INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH

WILEY

INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH

Intl. Trans. in Op. Res. 0 (2021) 1–24
DOI: 10.1111/itor.12993

On hub location problems in geographically flexible networks

Víctor Blanco[a,*] and Justo Puerto[b]

[a] *Institute of Mathematics, Universidad de Granada, Granada 18011, Spain*
[b] *Institute of Mathematics, Universidad de Sevilla, Seville 41012, Spain*
E-mail: vblanco@ugr.es [Blanco]; puerto@us.es [Puerto]

Received 24 July 2020; received in revised form 15 March 2021; accepted 30 April 2021

Abstract

In this paper, we propose an extension of the uncapacitated hub location problem where the potential positions of the hubs are not fixed in advance. Instead, they are allowed to belong to a region around an initial discrete set of nodes. We give a general framework in which the collection, transportation, and distribution costs are based on norm-based distances and the hub-activation setup costs depend not only on the location of the hub that are opened but also on the size of the region where they are placed. Two alternative mathematical programming formulations are proposed. The first one is a compact formulation while the second one involves a family of constraints of exponential size that we separate efficiently giving rise to a branch-and-cut algorithm. The results of an extensive computational experience are reported showing the advantages of each of the approaches.

Keywords: hub location; mixed integer nonlinear programming; neighborhoods; network design

1. Introduction

Hub-and-spoke networks have attracted the attention of the location analysis community in the recent years since they allow to efficiently route commodities between customers in many transportation systems. In these networks, the flow between customers, rather than being sent directly user to user, is routed via some transshipment points, *the hubs nodes*. Arcs between hubs nodes are cheaper than normal ones and thus the overall transportation costs are reduced due to the economy of scale induced by sending a large amount of flow through the hub arcs. Hub location problems combine two types of decisions: (1) the optimal placement of the hub facilities and (2) the best routing strategies on the induced hub-and-spoke network, that is, the amount of flow sent through the spoke-to-hub and the hub-to-hub links. The interested reader is referred to Alumur and Kara (2008), Campbell and O'Kelly (2012), Farahani et al. (2013), Contreras and O'Kelly (2019), and Alumur et al. (2021) for recent surveys on hub location problems.

*Corresponding author.

1.1. State of the art

The literature of single-allocation hub location problems usually distinguishes between capacitated and uncapacitated problems. In the latter case, one may consider either that a given number of hubs, p, must be located, dealing with the p-hub location problem (see O'Kelly, 1987) or that a setup cost for each of the potential hubs is provided, and the optimal number of hubs is part of the decision process. Here, we deal with a new version of the uncapacitated hub location problem with fixed costs (UHLPFC). The first formulation for UHLPFC was presented in O'Kelly (1992). There, a mixed integer programming problem with quadratic (nonconvex) objective function was proposed for the problem. Such a difficulty was overcome by solving different p-hub location problems for different values of p. In Abdinnour-Helm and Venkataramanan (1998), a different formulation was proposed based on modeling how the flow between each pair of nodes is sent between them. In Labbé and Yaman (2004), the authors presented different linearizations for the quadratic terms in the objective function using a family of 4-index variables and projecting them out. All the above papers deal with exact methods but also some heuristics and metaheuristic approaches have been proposed for the problem (see, for instance, Abdinnour-Helm, 1998; Topcuoglu et al., 2005). One can also find several extensions of the UHLPFC, as the incorporation of congestion and service time (Alumur et al., 2018), modular capacities (Hoff et al., 2017), incomplete but graph-structured backbone networks (Contreras et al., 2009, 2010; Martins de Sá et al., 2015; Blanco and Marín, 2019) flow-dependent transportation costs (Tanash et al., 2017), robust and uncertain versions (Habibzadeh Boukani et al., 2016; Peiró et al., 2019), extensions to r-allocation rules (Peiró et al., 2018; Brimberg et al., 2020), or covering models (Ernst et al., 2017). Also, a flexible approach based on ordered median functions have been incorporated to different hub location problems (Puerto et al., 2011, 2013, 2016), which is particularly useful to obtain robust solutions in hub problems by applying k-centrum, trimmed-mean, or antitrimmed-mean criteria.

