{c}_{jk}\geq \hat{c}_{(h)}}^{N} r^*_{jk} z_j$$
$$\text{s.t.} \sum_{j=1}^{N} z_j = s$$
$$z_j \in \{0, 1\}.$$

We observe that this problem can be solved easily by inspection. Indeed, we simply choose s non-repeated indices j such that r^*_{jk} are the greatest elements in the set $\{r^*_{jk} : j, k = 1, \ldots, N, \hat{c}_{jk} \geq \hat{c}_{(h)}\}$ and set $z_j = 1$ for those indices and $z_j = 0$ otherwise.

The fourth family of valid inequalities state disjunctive implications on the origin–first hub allocation costs. The first one ensures that either origin site j is allocated to a first hub at a cost of at least $\hat{c}_{(h)}$ or there is an open hub k such that $\hat{c}_{jk} < \hat{c}_{(h)}$.

$$\sum_{k=1:\hat{c}_{jk}\geq \hat{c}_{(h)}}^{N} r_{jk} + \sum_{k=1:\hat{c}_{jk}<\hat{c}_{(h)}}^{N} r_{kk} \geq 1, \quad \forall j = 1, \ldots, N, h = 1, \ldots, G. \tag{33}$$

Swapping the roles of the r_{jk}-and r_{kk}-variables in the above inequality, we obtain:

$$\sum_{k=1:\hat{c}_{jk}\geq \hat{c}_{(h)}}^{N} r_{kk} + \sum_{k=1:\hat{c}_{jk}<\hat{c}_{(h)}}^{N} r_{jk} \geq 1, \quad \forall j = 1, \ldots, N, h = 1, \ldots, G. \tag{34}$$

For the next sets of valid inequalities we need some additional notation. Given $h^* \in \{1, \ldots, G\}$, let $n(h^*)$ be the number of feasible allocations that could be done at a cost $\hat{c}_{(h^*)}$. Observe that it means the number of all pairs (j, k), such that, $\hat{c}_{jk} = \hat{c}_{(h^*)}$ and r_{jk} has not been fixed to 0 yet (either by the variable fixing process or by the branching in the branch-and-bound tree), i.e.

$$n(h^*) = |\{(j, k) : \hat{c}_{jk} = \hat{c}_{(h^*)} \text{ and } r^*_{jk} \text{ not fixed to } 0\}|. \tag{35}$$

Clearly, if $n(h^*) = 0$ then no possible allocations can be done at this cost, from the current feasible solution, and thus columns h^* and $h^* + 1$ must be equal for u-and-v-variables.

$$u_{ih^*} - u_{i,h^*+1} \leq 0, \qquad v_{ih^*} - v_{i,h^*+1} \leq 0, \quad \forall i = 1, \ldots, I. \tag{36}$$

Next, we observe that the number of allocations that can be done at a cost exactly of $\hat{c}_{(h^*)}$ is at most $n(h^*)$. Therefore, if an allocation is made in column h^* and it is ranked at position ℓ, clearly no other allocation at cost $\hat{c}_{(h^*)}$ can be ranked at position $\ell + n(h^*)$ since it would mean that at least $n(h^*) + 1$ allocations would have been made at a cost $\hat{c}_{(h^*)}$ which is not possible. This fact inspires the next set of valid inequalities. Set $m_0 = 1$ and

$$m_{i+1} = \min\left\{i' : \sum_{j=m_i+1}^{i'} (\alpha_j + \beta_j) \geq n(h^*), m_i < i' \leq I\right\}.$$

By convention, $m_{i+1} = +\infty$ if $\sum_{j=m_i+1}^{I} (\alpha_j + \beta_j) < n(h^*)$ (note that if $m_i = +\infty$ then $m_j = +\infty$ for all $j > i$). For the ease of notation, let $I_m = \max\{i : m_i < +\infty\}$. It is clear that m_i is the number of elements of the minimum number of complete consecutive blocks (non-null) that one has to add to the position m_{i-1} to go beyond of $m_{i-1} + n(h^*)$ positions.

Then, between all the differences of variables that are spaced at least $n(h^*)$ ordered positions at most one can attain the upper value. Hence, we have

$$\sum_{i=m_0}^{I_m} \frac{(v_{m_i,h^*} - v_{m_i,h^*+1})}{\beta_{m_i}} \leq 1, \quad h^* = 1, \ldots, G-1, \tag{37}$$

or equivalently, in terms of the u-variables:

$$\sum_{i=m_0}^{I_m} (u_{m_i,h^*} - u_{m_i,h^*+1}) \leq 1, \quad h^* = 1, \ldots, G-1. \tag{38}$$

3.3. A reformulation of the problem at the nodes where all location variables are fixed

Assume that a complete set of hubs has been opened, i.e. we have a set $X \subseteq \{1, \ldots, N\}$, such that, $|X| = p$ and $r_{kk} = 1$ for any $k \in X$ and thus $r_{kk} = 0$ for all $k \notin X$. This information simplifies the formulation of Problem (16)–(23) since the knowledge of which hubs are operating at that solution leads to have a compact form of the second block of costs, i.e. those that correspond to hub–hub, hub–final destination deliveries.

Indeed, depending on the character of the pair origin–final destination site we can easily compute the value of the minimum second block delivery cost. The reader easily understands that, for instance if the origin site j and the final destination m are not hubs, i.e. $r_{jj} = r_{mm} = 0$, the minimum cost of delivering the flow via the first hub k is given by the minimum among all potential choices for the second hub in X or the direct delivering, namely $\min_{\ell\in X}(\mu c_{k\ell}+\delta c_{\ell m})w_{jm}$. Considering all possible subcases the formula below gives us, $\zeta_{jkm}(X)$, the minimum second block cost of delivering flow from the origin site j via first hub k with final destination m.

$$\zeta_{jkm}(X) := \begin{cases} \min\limits_{\ell\in X}(\mu c_{k\ell}+\delta c_{\ell m})w_{jm} & \text{if } r_{jj}=0, r_{kk}=1, r_{mm}=0\\ \min\limits_{\ell\in X}(\mu c_{k\ell}+\delta c_{\ell m})w_{jm} & \text{if } r_{jj}=1, j=k, r_{mm}=0\\ \mu c_{jm}w_{jm} & \text{if } r_{jj}=1, j=k, r_{mm}=1\\ \mu c_{km}w_{jm} & \text{if } r_{jj}=0, r_{kk}=1, r_{mm}=1,\\ +\infty & \text{otherwise.}\end{cases} \tag{39}$$

From the above equations, we obtain, $\tilde{c}_{jk}(X)$, the minimum second block cost of all the flow sent from the origin site j via hub $k\in X$ in the conditions described above.

$$\tilde{c}_{jk}(X) := \sum_{m=1}^{N}\zeta_{jkm}(X), \quad \forall j,k=1,\dots,N. \tag{40}$$

Note that when $j\in X$, i.e. j is a hub itself, then the only term that makes sense is $\tilde{c}_{jj}(X)$, since we assume single allocation throughout the first hub.

The above information allows us to reformulate the problem as a special case of ordered assignment problem once a complete set of open hubs (p locations) is known. For this reformulation we only need variables r, u and v as defined in (2), (14) and (15), respectively. Thus, given the set of hubs X with $|X|=p$, the reformulation of Problem (16)–(23) is as follows:

$$\min \sum_{i=1}^{I}\sum_{h=2}^{G}\gamma_i(\hat{c}_{(h)}-\hat{c}_{(h-1)})v_{ih}+\sum_{j=1}^{N}\sum_{k=1}^{N}r_{jk}\tilde{c}_{jk}(X)$$

s.t. Constraints: (4), (5) and (18)–(23).

This problem is similar to Problem (16)–(23) where we have removed the constraints implied by the knowledge of which hubs are open and forbidden in the current solution.

Reformulating the original problem in the above form reduces the CPU time needed to solve the subproblems at complete location nodes (those nodes where the number of sites already fixed to be open hubs is p) in the B&B algorithm.

4. A specialized branch and bound for the discrete hub ordered location problem

The driving variables for the *Single-Allocation Ordered Hub location problem* are the binary r-variables, indicating which sites have been selected for hub location, r_{jj}, and the allocation of each origin site j to the first hub k in its delivery paths to all its final destinations, r_{jk}. Once these variables are known, the objective value is easy to calculate since it reduces to an "ad hoc" enumeration. It thus makes sense to build a branch and bound (B&B) method based initially on the r_{kk} variables, i.e. on decisions of whether or not a site is selected to be an open hub. Then, we proceed with the branching of the remaining variables.

We develop a B&B in which each node contains information of a disjoint pair of sets of sites, two sets of disjoint pairs and a range of integer values. For a given node, let $H\subseteq A$ denote the set of open hubs and $\overline{H}\subseteq A\setminus H$ denote the set of forbidden hubs. We refer to the sites in the set $A\setminus(H\cup\overline{H})$ as *undecided*. By OH we refer to the set of allocation pairs between origin sites and open hubs in H, whereas by $\overline{OH}$ we refer to the forbidden allocations, i.e. those that have been discarded and will never be feasible in this particular solution. Moreover, V denotes the ranges of admissible integer values for the v-variables in that node. Thus, a node in the B&B tree contains the 5-tuple $(H,\overline{H},OH,\overline{OH},V)$. Of course, a node is a *complete location* node if either $|H|\geq p$ or $|\overline{H}|\geq N-p$. Clearly, if $(j,k)\in OH, j\neq k$, it implies that $k\in H$ and $j\notin H$ since open hubs are always allocated to themselves.

Based on the above discussion, in the following we develop new lower bounds. Some of them are based on solving either the linear or the Lagrangian relaxations or an 'ad hoc' lower bound based on another type of problem. Yet another one, of combinatorial nature, that is simpler and it does not require the solution of any optimization problem. We discuss our lower bounds in detail in Section 4.1. We have also obtained several combinatorial upper bounds based on completions of partial solutions that are described in Section 4.2 and we discuss our branching rule in detail in Section 4.3. In Section 5 we compare computationally the performance of our B&B method with that of the best integer linear programming formulation.

4.1. Combinatorial lower bounds

At each node which is not a *complete location node*, we need to calculate lower bounds on the value of the cost function, aiming to discard the node by bounding. Assume that a set of sites H, $|H|\leq p$ has been opened as hubs, and another set $\overline{H}$, $|\overline{H}|\leq N-p$ of sites is forbidden to be used as hubs, i.e. $r_{kk}=1$ for any $k\in H$ and $r_{kk}=0$ for any $k\in\bar{H}$. Let $A\setminus\overline{H}$ be the set of sites which are either open hubs or undecided. Then, one can easily extend the definition in (39), to $\zeta'_{jkm}(H,\overline{H})$ to be the minimum second block cost of delivering from the origin site j to the final destination m, either directly or using as a first hub $k \in A \setminus \overline{H}$ when j is not a hub itself (and possibly a second hub also in $A \setminus \overline{H}$ when m is not either a hub).

From the above equations, we obtain, $\tilde{c}'_{jk}(H, \overline{H})$, the minimum second block cost of all the flow sent from the origin site j via hub $k \in A \setminus \overline{H}$ in the conditions described above,

$$\tilde{c}'_{jk}(H, \overline{H}) := \sum_{m=1}^{N} \zeta'_{jkm}(H, \overline{H}), \quad \forall j, k = 1, \dots, N. \tag{41}$$

Note that when $j \in H$, then the only term that makes sense is $\tilde{c}'_{jj}(H, \overline{H})$, since we assume self-service and single allocation throughout the first hub. Next, define

$$\hat{c}'_j(H, \overline{H}) = \begin{cases} \min\limits_{k \in A \setminus \overline{H}, k \neq j} \hat{c}_{jk}, & \text{if } j \in A \setminus H \\ 0, & \text{otherwise.} \end{cases} \tag{42}$$

$$\tilde{c}'_j(H, \overline{H}) = \min_{k \in A \setminus \overline{H}} \tilde{c}'_{jk}(H, \overline{H}). \tag{43}$$

Clearly (42) is the minimum first block delivery cost for a customer at site j to a hub in $k \in A \setminus \overline{H}$. Analogously, (43) stands for the minimum second block, namely between hubs and second hub–final destination, i.e., delivery cost, once the first hub has been achieved, of the commodity sent from origin j to all final destinations using as hubs only those in $A \setminus \overline{H}$. Let σ be the permutation of $1, \dots, N$ such that

$$\hat{c}'_{\sigma(1)}(H, \overline{H}) \leq \cdots \leq \hat{c}'_{\sigma(N)}(H, \overline{H}).$$

Next, observe that in any feasible solution there are p costs of the first block of costs of the objective function equal to zero. These costs correspond to deliveries of origin sites that are hubs to first hubs (themselves). The remaining sites, that are not hubs, send their flow necessarily via some first hub. This implies that the first block of costs of the objective function must have $N - p$ non-null addends.

Now, the second part of the cost of the objective function corresponds to the transportation costs between hubs and second hub–final destination delivery costs. Obviously, if we assume that only sites that are in $A \setminus \overline{H}$ can be used as hubs, then $\tilde{c}'_j(H, \overline{H})$ gives the cheapest way to send the delivery from site j to all destinations.

Then, we define $LB_2(H, \overline{H})$ as

$$LB_2(H, \overline{H}) = \sum_{j=1}^{N-p} \lambda_{p+j}\, \hat{c}'_{\sigma(|H|+j)}(H, \overline{H}) + \sum_{j=1}^{N} \tilde{c}'_j(H, \overline{H}). \tag{44}$$

In the following proposition we present a lower bound on the objective function value of any feasible solution having facilities in H as open hubs and $\overline{H} \subseteq A$ as forbidden to be open hubs.

Proposition 4.1. *Given $H, \overline{H} \subseteq A$, $H \cap \overline{H} = \emptyset$ with $|H| < p$ and $|\overline{H}| < N - p$, let $S, \overline{S} \subseteq A \setminus (H \cup \overline{H})$ with $S \cap \overline{S} = \emptyset$, $|H \cup S| \leq p$ and $|\overline{H} \cup \overline{S}| \leq N - p$. Then*

$$LB_2(H \cup S, \overline{H} \cup \overline{S}) \geq LB_2(H, \overline{H}).$$

Proof. Let σ^i with $i = 1, 2$ be the permutations such that

$$\hat{c}'_{\sigma^1(1)}(H, \overline{H}) \leq \cdots \leq \hat{c}'_{\sigma^1(N)}(H, \overline{H}),$$
$$\hat{c}'_{\sigma^2(2)}(H \cup S, \overline{H} \cup \overline{S}) \leq \cdots \leq \hat{c}'_{\sigma^2(N)}(H \cup S, \overline{H} \cup \overline{S}).$$

We know from (44) that:

$$LB_2(H \cup S, \overline{H} \cup \overline{S}) = \sum_{j=p+1}^{N} \lambda_j\, \hat{c}'_{\sigma^2(j)}(H \cup S, \overline{H} \cup \overline{S}) + \sum_{k=1}^{N} \tilde{c}'_j(H \cup S, \overline{H} \cup \overline{S}) \quad \text{and}$$

$$LB_2(H, \overline{H}) = \sum_{j=p+1}^{N} \lambda_j \hat{c}'_{\sigma^1(j)}(H, \overline{H}) + \sum_{k=1}^{N} \tilde{c}'_j(H, \overline{H}).$$

First of all, by their definitions (see (42) and (43)), $\hat{c}'_j(H \cup S, \overline{H} \cup \overline{S}) \geq \hat{c}'_j(H, \overline{H})$ and $\tilde{c}'_j(H \cup S, \overline{H} \cup \overline{S}) \geq \tilde{c}'_j(H, \overline{H})$ for all $j = 1, \dots, N$. Therefore, using [31, Theorem 1.1] we obtain that $\hat{c}'_{\sigma^2(i)}(H \cup S, \overline{H} \cup \overline{S}) \geq \hat{c}'_{\sigma^1(i)}(H, \overline{H})$ for all $j = 1, \dots, N$. Hence, since we assume $\lambda \geq 0$ the inequality $LB_2(H \cup S, \overline{H} \cup \overline{S}) \geq LB_2(H, \overline{H})$ holds. $\square$

As a consequence of Proposition 4.1 $LB_2(H,\overline{H})$ is a lower bound on the objective function value of any feasible solution having facilities in H as open hubs and in $\overline{H}$ as closed.

Example 4.1 (Example 2.1 *Continued*)**.** Suppose that we are at a node of the branch and bound tree, such that, $H=\emptyset$ and $\overline{H}=\{4\}$, i.e. site 4 is forbidden to be an open hub by branching. To calculate $LB_2(H,\overline{H})$, we compute the vectors $\hat{c}'(H,\overline{H})$, $\tilde{c}'(H,\overline{H})$ and $\hat{c}'_{\sigma(j)}(H,\overline{H})$.

j	1	2	3	4	5	6	7	8	9	10
$\tilde{c}'(H,\overline{H})$	247.4	277.2	161.2	169.3	308.8	338.5	344.4	382.2	253.7	374.8
$\hat{c}'(H,\overline{H})$	91	106	70	67	100	218	101	520	285	984
$\hat{c}'_{\sigma(j)}(H,\overline{H})$	67	70	91	100	101	106	218	285	520	984

Therefore, the computation of the three addends of the lower bound $LB_2(H,\overline{H})$ given by (44) reduces to:

$$\begin{aligned}\sum_{j=p+1}^{N}\lambda_j\,\hat{c}'_{\sigma(j)}(H,\overline{H}) &= 1\times 91+1\times 100+0\times 101+0\times 106+1\times 218\\ &\quad +1\times 285+1\times 520+0\times 984=1214,\end{aligned}$$

$$\sum_{k=1}^{N}\tilde{c}'_j(H,\overline{H})=247.4+277.2+161.2+169.3+308.8+338.5+344.4+382.2+253.7+374.8=2857.5.$$

From the above, the lower bound (44) applied to this example results in $LB_2(H,\overline{H})=4071.5$.

The third lower bound is based on a Lagrangian relaxation of the original problem. Assume that we are in a node of the B&B tree defined by $(H,\overline{H},OH,\overline{OH},V)$. We relax constraints (18) in the formulation (16)–(23), with multipliers $\xi=(\xi_1,\dots,\xi_G)$, $\xi_i\geq 0$ for all $i=1,\dots,G$, which results in:

$$\begin{aligned}L(\xi) \;=\; \min &\sum_{i=1}^{I}\left[\sum_{h=2}^{G}\Big(\gamma_i(\hat{c}_{(h)}-\hat{c}_{(h-1)})v_{ih}-\xi_h(\alpha_i u_{ih}+v_{ih})\Big)\right]\\ &-\sum_{i=1}^{I}\xi_1(\alpha_i u_{i1}+v_{i1})\sum_{k=1}^{N}\sum_{\ell=1}^{N}\sum_{m=1}^{N}x_{k\ell m}(\mu c_{k\ell}+\delta c_{\ell m})+\sum_{h=1}^{G}\sum_{j=1}^{N}\sum_{\substack{k=1\\ \hat{c}_{jk}\geq\hat{c}_{(h)}}}^{N}\xi_h r_{jk}-\alpha_{I+1}\sum_{h=1}^{G}\xi_h\end{aligned}$$

s.t. (4)–(10) and (19)–(23).