When designing a (discrete) hub-and-spoke network, one usually assumes that the positions of a set of potential hubs are known, that is, one makes the decision on the optimal location of the hubs based on the assumption that they have to be located on a set of nodes whose exact placement on a given space is provided. In many cases such an assumption becomes highly restrictive, especially in those cases in which some flexibility is allowed for the location of the hubs, as in telecommunication networks or in case some imprecision affects the positions of the nodes. An alternative to this assumption is the consideration of a continuous framework in which the hubs are allowed to be located (see O'Kelly, 1986b). Nevertheless, it is unrealistic in practice to assume that hubs can be located at any place. An intermediate perspective, which is the one that we explore in this paper, comes from modeling such an issue via *neighborhoods*. This framework allows us to provide a potential set of (exact) positions for the hub nodes, but in the decision process the decision maker is allowed to place the hubs not only at their potential original geographical coordinates but in a region around them, namely, their neighborhoods. This approach is particularly useful in networks in which the decision maker sets preferred regions where the nodes can be located, instead of an exact set of positions for them. Moreover, the continuous and the discrete cases of this problem are just particular cases of this more general form of the problem by adjusting adequately the parameters describing the neighborhoods. For instance, defining neighborhoods as singletons, the uncapacitated single-allocation hub location problem with variable-size neighborhoods (UHLPN) coincides with the discrete UHLPFC, while using big enough neighborhoods (and zero costs for their size) results in the continuous version of the problem.

This approach of considering neighborhoods in other combinatorial optimization problems is not totally new and some previous attempts can be found in the literature as for instance the minimum cost spanning tree problem (Dorrigiv et al., 2015; Blanco et al., 2017), traveling salesman problem (Löffler and van Kreveld, 2010; Gentilini et al., 2013; Disser et al., 2014; Váňa and Faigl, 2015; Carrabs et al., 2017; Yang et al., 2018), shortest path problem (Disser et al., 2014), Crossing postman problem (Puerto and Valverde, 2020), or different facility location problems (Blanco, 2019). However, as far as we know, there is no previous attempt on the simultaneous determination of the location and optimal size of the neighborhoods. Observe that neighborhoods may represent location imprecision and some constraint over the full flexibility on the placement of the objects under analysis. Thus, fixing a prespecified neighborhood size implies losing one degree of freedom, which reduces the ability of the model to choose the right neighborhood size. This is particularly convenient in the hub-and-spoke network model under analysis. Note also that the information on the *optimal* size of the neighborhood may help the decision maker to adjust the original positions of the nodes or to restrict the sizes conveniently.

1.2. Contributions

In this paper, we analyze hub location problems allowing some flexibility on the underline network design problem in a very general framework. In particular, we introduce an extension of the classical uncapacitated single-allocation hub location problem, which we call the UHLPN. As far as we know, this problem has not been previously addressed in the literature. Actually, only a few papers have analyzed hub location problems in the continuous framework (see O'Kelly, 1986a, 1986b; Aykin, 1988; Aykin and Brown, 1992; Aykin, 1995). We provide a general mathematical programming formulation for the problem and propose two different mixed integer nonlinear programming (MINLP) reformulations of the problem where the nonlinearities come from the products of binary variables with binary and/or continuous variables and the constraints modeling the membership of hubs to their neighborhoods. While both reformulations have, initially, a polynomial number of constraints, the latter is rewritten as an MINLP with a smaller number of variables but exponentially many constraints. A branch-and-cut approach is derived for solving this model. The results of our computational experiments show that the branch-and-cut approach is more convenient in terms of the required CPU time for solving the problems.