We observe that the above problem can be split into two subproblems.

1. The problem on the u, v variables is

$$P_{u,v}(\xi):=\min\sum_{i=1}^{I}\left[\sum_{h=2}^{G}\Big(\gamma_i(\hat{c}_{(h)}-\hat{c}_{(h-1)})v_{ih}-\xi_h(\alpha_i u_{ih}+v_{ih})\Big)\right]-\sum_{i=1}^{I}\xi_1(\alpha_1 u_{i1}+v_{i1}) \tag{45}$$

s.t. (19)–(22).

This first subproblem can be easily solved by inspection. Assume that ξ is given, we proceed as follows. First, using the expression of (45) we can fix u_{i1} and v_{i1} for all $i=1,\dots,I$: $u_{i1}:=1$ and $v_{i1}:=\beta_1$. Then for a given $h\in\{2,\dots,G\}$ we compute

$$\eta^h_{ri}=\begin{cases}\gamma_i(\hat{c}_{(h)}-\hat{c}_{(h-1)})r+\displaystyle\sum_{j=i+1}^{I}\gamma_i(\hat{c}_{(h)}-\hat{c}_{(h-1)})\beta_j-\left(r+\sum_{j=i+1}^{I}(\alpha_j+\beta_j)\right)\xi_h, & \text{if } r<\beta_i\\ \displaystyle\sum_{j=i}^{I}\gamma_i(\hat{c}_{(h)}-\hat{c}_{(h-1)})\beta_j-\left(\sum_{j=i}^{I}(\alpha_j+\beta_j)\right)\xi_h, & \text{if } r=\beta_i\end{cases}$$

for all $i=1,\dots,I$, $r=1,\dots,\beta_i$. Clearly, these values correspond to the part of the objective function that is due to the allocation in the h-th column of $(0,\overset{i}{\dots},0,1,\dots,1)$ for the u variables and $(0,\overset{i-1}{\dots},0,r,\beta_{i+1},\dots,\beta_I)$ for the v variables, if $r<\beta_i$; and $(0,\overset{i-1}{\dots},0,1,\dots,1)$ for the u and $(0,\overset{i-1}{\dots},0,\beta_i,\beta_{i+1},\dots,\beta_I)$ for the v variables, if $r=\beta_i$. Hence, we look for the minimum among those allocations. Let $\eta^h_{r^*i^*}=\min_{\left(\substack{i=1,\dots,I\\ r=1,\dots,\beta_i}\right)}\eta^h_{ri}$. Then, an optimal assignment for the variables in the h-column is

$$u_{ih}:=\begin{cases}1, & \text{if } i>i^*\\ 1, & \text{if } i=i^* \text{ and } r=\beta_i\\ 0, & \text{otherwise.}\end{cases}\qquad v_{ih}:=\begin{cases}0, & \text{if } i<i^*\\ r^*, & \text{if } i=i^*\\ \beta_i, & \text{if } i>i^*.\end{cases}$$

2. The problem on the x and r variables is

$$P_{xr}(\xi) := \min \sum_{k=1}^{N}\sum_{\ell=1}^{N}\sum_{m=1}^{N} x_{k\ell m}(\mu c_{k\ell} + \delta c_{\ell m}) + \sum_{h=1}^{G}\sum_{j=1}^{N}\sum_{\substack{k=1\\ \hat{c}_{jk}\geq \hat{c}_{(h)}}}^{N} \xi_h r_{jk}$$

s.t. (4)–(10) and (23).

This problem does not have the integrality property, i.e. its linear relaxation may not be integer. Nevertheless, it is relatively easy to solve to optimality because the number of integer variables and constraints is not too large.

Clearly, $L(\xi) = P_{uv}(\xi) + P_{xr}(\xi) - \alpha_{I+1}\sum_{h=1}^{G}\xi_h^*$. In our lower bound we replace the original $P_{xr}(\xi)$ by its linear relaxation, $LP_{xr}(\xi)$. We solve the Lagrangian dual of the modified Lagrangian relaxation where the second problem is substituted by its linear relaxation, namely *DualVal* $:= \max_\xi L'(\xi) := P_{uv}(\xi) + LP_{x,r}(\xi) - \alpha_{I+1}\sum_{h=1}^{G}\xi_h^*$. In solving this problem we use standard subgradient algorithm and our initial Lagrangian multipliers are $\xi_h = \frac{1}{N-1}\sum_{i=2}^{I}\gamma_i(\hat{c}_{(h)} - \hat{c}_{(h-1)})\beta_i \,\forall h \geq 2$ and $\xi_1 = 1$. The value *DualVal* coincides with the optimal value of the linear relaxation of the formulation given by (16)–(23). Nevertheless, once the optimal dual multiplier, ξ^*, is obtained we improve this lower bound by solving, in integer variables, $P_{x,r}(\xi^*)$ (Note that since $P_{xr}(\xi^*)$ does not exhibit the integrality property, one may expect to raise the lower bound). Once this value is available the third lower bound

$$LB_3 := P_{uv}(\xi^*) + P_{xr}(\xi^*) - \alpha_{I+1}\sum_{h=1}^{G}\xi_h^*. \tag{46}$$

Note that this strategy works because solving to optimality, in integer variables, the subproblem $P_{x,r}(\xi^*)$ is not very time consuming since these problems are easy to solve.

Example 4.2. Here, we compare the power of the lower bounds derived from (Proposition 3.1), (44) and (46) and the linear relaxation of Problem (16)–(23). We consider the data given in Example 2.1 with $H = \emptyset, \overline{H} = \{4\}$ and its original λ-weight. In addition, we also show the result with a different vector of non-decreasing monotone lambda weights $\lambda = (0, 0, 0, 0, 1, 1, 1, 1, 1, 1)$. The results are:

λ-weight	(0, 0, 0, 0, 1, 1, 1, 1, 1, 1)	(0, 0, 1, 1, 0, 0, 1, 1, 1, 0)
LB_1 (Proposition 3.1)	8012.5	6084.15
LB_2 (44)	6769.9	4071.5
LB_3 (46)	6897.9	6679.9
Linear relaxation	5503.6	4616.15
Objective	9761.9	8162.9

From the above table we conclude that none of the lower bounds uniformly dominates the others. Indeed, for $\lambda = (0, 0, 0, 0, 1, 1, 1, 1, 1, 1)$ bound LB_1 is the best whereas for $\lambda = (0, 0, 1, 1, 0, 0, 1, 1, 1, 0)$ the best one is LB_3. In addition, although most of the times LB_2 is dominated by some other (not always) its quality is rather good and its computing time is very short, making it competitive. Actually, we have observed that the second and fourth (LP relaxation) lower bounds can easily be computed whereas the first and third lower bounds require more effort and CPU time. The goal of this approach is to avoid solving the subproblem improving the lower bound, at the cost of increasing the CPU time needed to get them. Hence, *LB*, the best bound among them, could be:

$$LB := \max\{LB_1, LB_2, LB_3\}. \tag{47}$$

We observe that none of the lower bounds are trivial at the root node.

In the implementation, our strategy has been to compute LB_2 at any node of the B&B tree since it is not time consuming and its performance is better provided that the number of fixed variables in the B&B tree increases. On the other hand, bounds LB_1 and LB_3 have been computed in all nodes, only up to a fixed depth due to the effort to compute them (in our experiments we choose a depth equal to 5). They are also computed in those nodes such that the number of location variables that are fixed is above a threshold value that depends on the problem type and size.

4.2. Combinatorial upper bounds

This section describes combinatorial upper bounds for our problem. These bounds are computed at the root node but also at any node of the branching tree trying to improve the incumbent solution or even pruning the node. Therefore, we assume that a partial solution $(H, \overline{H}, OH, \overline{OH}, V)$ is given, then the idea is to construct feasible solutions that are consistent with the structure of $(H, \overline{H}, OH, \overline{OH}, V)$.

First of all, we have considered two different approaches to the completion process of the set H up to a set X with cardinality p ($|X| = p$). Once, this completion has been done, we will find feasible delivery paths between each pair of origin–destination sites that only use hubs in X.