The rest of the paper is organized as follows. In Section 2, we introduce the elements describing the UHLPN and fix the notation for the rest of the sections. Section 3 provides a mathematical programming formulation for the problem as well as two different reformulations. While the first reformulation is based on linearizing the bilinear and trilinear terms in the original formulation, the second reformulation consists of replacing some of the nonlinear constraints by a family of linear constraints of exponential size. Based on the latter, in Section 4 we derive a branch-and-cut approach to solve the HLPN. In Section 5, we report the results of our extensive computational experience based on the usual hub location datasets. Finally, in Section 6 we draw some conclusions and further research on the topic.

2. Uncapacitated single-allocation hub location problem with neighborhoods

In this section, we formally introduce the UHLPN and fix the notation for the rest of the paper. We describe here the input data, the feasible solutions of the UHLPN, and the objective function of the problem that allows us to evaluate the feasible alternatives.

For the UHLPN, we are given a set of demand points, a set of potential hubs (as coordinates in $\mathbb{R}^d$), an OD flow matrix between demand points, a setup cost for opening each potential hub, a cost measure in $\mathbb{R}^d$, and for each potential facility, a neighborhood shape that represents some piece of information on the placement of the hub, and an additional cost based on the size of the neighborhood. The goal of UHLPN is to set up a hub-and-spoke network minimizing transportation, collection, distribution, and setup costs making decisions on

- How many and which hubs must be opened and the assignment pattern of demand points to hubs.
- The size of the neighborhood for each one of the open hubs. The activation cost of a neighborhood may depend on the volume of the region that it defines. Smaller neighborhoods incur smaller activation costs.
- Finding for each demand point the location of its hub server within the neighborhood where it must be served.

In what follows, we provide further details on this problem and the elements involved in its definition.

2.1. Input data

We are given an undirected connected (nonnecessarily complete) graph, $G = (N, E)$, with nodes $N = \{1, \ldots, n\}$ and set of edges E. The graph is assumed to be embedded in $\mathbb{R}^d$, in the sense that each node $i \in N$ is identified with a point $a_i \in \mathbb{R}^d$ and edges are segments connecting its two extremes in $\mathbb{R}^d$. For each pair of nodes $i, j \in N$ there is a known demand, $w_{ij} \geq 0$, which represents the amount of flow that must be sent (*routed*) from origin i to destination j. Each origin/destination pair of nodes with positive demand will be referred to as an OD pair. We denote by $W = (w_{ij})_{i,j\in N}$ the OD flow matrix, by $O_i = \sum_{\substack{j\in N:\\ \{i,j\}\in E}} w_{ij}$ the overall flow with origin at $i \in N$, and by $D_j = \sum_{\substack{i:\\ \{i,j\}\in E}} w_{ij}$ the total flow with destination at $j \in N$.

2.2. Feasible actions

Like in other hub location models, in the UHLPN a set of hubs must be selected, among the nodes in N, to be used as potential intermediate points when routing the flows associated with OD pairs. In particular, the flow associated with a given OD pair (i, j) is assumed to be *physically* routed via a *feasible path* of the form (a_i, x_k, x_m, a_j), where k and m are the indices of the open hubs associated with locations x_k and x_m, respectively, and it is possible that $k = m$ (so $x_k = x_m$). When nodes i and

j are both selected to host hubs, these paths reduce to (x_i, x_i, x_j, x_j) and consist of one single arc. Otherwise, if i is not a hub then $x_k \neq a_i$; similarly, $x_m \neq a_j$ if j is not a hub.

We assume *single allocation* of nodes to open hubs. That is, for each node $i \in N$ all the flows *leaving* and *entering* i must be routed via a single (unique) hub node, which represents the access and exit point of i to and from the distribution system.

Contrary to classical hub location models, in the UHLPN there is not a discrete set of points where hubs may be located at, even if the problem is stated on a graph with a discrete set of nodes. In particular, when node $k \in N$ is selected to host an open hub, the actual position of the hub is not known in advance and must be set within a neighborhood of the selected node k. For this, associated with each potential hub $k \in N$ we are given a *basic neighborhood*, $S_k \subseteq \mathbb{R}^d$, which is assumed to be a second-order cone (SOC) representable set containing the origin. The neighborhood set for a_k, then, is assumed to be in the form

$$\mathcal{N}_k(r) = a_k + \{r \cdot z : z \in S_k\},$$

for $r \geq 0$, that is, the a_k-translation of the r-dilation of the set S_k. The choice of r, together with the specific location of the hub, x_k, is part of the decision-making process.

Thus, we assume that feasible paths are in the form (a_i, x_k, x_m, a_j), where a_i and a_j correspond to OD pairs and x_k and x_m belong to the dilated neighborhoods of nodes k and m in the graph G. Observe that in case G is a complete graph, this assumption is no longer needed since the triangle inequality assures its verification.

In summary, for determining a feasible UHLPN solution the following decisions must be made: (1) the nodes that host the open hubs into their neighborhoods; (2) the dilation factor applied to the basic neighborhood of each selected node, as well as the actual position of its associated hub within its dilated neighborhood; and (3) the (single) allocation of nonhub nodes (*spokes*) to open hubs.

A possible choice for modeling different SOC-representable neighborhoods is by means of unit balls of norms, that is, when $S_k = \{z \in \mathbb{R}^d : \|z\| \leq 1\}$ for some norm $\|\cdot\|$ in $\mathbb{R}^d$, which belong to the class of ℓ_p-norm or polyhedral norms (Blanco et al., 2014). For instance, if the basic neighbors are Euclidean unit balls, dilations by r consist of Euclidean balls of radius r and center a_k, while for polyhedral norms, the resulting neighborhoods are dilations of symmetric polytopes. One may also consider more sophisticated shapes for the basic neighborhoods, as for instance polyellipsoids (Blanco and Puerto, 2021). We assume that for every potential hub location $k \in N$, there is an upper bound, $R_k \in \mathbb{R}_+$, on the maximum size of the dilation factor of its associated neighborhood.

In Fig. 1, we show a feasible solution of our problem. Black dots are the spoke nodes. Gray disks (ℓ_2-balls on the plane) are the dilated neighborhoods centered at the initial position of the active hub nodes (asterisks). The position on the neighborhoods of the hubs are the triangles in the picture. Finally, dotted lines represent allocation between spoke and hub nodes and the solid lines form the hub backbone network.

2.3. Costs

The cost of a feasible solution is the sum of the setup costs of the activated hubs plus the transportation costs for routing the commodities, through the allocated hubs.