1. The first completion approach is based on ranking any candidate site k_0 by its overall transportation cost, assuming that the flow of all the sites of $A \setminus H$ go through k_0 as first hub. Thus, we compute for each site $k_0 \in A \setminus (\overline{H} \cup H)$ the index $\Sigma_{k_0}(H)$ defined below. Recall that $\hat{c}_{jk_0} = c_{jk_0} W_j$, for all $j \notin A \setminus H$, $(j, k_0) \notin \overline{OH}$ and $j \neq k_0$. Let π^{k_0} be a permutation that sort the vector $(\hat{c}_{jk_0})_{j \in A \setminus H}$, such that

   $$\hat{c}_{\pi^{k_0}(1)} \leq \cdots \leq \hat{c}_{\pi^{k_0}(N-|H|)}.$$

   Then, we define

   $$\Sigma_{k_0}(H) := \sum_{j=1}^{N-|H|} \lambda_{j+|H|} \hat{c}_{\pi^{k_0}(j)} + \sum_{j \in X \setminus H} \sum_{m=1}^{N} \zeta'_{jk_0 m}(H, \overline{H}), \quad \forall k_0 \in A \setminus (\overline{H} \cup H). \tag{48}$$

   Let $\psi = N - |\overline{H} \cup H|$ and let η be a permutation that sorts the expressions above for $k_0 \in A \setminus (\overline{H} \cup H)$ such that

   $$\Sigma_{\eta(1)} \leq \Sigma_{\eta(2)} \leq \cdots \leq \Sigma_{\eta(\psi)}.$$

   Then, $X = H \cup \{\eta(1), \ldots, \eta(p - |H|)\}$.
2. The second completion approach is based on setting to 1 those location variables (r_{kk}) that support the largest proportion of allocations from the origin sites. First of all, we fix to one those variables that correspond to hubs in H and to zero those in $\overline{H}$, i.e. we set $r_{kk} = 1, \forall k \in H$ and $r_{kk} = 0, \forall k \in \overline{H}$. Next, we solve the relaxed LP-problem with the above variables already fixed and from the continuous solution we compute for each hub k the amount of allocations delivered via first hub k, i.e. $F_k = \sum_{j=1}^{N} r_{jk}$. Then, we fix to one $p - |H|$, r_{kk} variables corresponding to those k with the largest values of F_k.

Once we have completed a set X of open hubs with $|X| = p$, we describe two ways to compute two different upper bounds based on the pattern given to the allocations of origin sites to first hubs. Therefore, we will have four different upper bounds since we have two different ways to complete the set X.

1. The first bound is the simplest one because it requires less computation burden. For any origin $j \in \{1, \ldots, N\}$ we choose as its first hub $\mathcal{K}(j) \in X$ if either $(j, \mathcal{K}(j)) \in OH$ or $\hat{c}_{j\mathcal{K}(j)} = \min_{k \in X, (j,k) \notin \overline{OH}} \hat{c}_{jk}$.

   Now, for each origin site $j = 1, \ldots, N$, we compute the value of the second block of delivery cost provided that the first hub used by j is $\mathcal{K}(j)$. That is,

   $$\tilde{c}_{j\mathcal{K}(j)}(X) := \sum_{m=1}^{N} \zeta_{j\mathcal{K}(j)m}(X). \tag{49}$$

   Clearly, we have constructed a solution that for each origin site j delivers its flow, first via $\mathcal{K}(j)$ and then optimally to all their final destinations using only hubs in X. To compute the upper bound, we sort the elements $(\hat{c}_{j\mathcal{K}(j)})_{j=1,\ldots,N}$ in nondecreasing sequence with the permutation $\sigma^{\mathcal{K}}$. Moreover, its objective value is:

   $$UB_1(H, \overline{H}, OH, \overline{OH}) := \sum_{j=1}^{N} \left[\hat{c}_{\sigma^{\mathcal{K}}(j)} \lambda_j + \tilde{c}_{j\mathcal{K}(j)}(X) \right]. \tag{50}$$

2. The second upper bound is similar but using $\mathcal{K}'(\cdot)$ a different allocation rule for the assignment of origin sites to their first hub. This allocation rule is based on a ranking of the hubs by an estimate of the overall cost they would produce. The rationale is as follows. For each origin site j we compute the estimate of its contribution to the objective function provided that it would go via first hub k:

   $$\Lambda(\hat{c}_{jk}) \hat{c}_{jk} + \sum_{m=1}^{N} \zeta_{jkm}(X).$$

   Note that since we do not know "a priori" the position of $\hat{c}_{jk}$ in the sorted sequence of costs we do not know the actual λ-parameter that would multiply it. To avoid this problem we use the estimate $\Lambda(\hat{c}_{jk})$ which is based on assuming a uniform distribution of costs in its entire range $[0, \hat{c}_{(G)}]$. We define $\Lambda : \mathbb{R} \to \{\lambda_1, \ldots, \lambda_N\}$ so that $\Lambda(a) = \lambda_i$ if $a \in \left[\hat{c}_{(\lfloor (i-1)*G/N \rfloor)}, \hat{c}_{(\lfloor i*G/N \rfloor)}\right)$, for $i = 1, \ldots, N$.

   Next, we compute

   $$\check{c}_j(X) := \min_{k \in X, (j,k) \notin OH \cup \overline{OH}} \left[\Lambda(\hat{c}_{jk}) \hat{c}_{jk} + \sum_{m=1}^{N} \zeta_{jkm}(X) \right].$$

   Then, set $\mathcal{K}'(j)$, for all $j = 1, \ldots, N$, to be either $\mathcal{K}'(j) = k$ if $(j, k) \in OH$ or the hub that provides the above minimum, i.e.

   $$\check{c}_j(X) = \Lambda(\hat{c}_{j\mathcal{K}'(j)}) \hat{c}_{j\mathcal{K}'(j)} + \sum_{m=1}^{N} \zeta_{j\mathcal{K}'(j)m}(X), \quad \text{if } (j, k) \notin OH.$$

Table 2
The indices $\Sigma_k(H)$ and $\sum_j r_{jk}$ applied to Example 2.1.

k	1	2	3	5	6	7	8	9	10
$\Sigma_k(H)$	7223	7600.2	8472.1	8492.7	7542.6	9359.2	**6655**	**6805.4**	8626.3
$\sum_j r_{jk}$	0.9236	1.0593	0.9522	1.0774	**1.8422**	1.2338	**1.6719**	1.1712	0.0685

Table 3
Computation of the first upper bound with the set $X = \{8, 9\}$.

j	1	2	3	4	5	6	7	8	9	10
$\hat{c}_{j,8}$	**455**	**106**	**560**	1072	**1300**	**218**	606	**0**	475	2337
$\hat{c}_{j,9}$	1638	212	840	**268**	2000	218	**101**	1248	**0**	**984**
$\hat{c}_{\sigma^{\mathcal{K}}(j)}$	0	0	101	106	218	268	455	560	984	1300
$\tilde{c}_{j\mathcal{K}(j)}(\{8, 9\})$	687.6	870.3	639	504.8	854.7	981.9	654.9	1054.8	756.2	933.7

Table 4
Computation of the first upper bound with the set $X = \{6, 8\}$.

j	1	2	3	4	5	6	7	8	9	10
$\hat{c}_{j,6}$	819	1696	1330	**67**	**200**	**0**	1111	1248	1140	**1968**
$\hat{c}_{j,8}$	**455**	**106**	**560**	1072	1300	218	**606**	**0**	**475**	2337
$\hat{c}_{\sigma^{\mathcal{K}}(j)}(\{6, 8\})$	0	0	67	106	200	455	475	560	606	1968
$\tilde{c}_{j\mathcal{K}(j)}$	627.3	788.1	519.3	321.6	658.9	558.3	874.5	1008.6	631.5	720.3

To compute the upper bound, we sort the elements $(\hat{c}_{j\mathcal{K}'(j)})_{j=1,\ldots,N}$ in nondecreasing sequence with the permutation $\sigma^{\mathcal{K}'}$. Finally, the actual objective value of the feasible solution built with the allocation given by $\mathcal{K}'(j)$ for all j is

$$UB_2(H, \overline{H}, OH, \overline{OH}) := \sum_{j=1}^{N} \left[\hat{c}_{\sigma^{\mathcal{K}'}(j)} \lambda_j + \tilde{c}_{j\mathcal{K}'(j)}(X) \right]. \tag{51}$$

Example 4.3. Once more, we illustrate the upper bounds with the data of Example 2.1. There are two methods to complete the set of open sites up to the total number of p. The first way is based on choosing the $p - |H|$ smallest values of the index $\Sigma_k(H)$.

Indeed, consider that $\overline{H} = \{4\}$ and $H = OH = \overline{OH} = \emptyset$, i.e. site 4 is forbidden to be a hub and no site is already open for a hub in this solution. Note that $\psi = 10 - 1$. We compute the index $\Sigma_k(H)$, for all k and the two smallest ones correspond to sites 8 and 9. These results are shown in Table 2. Hence, the set of open hubs is given by $X = \{\eta(1), \eta(2)\} = \{8, 9\}$.

The second method to complete a partially defined set of open hubs is based on some manipulations of the values of the linear relaxation of the problem. Indeed, for each $k \in A \setminus (\overline{H} \cup H)$, we compute the values $\sum_j r_{jk}$. Based on Table 2, we complete the partial solution with the two sites providing the largest values of $\sum_j r_{jk}$. In our example, since $H = \emptyset$ we complete with the sites 6, 8 that correspond to the two largest values 1.8422, 1.6719. Thus $X = \{6, 8\}$.

As for the upper bounds we have defined two different approaches which in turns when combined with the two completion methods results in 4 different upper bounds.

The first upper bound consists of applying the rationale of (50) on the first completion approach that in this example results in the set $X = \{8, 9\}$. For each origin site j, find the open hub $\mathcal{K}(j)$ and the sorted sequence $\hat{c}_{\sigma^{\mathcal{K}}(j)}$ of such values.