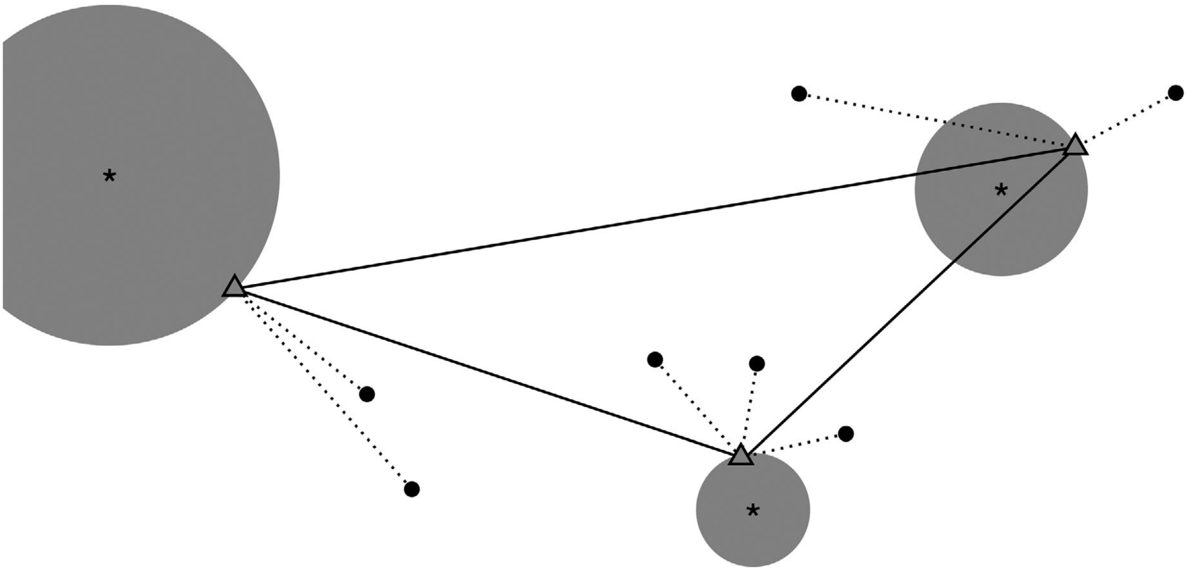

Fig. 1. Feasible solution for the UHLPN.

Setup costs: Activating a hub associated with node $k \in N$ brings a setup cost $F_k(r)$, which depends on k and the dilation factor, r, that is applied to the basic neighborhood. In particular, we consider setup costs of the form:

$$F_k(r) = f_k + g_k(r),$$

where the term f_k represents a fixed cost for establishing a hub at node k, whereas $g_k : \mathbb{R}_+ \to \mathbb{R}$ is a SOC-representable function of the dilation factor, verifying that $g_k(r) \geq 0$, for all $r \geq 0$, and $g_k(0) = 0$ (null dilations have zero cost). Some possible choices of g_k are those in which the cost of installing a hub with r-dilation increases linearly with r, being then $g_k(r) = \Lambda_k r$ for a given coefficient $\Lambda_k \geq 0$; or a polynomial function $g_k(r) = \Lambda_k r^d$, which may represent the per unit (area/volume) installation cost of a neighborhood associated with node k.

The above setup costs allow to measure, on the one hand the fixed cost of constructing the necessary infrastructure to locate a hub around the given initial position of the hub node, but also the cost of moving from such an initial position to the *border* of its neighborhood region in order to better attend the demand of the customers ($g_k(r)$).

Routing costs: Sending flow though the path (a_i, x_k, x_m, a_j) incurs a unit routing cost $\mathrm{d}^C(a_i, x_k) + \mathrm{d}^H(x_k, x_m) + \mathrm{d}^D(x_m, a_j)$, where $\mathrm{d}^C, \mathrm{d}^H, \mathrm{d}^D : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_+$ are given cost/distance functions for the collection, transportation and distribution of unitary flows, respectively . To reflect economies of scale between hub nodes, we assume that $\mathrm{d}^H(x, y) < \mathrm{d}^C(x, y)$ and $\mathrm{d}^H(x, y) < \mathrm{d}^D(x, y)$, for all $x, y \in \mathbb{R}^d$, that is, the cost of sending a unit of flow through two hub node positions (in the hub backbone network) is smaller than the one using spoke-to-hub links between the same positions. Moreover, we assume that in all cases, $\mathrm{d}^C(x, x) = \mathrm{d}^H(x, x) = \mathrm{d}^D(x, x) = 0$ for all $x \in \mathbb{R}^d$. A very interesting case that generalizes the usual practices for economies of scale in hub location problems is to consider that $\mathrm{d}^C = \mathrm{d}^D$ and $\mathrm{d}^H = \alpha \mathrm{d}^C$, for some $\alpha \in (0, 1)$ for some

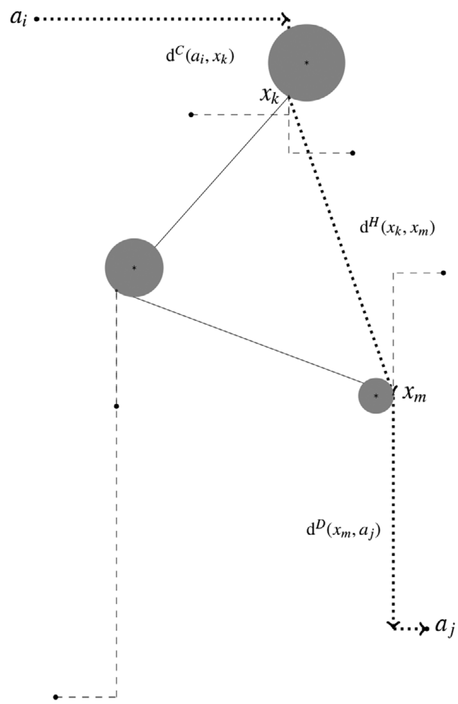

Fig. 2. Routing costs in the UHLPN.