Then, $\sum_{j=1}^{N} \hat{c}_{\sigma^{\mathcal{K}}(j)} \lambda_j = 2206$ and (49) is given by $\sum_{j=1}^{N} \tilde{c}_{j\mathcal{K}(j)}(X) = 7937.9$, see Table 3. Thus, the first upper bound with the first completion scheme is:

$$UB_1^1(\{8, 9\}, \{4\}, \emptyset, \emptyset) := \sum_{j=1}^{N} \left[\hat{c}_{\sigma^{\mathcal{K}}(j)} \lambda_j + \tilde{c}_{j\mathcal{K}(j)}(X) \right] = 10143.9.$$

Now, we apply this upper bound to the second completion scheme based on the values of the linear relaxation. Recall that for this completion $X = \{6, 8\}$. Then, we apply the same rationale as above resulting the values in Table 4. Based on these data the resulting upper bound is

$$UB_1^2(\{6, 8\}, \{4\}, \emptyset, \emptyset) := \sum_{j=1}^{N} \left[\hat{c}_{\sigma^{\mathcal{K}}(j)} \lambda_j + \tilde{c}_{j\mathcal{K}(j)}(X) \right] = 8522.4.$$

Now, we illustrate the computation of the second upper bound for the first completion $X = \{8, 9\}$. Recall that we need to compute $\Lambda(\hat{c}_{j\mathcal{K}'(j)})$ for all $j = 1, \ldots, N$. Table 5 gives us $\Lambda(\hat{c}_{jk})$ for all pairs j, k. Next, we determine the assignment of each origin site j to the chosen first hub, according to the values of $\check{c}_j(X)$. Table 6 shows in bold letters the value of $\check{c}_j(X)$,

Table 5
$\Lambda(\hat{c}_{jk})$, for all $j, k = 1, \ldots, N$.

j	k									
	1	2	3	4	5	6	7	8	9	10
1	0	1	1	1	1	0	0	1	1	0
2	1	0	1	0	0	1	0	0	0	0
3	1	0	0	1	0	1	1	1	0	1
4	1	0	1	0	0	0	0	0	0	1
5	0	0	0	1	0	0	1	1	0	0
6	0	0	1	0	1	0	0	0	0	1
7	0	1	0	1	1	0	0	1	0	0
8	1	1	1	1	1	1	1	0	1	1
9	1	1	1	0	0	1	1	1	0	0
10	0	0	0	1	1	0	0	0	0	0

Table 6
Assignments $\mathcal{K}'(j)$ given by $\check{c}_j(\{8, 9\})$.

j	1	2	3	4	5	6	7	8	9	10
$\check{c}_j(\{8\})$	**1142.6**	870.3	1199	547.8	2154.7	981.9	1502.1	**1054.8**	1201.3	989.4
$\check{c}_j(\{9\})$	2226.7	**820.9**	**628.2**	**504.8**	**620.1**	**753.4**	**654.9**	1323.6	**756.2**	**933.7**
$\mathcal{K}'(j)$	8	9	9	9	9	9	9	8	9	9
$\hat{c}_{j\mathcal{K}'(j)}$	455	212	840	268	2000	218	101	0	0	984
$\tilde{c}_{j\mathcal{K}'(j)}(\{8, 9\})$	687.6	820.9	628.2	504.8	620.1	753.4	654.9	1054.8	756.2	933.7

Table 7
Assignments $\mathcal{K}'(j)$ given by $\check{c}_j(\{6, 8\})$.

j	1	2	3	4	5	6	7	8	9	10
$\check{c}_j(\{6\})$	**437.9**	2116.3	1532.5	**321.6**	**658.9**	**558.3**	**620.8**	2018.4	1554.2	**720.3**
$\check{c}_j(\{8\})$	1082.3	**788.1**	**1079.3**	473.1	2105.8	1017.9	1480.5	**1008.6**	**1106.5**	954.3
$\mathcal{K}'(j)$	6	8	8	6	6	6	6	8	8	6
$\hat{c}_{j\mathcal{K}'(j)}$	819	106	560	67	200	0	1111	0	475	1968
$\tilde{c}_{j\mathcal{K}'(j)}(\{6, 8\})$	437.9	788.1	519.3	321.6	658.9	558.3	620.8	1008.6	631.5	720.3

the sequence $\hat{c}_{j\mathcal{K}'(j)}$ and $\tilde{c}_{j\mathcal{K}'(j)}(\{8, 9\})$ for all $j = 1, \ldots, N$. Hence, $\sum_{j=1}^{N} \hat{c}_{\sigma\mathcal{K}'(j)}\lambda_j = 2592$, $\sum_{j=1}^{N} \tilde{c}_{j\mathcal{K}'(j)}(X) = 7414.6$ and the upper bound results in:

$$UB_2^1(\{8, 9\}, \{4\}, \emptyset, \emptyset) = 10006.6.$$

Analogously, we also compute the second upper bound for the second completion given by the set $X = \{6, 8\}$. Table 7 shows in bold letters the value of $\check{c}_j(X)$, the sequence $\hat{c}_{j\mathcal{K}'(j)}$ and $\tilde{c}_{j\mathcal{K}'(j)}(\{6, 8\})$ for all $j = 1, \ldots, N$. Therefore, $\sum_{j=1}^{N} \hat{c}_{\sigma\mathcal{K}'(j)}(X)\lambda_j = 2663$, $\sum_{j=1}^{N} \tilde{c}_{j\mathcal{K}'(j)}(X) = 6265.3$ and the upper bound is:

$$UB_2^2(\{6, 8\}, \{4\}, \emptyset, \emptyset) = 8928.3.$$

Finally, we compare the values of the four upper bounds and observe that $\min\{UB_1^1, UB_1^2, UB_2^1, UB_2^2\} = \min\{10143.9, 8522.14, 10006.6, 8928.34\} = 8522.14$ that corresponds to the first upper bound with the second completion strategy. The reader may note that the best integer solution assuming that hub 4 is closed is 8162.9.

4.3. Branching

We begin by describing the generic form of the branching rule, which is strong. For its description we assume that (1) an ordering of the undecided sites is given, i.e. a rule to branch on the location variables, (2) an ordering on the allocation of origin sites to hubs in the current solution is given (note that those origin sites that are at the same time hubs in this solution are always assigned to themselves therefore these allocations are not needed), i.e. a rule to choose the variable to branch on the allocation variables, and (3) a rule to select a fractional u or v variable is available. These three elements allow us to make the decision on which variable to branch in a given node of the B&B tree and determine the specific form of our branching strategy. We have tried several ordering although we will only describe the one that finally is implemented in our algorithm.

We assume that we first have fixed as many variables as possible, according to our variable fixing strategies described in Section 3.1. Then, our branching strategy works as follows. Initially it always proceeds first with the augmentation of the set $\overline{H}$ until a given size MAXLEVEL $< N - p$, i.e. branching on the location variables r_{kk} up to the level MAXLEVEL in the B&B

tree. For these variables we use as branching criteria the strong branching based on testing the best progress among all the location variables. This is done by fixing, one at a time, all the fractional r_{kk} to 0 and performing some iterations of the dual simplex method (75 iterations in our implementation) in order to choose that one that provides the best lower bound value. Note that this strategy is not much time consuming since the number of location variables in our test problems are not very large ($N \leq 30$).

Then, we proceed to apply the branching on the remaining fractional variables, namely r_{jk} and u_{ih}, v_{ih}, with $j, k = 1, \dots, N$ and $i = 1, \dots, I$, $h = 2, \dots, G$. For these variables we apply pseudocost branching (see [1]) to detect the most promising variable among the fractional ones. If the choice is either a location variable (r_{kk} for some k) or an allocation variable (r_{jk}, $j \neq k$) then we directly branch on that variable. However, if the choice is an u, v-variable we redefine the branching variable using an ad-hoc strategy based on some bounds on their fractional values that gives us good performance in terms of CPU time for these problems. (The interested reader may find further details in [33].)

Example 4.4. We illustrate the B&B scheme with the data of Example 1. Fig. 3 shows some nodes of the branching tree. Each node shows the set H and $\overline{H}$ of open and closed hubs, respectively. In addition, we show the best upper bound found so far *UB* and the lower bound *LB* in each node. In this example we set MAXLEVEL $= 2$. The algorithm proceeds from the root node $H = \emptyset, \overline{H} = \emptyset$ with $UB = 9725.2$ and $LB = 4082.1$. Next, it chooses $r_{22} = 0$ to branch, i.e. the new set $\overline{H} = \{2\}$, and then it moves to the left most node of the second level (see Fig. 3). Then, the algorithm branches down on $r_{66} = 0$ and after some steps, represented in the figure by the dashed arrows, it finds the only two non-pruned integer solutions of this branch that correspond to the two left-most nodes of the fourth level ($H = \{5, 8\}$ and $H = \{1, 7\}$). Analogously, the branch on $r_{66} = 1$ finds as unique integer solutions the configurations with $H = \{4, 6\}$ and $H = \{1, 6\}$. These two nodes are shown by the two right-most nodes of the fourth level of the tree.

We observe that in this example the remaining two nodes of the second level, namely $H = \{2\}, \overline{H} = \{7\}$ and $H = \{2, 7\}, \overline{H} = \emptyset$ are pruned by bounding. The overall number of nodes of the tree being 94.

5. Computational results

We implemented the B&B method using the Mosel programming language, with the upper bound initialized by a heuristic method based on our combinatorial methods described on Section 4.2. Our tree search strategy was best bound. The method was run on a Intel(R) Core(TM)2 Quad CPU Q6600 4 GB RAM.

This section reports on the computational comparison between the Pre-Covering 3-Index formulation (that uses the formulation (3)–(13) with a preprocessing of fixing variables, see [32] for further details), Improved formulation (based on (16)–(23) with the variable fixing in Section 3.1), and the B&B&Cut algorithm developed in this paper. For this purpose we use the AP data set publicly available at http://www.cmis.csiro.au/or/hubLocation (see [12]). We tested the formulations on a testbed of five instances for each combination of (i) costs matrices, (ii) N in $\{15, 20, 25, 28, 30\}$ (iii) different values of p depending on the case and (iv) $\mu = 0.7$, $\delta = 1.2\mu$ and six different λ-vectors. Among them we consider center ($\lambda = (0, \dots, 0, 1)$), k-centrum ($k = \lceil 0.2N \rceil$, $\lambda = (0, \dots, 0, 1, \overset{k}{\dots}, 1)$), $(k_1 + k_2)$-trimmed-mean ($k_1 = k_2 = \lceil 0.2N \rceil$, $\lambda = (0, \overset{k_1}{\dots}, 0, 1, \dots, 1, 0, \overset{k_2}{\dots}, 0)$), anti-$(k_1 + k_2)$-trimmed-mean ($k_1 = k_2 = \lceil 0.2N \rceil$, $\lambda = (1, \overset{k_1}{\dots}, 1, 0, \dots, 0, 1, \overset{k_2}{\dots}, 1)$), median ($\lambda = (1, \dots, 1)$) and 3-blocks (three alternate $\{0 - 1\}$-blocks of lambda weights, i.e. $\lambda = (0, \dots, 0, 1, \dots, 1, 0, \dots, 0, 1, \dots, 1, 0, \dots, 0, 1, \dots, 1)$).

Tables 8 and 9 report the comparative results of the Pre-Covering 3-Index formulation ((3)–(13), see [32]), Improved formulation (16)–(23) and our B&B&Cut. Table 8 includes problem types *Median, Center, k-Centrum* and Table 9 the remaining problems type, namely *Trimmed Mean, Anti-Trimmed Mean, 3-Blocks*. These tables have three groups of columns. The first group, with columns *R-Gap, Nodes,* and *Time*, refers to the formulation Pre-Covering 3-Index (3)–(13), the second group with columns *RGAP, %Fixed*, UB_v, *Nodes* and *Time* refers to the Improved formulation (16)–(23), whereas the third group, including columns *Nodes, TCuts, Cuts*, and *Time*, presents the information regarding our B&B&Cut procedure based on formulation (16)–(23). The first three columns of these two tables stand for the different types of problems in the study. In all groups, columns *R-GAP*, *Nodes* and *Time*, stand, respectively, for the averages of: the gap in the root node, the number of nodes in the B&B tree, and the CPU time in seconds. The columns of the second group *%Fixed* and UB_v stand for the percentage of integer variables and the number of upper bounds for v-variable that are fixed by our preprocessing (see Section 3.1). The columns in the third group *TCuts* and *Cuts*, stand for the overall number of cuts of type (27)–(34) and the number of cuts of types (29)–(32) added to the formulation. The reader may observe that columns *RGAP*, *%Fixed* and UB_v for Improved formulation (second group) are also valid for the B&B&Cut procedure (third group). The reason for that is that they report the average gap with respect to the linear relaxation of the same formulation, namely Improved (16)–(23), using the same variable fixing schemes.

The symbol "$*$" in some columns means that Xpress ran "out of memory" solving some of the five instances for the corresponding combination of parameters. The notation "$*i$" represents that only i out of the five instances were solved.

In both tables, we observe that, with few exceptions, our procedure solves the instances with sizes lower than or equal to 25 or larger, but with a small number of hubs ($p = 3$), much faster than using Pre-Covering 3-Index and faster than using Improved formulation. One can also observe that Improved is much faster than Pre-Covering 3-Index, despite that *R-Gap* is slightly larger than the one obtained by the latter formulation.

Table 8
Results for problem types median, center, and k-centrum.

	N	P	PRE-Cov-3-Index			Improved formulation					B&B& Cut			
			R-GAP	Nodes	Time	R-GAP	% Fixed	UB_v	Nodes	Time	Nodes	TCuts	Cuts	Time
	15	3	12.12	259	22.03	15.56	26.96	220.60	393	7.76	433	225	0	6.67
	15	5	5.61	266	19.61	8.21	18.05	156.00	250	5.37	829	180	0	6.61
	15	8	4.07	116	7.93	5.14	8.92	61.20	78	2.40	691	207	0	3.57
	20	3	16.77	1667	318.92	24.62	30.78	445.40	1447	51.53	1334	920	0	50.67
	20	8	4.89	1492	234.83	6.64	15.45	218.20	1258	38.98	1956	354	0	27.70
	20	10	3.53	201	47.48	4.17	6.62	103.20	213	10.49	553	262	0	11.89
	25	3	15.13	2358	1484.42	21.34	33.74	799.40	1701	139.36	490	2776	0	114.85
MEDIAN	25	8	4.02	8701	5059.93	7.59	15.47	361.80	11443	585.14	12340	626	1	277.15
	25	10	3.55	4239	2100.98	4.42	13.99	285.80	4631	221.69	5852	290	0	173.08
	28	3	16.01	4544	6496.44	23.89	27.89	853.60	4043	414.73	2114	784	0	274.27
	28	8	*2	*	*	11.40	18.22	507.40	42867	2884.19	56531	787	3	1750.55
	28	10	*3	*	*	6.66	14.55	423.20	11039	812.73	22025	786	2	795.73
	30	3	16.87	7513	15556.82	23.58	32.04	1098.40	6854	825.82	2898	1497	0	640.98
	30	8	*2	*	*	9.75	18.96	591.00	*2	*	23213	1162	2	1801.57
	30	10	*	*	*	6.42	16.66	508.60	*3	*	82937	1342	24	5123.17
	15	3	16.25	425	26.59	20.76	60.73	0.00	283	6.63	341	1342	1127	6.24
	15	5	12.00	707	27.30	13.43	32.20	0.00	493	9.00	507	1028	891	10.00
	15	8	8.61	612	17.35	9.58	11.81	0.00	453	7.44	509	1195	1015	8.29
	20	3	17.70	2520	328.03	21.53	64.61	0.00	1432	54.04	539	2574	2111	44.35
	20	8	16.65	10136	420.72	17.20	24.48	0.00	2555	72.98	1256	2259	1983	58.14
	20	10	11.87	18014	593.77	11.93	12.50	0.00	9817	210.09	4777	3589	3349	157.70
	25	3	17.40	5627	1937.44	21.16	69.18	0.00	5314	359.93	1605	6907	6482	257.66
CENTER	25	8	16.06	16353	1693.88	17.30	36.41	0.00	9333	534.09	4290	4379	4207	301.40
	25	10	13.62	15890	2117.80	13.56	26.13	0.00	18458	1051.05	11310	1909	1881	783.82
	28	3	18.66	8842	4914.40	22.60	76.22	0.00	7074	650.22	2623	9896	8783	533.76
	28	8	*1	*	*	26.24	31.87	0.00	*2	*	73797	50358	48472	20225.24
	28	10	*2	*	*	18.60	26.36	0.00	*4	*	41533	32273	31014	10536.06
	30	3	19.99	13523	11003.93	22.86	68.72	0.00	11711	1886.08	3398	10425	9813	1048.07
	30	8	*	*	*	25.03	38.74	0.00	*	*	97474	44793	42627	23519.10
	30	10	*	*	*	21.56	30.49	0.00	*1	*	110811	55008	32663	25377.11
	15	3	12.18	287	20.88	20.66	22.04	139.40	574	12.63	193	810	585	8.85
	15	5	8.25	786	26.88	14.15	15.62	97.00	1608	19.08	1101	664	457	14.89
	15	8	5.20	309	8.10	6.83	6.55	37.00	534	7.24	569	525	300	7.67
	20	3	14.68	3094	368.09	25.94	21.23	242.20	3932	142.77	550	2166	1766	75.38
	20	8	5.77	6472	332.36	8.67	10.29	108.60	5166	136.94	2859	1148	795	92.03
	20	10	4.45	2682	123.14	6.31	5.42	70.00	4153	98.51	2416	1149	749	65.61
	25	3	13.89	7063	2318.49	25.90	22.78	412.60	7259	570.43	991	3136	1406	242.95
K-CENTRUM	25	8	6.06	29662	4411.60	8.42	14.56	263.20	*4	*	31874	8050	6675	1980.21
	25	10	4.86	13956	2016.96	6.78	10.22	212.80	17982	851.57	9537	2028	1436	539.29
	28	3	11.58	8168	3391.19	21.20	27.90	554.40	13956	1274.51	5575	4010	2359	812.94
	28	8	*1	*	*	11.94	10.72	369.60	*	*	80407	16860	13870	7229.00
	28	10	*	*	*	11.92	8.96	224.20	*1	*	123084	54216	46667	10516.22
	30	3	*3	*	*	31.79	20.51	686.80	41015	4619.80	2340	9046	8146	1939.37
	30	8	*	*	*	14.20	12.42	465.20	*	*	122895	42630	33252	15795.39
	30	10	*	*	*	8.89	12.89	344.20	*2	*	105415	109408	100742	14227.02

Tables 8 and 9 show, as a general trend, that the CPU time increases similarly, for all choices of the λ-vector, with the size of the instances. In addition, Pre-covering 3-Index and the Improved formulations prove to be unable to reach optimality for most of the instances of sizes larger than $N = 25$ and $p = 8, 10$ because the solver runs out of memory.