distance measure d^C in $\mathbb{R}^d$. Also, using the relationship between different ℓ_p-norm-based distances ($\mathrm{d}_{\ell_{p_1}} < d_{\ell_{p_2}}$ whenever $p_1 > p_2 \geq 1$), one could also choose $\mathrm{d}^C = \mathrm{d}^D = \mathrm{d}_{\ell_{p_1}}$ and $\mathrm{d}^H = \mathrm{d}_{\ell_{p_2}}$ for any $p_1, p_2 \geq 1$ (here d_{ℓ_p} stands for the distance induced by the ℓ_p norm). In Fig. 2, we illustrate the different routing costs considered in the UHLPN, in a case in which $\mathrm{d}^C = \mathrm{d}^D = \mathrm{d}_{\ell_1}$ and $\mathrm{d}^H = \mathrm{d}_{\ell_2}$. The costs of routing a unit flow from node i to node j are highlighted with dotted lines (dashed lines correspond with collection and distribution connections while solid lines are hub links).

The UHLPN is to find a feasible solution of minimum total cost. Thus, the UHLPN generalizes classical hub location problems with facility setup costs where neighborhoods are not allowed (upper bound for the dilations $R = 0$), and hubs must be necessarily located at their associated points a_i, $i \in N$. These problems can also be obtained as particular cases of the UHLPN by setting to an arbitrarily large value all the coefficients Λ_k, $k \in K$, thus enforcing all the dilations to be 0.

Figure 3 shows the solution of a 10-node instance from Ernst and Krishnamoorthy (1996) in two different situations: without neighborhoods (left) and with ℓ_2-norm neighborhoods. Dashed lines represent spoke allocations to open hubs and solid lines interhub connections. Observe that the optimal design of the network changes if some flexibility is allowed on the location of the hubs.

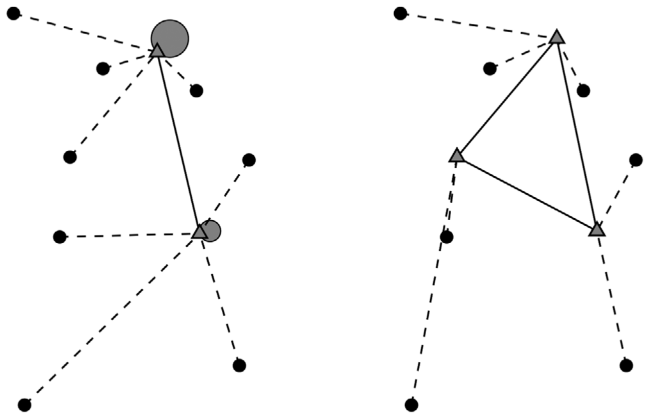

Fig. 3. Solutions of an instance of the AP dataset.

3. Mathematical programming formulations

In this section, we develop alternative mathematical programming formulations for the UHLPN. The first is an MINLP formulation, which includes bilinear and trilinear terms in the objective function. These terms come from the products of binary variables with binary and/or continuous variables. In addition, it also includes nonlinear terms in the constraints modeling the membership of hubs to their dilated neighborhoods. Then we present two reformulations. The first reformulation introduces additional decision variables in order to linearize the nonlinear terms of the original formulation, whereas the second reformulation is obtained from the aggregation of some of the decision variables introduced for the first reformulation.