Analogously, we also observe that, in general, the number of nodes in the B&B tree, the number of cuts and the CPU time increase as a function of N and for fixed N also increase with p. It is also interesting to point out that the R-Gap decreases with p. Comparing columns *TCuts* and *Cuts* one observes that the highest number of cuts corresponds to those of type (29)–(32) (*Cuts*). As for the implementation these cuts have been added to all nodes of the branch and bound tree up to a level depth of 10 and depending on the problem they have been added each fix number of branching nodes (between 30 and 100 for large problems). These cuts help usually reducing the number of nodes of the Branch and Bound tree. Roughly speaking, the preprocessing produces a slight reduction of the CPU time and it has a similar behavior for all the considered problem types, although for the center problem the average percentage of fixed variables is slightly higher. Regarding the number of added upper bounds for the v-variables, we observe that the highest number is stated for median problems and the lowest number for the center. Observe that both models correspond to the ones with the biggest and the smallest block of ones in their lambda parameters, respectively.

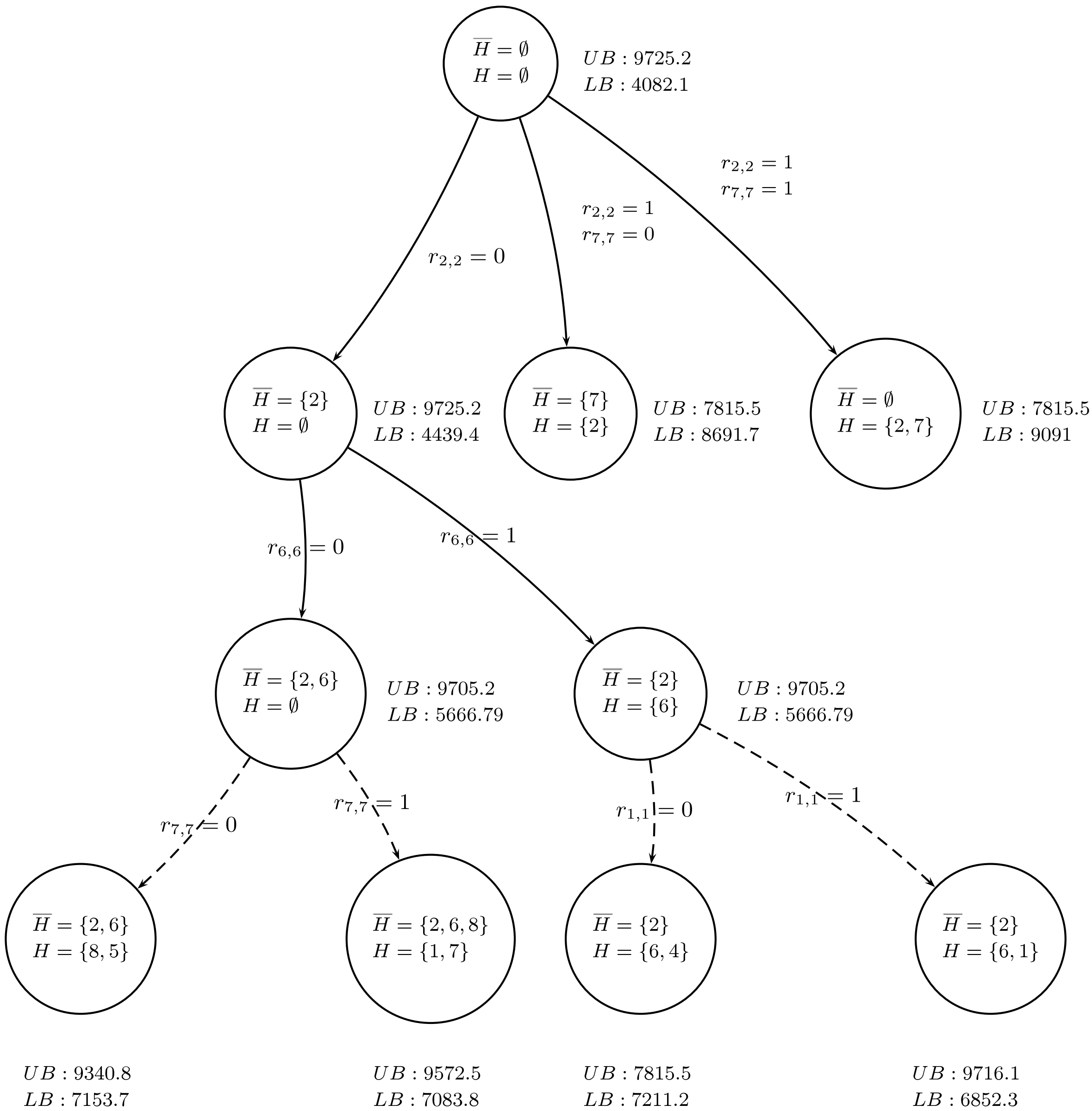

Fig. 3. A partial view of the B&B tree of Example 4.4.

We have also observed that median problems ($\lambda = (1, \ldots, 1)$) are particularly easy since, in general, these problems are solved in the root node by the lower bound LB_1 (remark that this bound reaches the optimal objective value). Finally, we also point out that our resolution method is specially suitable for Trimmed Mean problems in terms of CPU time. This effect is due to the size reduction induced by our formulation which takes advantage of the last block of zero lambda weights that does not affect the objective value.

6. Conclusions

This paper deals with the Single Allocation Ordered Median Hub Location problem introduced in [32]. Here, we have developed an exact Branch and Bound and Cut solution approach based on an improved reformulation of the original one given in [32]. This methodology provides better results than the previous one, both in CPU times and problem sizes that can be solved to optimality.

The results in this paper provide an step further on the knowledge of this class of problems although we are aware that, so far, we can handle medium size problems. Beyond these limitations, we are currently working on taking advantage of our findings to be the basis of some heuristic approaches that allow us to enlarge the problem sizes to be tackled.

Table 9
Results for problem types trimmean, antitrimmean, and blocks.

	N	P	PRE-Cov-3-Index			Improved formulation					B&B& Cut			
			R-GAP	Nodes	Time	R-GAP	% Fixed	UB_v	Nodes	Time	Nodes	TCuts	Cuts	Time
TRIMMEAN	15	3	15.80	401	12.20	17.90	22.69	75.6	415	6.45	314	341	116	5.62
	15	5	14.68	795	12.16	15.64	20.86	86.4	674	10.64	945	336	111	6.85
	15	8	10.74	288	5.28	11.09	10.31	26	271	4.51	986	389	164	8.82
	20	3	16.00	3882	194.93	18.55	23.61	138.8	1911	55.97	1177	1041	641	32.90
	20	8	12.28	4766	136.73	13.41	13.15	77.4	6061	145.50	2502	1003	603	53.59
	20	10	8.88	2944	53.12	9.49	13.54	58.2	1597	47.82	1520	651	389	40.89
	25	3	17.37	9311	1148.53	19.99	28.93	190.4	9726	451.23	1967	3547	2922	118.44
	25	8	12.32	37181	2559.94	13.58	18.00	147.6	32288	1430.17	4511	1919	1294	314.07
	25	10	17.39	23402	1198.89	17.75	16.30	116	10837	545.43	56427	5710	5079	220.99
	28	3	18.36	13508	3079.50	21.49	21.89	264.2	12768	966.67	4295	3819	3029	561.06
	28	8	*2	*	*	17.04	21.68	237.2	27581	2586.42	51996	562	130	1532.8
	28	10	*3	*	*	15.88	13.69	143.2	*4	*	22169	4750	3967	2063.51
	30	3	17.48	14414	4346.09	19.46	31.85	317.6	19143	1627.12	6871	4445	3552	1128.50
	30	8	*2	*	*	17.27	14.87	259.2	*	*	24962	5719	4819	2889.74
	30	10	*3	*	*	16.89	20.31	224.2	*3	*	11806	3228	2328	1389.32
ANTITRIMMEAN	15	3	12.77	339	19.17	20.88	36.68	142.2	813	14.62	776	757	525	9.63
	15	5	7.09	526	17.64	10.96	28.57	67.8	986	14.92	376	874	646	7.74
	15	8	4.60	261	5.63	5.83	14.87	25.6	497	8.00	559	690	460	6.61
	20	3	15.22	3129	238.71	24.08	32.35	272.6	3361	132.44	632	1901	1485	55.16
	20	8	6.33	5266	255.58	9.16	17.42	111.4	6709	183.33	4155	1321	1024	106.61
	20	10	4.16	4110	138.52	4.62	14.35	68.8	4227	109.67	5080	1883	1464	89.81
	25	3	14.94	7264	1760.41	25.11	38.05	453.8	8117	713.99	2427	3945	3033	307.64
	25	8	6.77	14715	1692.35	8.96	21.33	284.6	16602	997.59	27330	5936	5305	780.65
	25	10	6.16	52571	3653.86	6.78	15.83	194.6	43008	2384.22	43259	6902	6463	1473.78
	28	3	15.42	16266	6964.46	25.98	37.30	708.4	13989	1711.95	3384	4231	3014	887.16
	28	8	*2	*	*	12.72	28.62	272.4	43909	3806.46	23277	475	86	1906.43
	28	10	*2	*	*	10.12	16.22	222.8	*2	*	76471	173753	39429	8933.26
	30	3	18.03	23595	12139.43	26.58	39.60	736.4	18338	2476.20	5677	7582	5913	1803.36
	30	8	*	*	*	12.50	26.14	420	*1	*	92013	65513	59022	20700.93
	30	10	*	*	*	9.61	22.18	275.2	*	*	132451	87457	77763	26269.46
3-BLOCKS	15	3	12.65	383	21.23	19.65	36.41	133	575	16.65	451	858	626	11.96
	15	5	8.06	521	15.29	9.12	28.28	100.8	957	19.02	1099	710	520	16.23
	15	8	8.84	375	9.91	9.41	13.59	34.2	551	9.06	888	533	304	7.62
	20	3	15.01	5028	388.57	19.70	33.48	226.8	1600	65.60	1321	1774	1358	52.72
	20	8	6.29	6892	400.83	7.49	20.10	85.8	7685	271.88	9019	1022	725	231.55
	20	10	3.93	4606	195.19	4.31	10.14	53.2	9020	222.13	9038	2550	2133	186.67
	25	3	15.91	13447	3549.53	20.48	41.62	409.2	5663	479.20	5694	2173	285	456.39
	25	8	10.12	36990	4158.78	10.82	18.39	173.8	33855	1715.91	11662	6012	5359	1011.68
	25	10	*3	*	*	9.87	16.53	146.4	*4	*	72672	10985	10463	3178.88
	28	3	16.44	19107	5852.49	20.82	38.65	386.4	5231	899.58	5201	5304	2507	775.14
	28	8	*2	*	*	11.88	26.41	274.4	*4	*	43053	9554	8710	5111.62
	28	10	*	*	*	8.10	17.04	255.4	*	*	69039	23206	19397	5571.86
	30	3	18.12	26726	17905.67	21.95	43.93	591	26107	4102.11	14127	8740	6343	2032.55
	30	8	*	*	*	10.07	30.16	301.4	*	*	108583	95711	94371	21965.72
	30	10	*	*	*	10.03	22.41	272.8	*	*	64493	29936	23883	11593.95

Acknowledgments

This research has been partially supported by Spanish Ministry of Education and Science/FEDER grant numbers MTM2007-67433-C02-(01-02), MTM2010-19576-C02-(01-02) and DE2009-0057, Junta de Andalucia grant number FQM 05849 and Fundacin Séneca, grant number 08716/PI/08.

References

[1] T. Achterberg, T. Koch, A. Martin, Branching rules revisited, Operations Research Letters 33 (1) (2005) 42–45.
[2] S. Alumur, B.Y. Kara, Network hub location problems: the state of the art, European Journal of Operational Research 190 (1) (2008) 1–21.
[3] N. Boland, P. Domínguez-Marín, S. Nickel, J. Puerto, Exact procedures for solving the discrete ordered median problem, Computers and Operations Research 33 (2006) 3270–3300.
[4] N. Boland, M. Krishnamoorthy, A.T. Ernst, J. Ebery, Preprocessing and cutting for multiple allocation hub location problems, European Journal of Operational Research 155 (3) (2004) 638–653.
[5] R. Bollapragada, Y. Li, U.S. Rao, Budget-constrained, capacitated hub location to maximize expected demand coverage in fixed-wireless telecommunication networks, INFORMS Journal on Computing 18 (4) (2006) 422–432.
[6] J.F. Campbell, Hub location and the p-hub median problem, Operations Research 44 (6) (1996) 923–935.
[7] J.F. Campbell, A. Ernst, M. Krishnamoorthy, Hub location problems, in: Facility Location, Springer, Berlin, 2002, pp. 373–407.
[8] A.M. Campbell, T.J. Lowe, L. Zhang, The p-hub center allocation problem, European Journal of Operational Research 176 (2) (2007) 819–835.

[9] J.F. Campbell, M.E. O'Kelly, Twenty five years of hub location research, Transportation Science 46 (2) (2012) 153–169.
[10] L. Cánovas, S. García, A. Marín, Solving the uncapacitated multiple allocation hub location problem by means of a dual-ascent technique, European Journal of Operational Research 179 (3) (2007) 990–1007.
[11] I. Contreras, J.A. Díaz, E. Fernández, Lagrangean relaxation for the capacitated hub location problem with single assignment, OR Spectrum 31 (3) (2009) 483–505.
[12] A.T. Ernst, M. Krishnamoorthy, Efficient algorithms for the uncapacitated single allocation p-hub median problem, Location Science 4 (3) (1996) 139–154.
[13] A.T. Ernst, M. Krishnamoorthy, Solution algorithms for the capacitated single allocation hub location problem, Annals of Operations Research 86 (1999) 141–159.
[14] M.C. Fonseca, A. García-Sánchez, M. Ortega-Mier, F. Saldanha-da-Gama, A stochastic bi-objective location model for strategic reverse logistics, TOP 18 (1) (2010) 158–184.
[15] H. Hamacher, M. Labbé, S. Nickel, T. Sonneborn, Adapting polyhedral properties from facility to hub location problems, Discrete Applied Mathematics 145 (1) (2004) 104–116.
[16] J. Kalcsics, S. Nickel, J. Puerto, A.M. Rodríguez-Chía, Distribution systems design with role dependent objectives, European Journal of Operational Research 202 (2010) 491–501.
[17] J. Kalcsics, S. Nickel, J. Puerto, A.M. Rodríguez-Chía, The ordered capacitated facility location problem, TOP 18 (1) (2010) 203–222.
[18] B.Y. Kara, B.C. Tansel, On the single-assignment p-hub center problem, European Journal of Operational Research 125 (3) (2000) 648–655.
[19] B.Y. Kara, B.C. Tansel, The single-assignment hub covering problem: models and linearizations, Journal of the Operational Research Society 54 (1) (2003) 59–64.
[20] J. Kratica, Z. Stanimirovic, Solving the uncapacitated multiple allocation p-hub center problem by genetic algorithm, Asia-Pacific Journal of Operational Research 23 (4) (2006) 425–437.
[21] M. Labbé, H. Yaman, Projecting the flow variables for hub location problems, Networks 44 (2) (2004) 84–93.
[22] M. Labbé, H. Yaman, Solving the hub location problem in a star–star network, Networks 51 (1) (2008) 19–33.
[23] M. Labbé, H. Yaman, E. Gourdin, A branch and cut algorithm for hub location problems with single assignment, Mathematical Programming 102 (2) (2005) 371–405.
[24] A. Marín, Formulating and solving splittable capacitated multiple allocation hub location problems, Computers and Operations Research 32 (12) (2005) 3093–3109.
[25] A. Marín, Uncapacitated Euclidean hub location: strengthened formulation, new facets and a relax-and-cut algorithm, Journal of Global Optimization 33 (3) (2005) 393–422.
[26] A. Marín, L. Cánovas, M. Landete, New formulations for the uncapacitated multiple allocation hub location problem, European Journal of Operational Research 172 (1) (2006) 274–292.
[27] A. Marín, S. Nickel, J. Puerto, S. Velten, A flexible model and efficient solution strategies for discrete location problems, Discrete Applied Mathematics 157 (5) (2009) 1128–1145.
[28] A. Marín, S. Nickel, S. Velten, An extended covering model for flexible discrete and equity location problems, Mathematical Methods of Operations Research 71 (1) (2010) 125–163.
[29] T. Meyer, A.T. Ernst, M. Krishnamoorthy, A 2-phase algorithm for solving the single allocation p-hub center problem, Computers & Operations Research 36 (12) (2009) 3143–3151.
[30] A. Newman, C. Yano, P. Kaminsky, Third party logistics planning with routing and inventory costs, in: J. Geunes, P. Pardalos (Eds.), Supply Chain Optimization, Springer, USA, 2005, pp. 87–122.
[31] S. Nickel, J. Puerto, Location Theory—A Unified Approach, Springer, 2005.
[32] J. Puerto, A.B. Ramos, A.M. Rodríguez-Chía, Single-allocation ordered median hub location problems, Computers and Operations Research 38 (2011) 559–570.
[33] A.B. Ramos, Localización de concentradores para mediana ordenada, Ph.D. Thesis Facultad de Matemáticas, Universidad de Sevilla, 2012 (in Spanish).
[34] I. Rodríguez-Martín, J.J. Salazar-González, Solving a capacitated hub location problem, European Journal of Operational Research 184 (2) (2008) 468–479.
[35] D. Simchi-Levi, P. Kaminsky, E. Simchi-Levi, Designing & Managing the Supply Chain: Concepts, Strategies & Case Studies, second ed., McGraw-Hill/Irwin, New York, 2003.
[36] P.Z. Tan, B.Y. Kara, A hub covering model for cargo delivery systems, Networks 49 (1) (2007) 28–39.
[37] B. Wagner, Model formulations for hub covering problems, Journal of the Operational Research Society 59 (7) (2008) 932–938.
[38] H. Yaman, Polyhedral analysis for the uncapacitated hub location problem with modular arc capacities, SIAM Journal on Discrete Mathematics 19 (2) (2005) 501–522.
[39] G. Zhou, H. Min, M. Gen, The balanced allocation of customers to multiple distribution centers in a supply chain network: a genetic algorithm approach, Computers and Industrial Engineering 43 (2002) 251–261.