3.1. Decision variables

All our formulations use the following common sets of decision variables:

- *Location-allocation variables*. They determine the nodes selected for locating hubs, as well as the allocation of nodes to open hubs. For every edge $\{i, k\} \in E$:

$$z_{ik} = \begin{cases} 1 & \text{if node } i \text{ is allocated to hub } k, \text{ for } i \neq k, \\ 0 & \text{otherwise,} \end{cases}$$

  and

$$y_k = \begin{cases} 1 & \text{if node } k \text{ is an open hub,} \\ 0 & \text{otherwise.} \end{cases}$$

- *Radii of the neighborhoods of the selected hubs*. We consider variables $r_i \in \mathbb{R}_+$, for $i = 1, \ldots, n$. Each r_i determines the dilation factor applied to the neighborhood of the potential hub node i if it were activated (and 0 otherwise).

- *Positions of hubs within the dilated neighborhoods associated with the open hubs*. The position of hub $k \in N$ is given by $x_k \in \mathbb{R}^d$.

Using the above decision variables, the routing costs can be expressed as

$$f_{\text{rout}}(z, x) = \sum_{\{i,k\} \in E} (O_i \mathrm{d}^C(a_i, x_k) + D_i \mathrm{d}^D(a_i, x_k)) z_{ik} + \sum_{k,m \in N} \sum_{\substack{i,j \in N: \\ \{i,k\},\{j,m\} \in E}} w_{ij} \mathrm{d}^H(x_k, x_m) z_{ik} z_{jm}.$$

The term $\sum_{\{i,k\} \in E} O_i \mathrm{d}^C(a_i, x_k) z_{ik}$ accounts for the collection costs induced for sending flows from nonhub nodes (spokes) to their allocated hubs. The term $\sum_{\{i,k\} \in E} D_i \mathrm{d}^D(a_i, x_k) z_{ik}$ accounts for the distribution costs due to sending flows from hubs to the (final) spokes assigned to them. The last term determines the interhub transportation costs: each flow w_{ij} associated with the OD pair (i, j) incurs an interhub cost, which depends on the hubs to which i and j are allocated. Note that the interhub transportation cost of an OD pair (i, j) in which i and j are allocated to the same hub is 0.

Furthermore, the setup costs for the activated hubs, taking into account the radii of their associated dilated neighborhoods is given by

$$f_{\text{setup}}(y, r) = \sum_{k \in N} F_k(r_k) y_k.$$

Thus, the overall objective function of the problem is

$$f(z, y, r, x) = f_{\text{rout}}(z, x) + f_{\text{setup}}(y, r).$$

A mathematical programming formulation for the problem, based on the one proposed by O'Kelly (1987) is

$$\min \ f(z, y, r, x)$$

$$\text{s.t.} \ \sum_{\substack{k \in N: \\ \{i,k\} \in E}} z_{ik} = 1, \quad \forall i \in N, \tag{1}$$

$$z_{ik} \leq y_k, \quad \forall \{i, k\} \in E, \tag{2}$$

$$x_k \in \mathcal{N}_k(r_k), \quad \forall k \in N, \tag{3}$$

$$z_{ik}, y_k \in \{0, 1\}, \quad \forall \{i, k\} \in E,$$

$$0 \leq r_k \leq R_k, \quad \forall k \in N.$$

Constraints (1) ensure that each spoke is assigned to a single hub node, while constraints (2) state that allocations are only permitted to open hubs. Finally, constraints (3) establish that the access points of assignments to hubs belong to their dilated neighborhoods. Observe that if node $k \in N$ is not selected as a hub node, its dilation factor r_k becomes zero since otherwise it would incur in a positive setup cost.

The above model is an MINLP formulation, not only because the products of z-variables in the objective function (f_{rout}) but also because the terms $g_k(r_k)y_k$ in f_{setup}, the distances between nodes and the membership to the neighborhoods (Constraint 3).

Some observations are in order concerning the r-variables and their representation in the objective function. First, observe that the nonlinear term $g_k(r_k)y_k$ appears in the objective function to represent the cost induced by the actual size of the dilated neighbourhood associated with hub k. However, in case g_k is a nondecreasing function, one can replace the term $g_k(r_k)y_k$ by just $g_k(r_k)$. This can be done because the minimization criterion and the nonnegativity of g_k guarantee that in case hub k is not open, $g_k(r_k)$ will take the smallest possible value, that is, 0. Otherwise, $y_k = 1$, so $g_k(r_k)y_k = g_k(r_k)$. Thus, for the sake of simplicity, we assume from now on that g_k is a nondecreasing function, although one could apply similar strategies for general choices of g_k.

As mentioned above, different shapes are possible for the functions g_k. In case g_k is a linear function, the overall function $f_{\text{set-up}}$ is linear. If $g_k(r) = \Lambda_k r^d$, the term r_k^d in the objective function can be adequately rewritten as a linear function by adding a new auxiliary variable and representing it as a set of SOC constraints. In particular, denoting by γ_k the dth power of r_k, that is, $\gamma_k = r_k^d$, we get that $f_{\text{setup}}(y, r) = \sum_{k\in N} f_k y_k + \sum_{k\in N} \Lambda_k \gamma_k$. Then, constraints in the form $\gamma_k \geq r_k^d$ allows us to represent such a term in the mathematical programming model as a small number of rotated SOC constraints. The next result reformulates the dth powers in the objective function by means of a set of $O(nd)$ new variables and SOC constraint.

Lemma 3.1. *Let $\alpha = (\alpha_0, \ldots, \alpha_{q-1}) \in \{0, 1\}^q$ be the coefficients of the binary decomposition of $d - 1$, that is, $d - 1 = \alpha_0 2^0 + \alpha_1 2^1 + \cdots, +\alpha_{q-1}2^{q-1}$ with $q \in \mathbb{Z}_+$ such that $2^{q-1} \leq d < 2^q$. Then, for each $k \in N$, constraint $\gamma_k \geq r_k^d$ can be equivalently represented as the following set of q SOC/linear constraints:*

$$
\begin{aligned}
r_k^2 &\leq \omega_{q-1} \cdot r_k^{(1-\alpha_{q-1})},\\
\omega_{i+1}^2 &\leq \omega_i \cdot r_k^{(1-\alpha_i)}, \ \text{for } i = 1, \ldots, q-2,\\
\omega_1^2 &\leq \gamma_k \cdot r^{(1-\alpha_0)},
\end{aligned}
$$

where $\omega_1, \ldots, \omega_{q-1}$ are nonnegative auxiliary variables.

Proof. The proof follows using Lemma 1 in Blanco et al. (2014) to represent constraint $r^{2^q} \leq \gamma r^{2^q-d}$ (which is equivalent to the one to be represented). First, the binary representations of 1, $d - 1$ and $2^q - 1$ are computed, that is,

$$
\begin{aligned}
1 &= 1 \times 2^0 + 0 \times 2^1 + \cdots + 0 \times 2^{q-1},\\
d - 1 &= \alpha_0 \times 2^0 + \alpha_1 \times 2^1 + \cdots + \alpha_{q-1} \times 2^{q-1},\\
2^q - d &= \beta_0 \times 2^0 + \beta_1 \times 2^1 + \cdots + \beta_{q-1} \times 2^{q-1}.
\end{aligned}
$$

Observe that the above decompositions allow to represent 2^q not only as $1 + \sum_{i=1}^{q-1} 2^i$ but also as

$$2^q = (1 + \alpha_0 + \beta_0) + (\alpha_1 + \beta_1) \times 2 + (\alpha_2 + \beta_2) \times 2^2 + \cdots + (\alpha_{q-1} + \beta_{q-1}) \times 2^{q-1}.$$

Following similar arguments to those explicitly detailed in Blanco et al. (2014), we get that $\alpha_i + \beta_i = 1$ for all $i = 0, \dots, q-1$. Thus, one can represent constraint $r^{2^q} \leq \gamma r^{2^q-d}$ by concatenating the q rotated SOC constraints stated in the result and then

$$\begin{aligned} r^{2^q} &\leq w_{q-1}^{2^{q-1}} r^{2^{q-1}(1-\alpha_{q-1})} \leq w_{q-2}^{2^{q-2}} r^{2^{q-1}(1-\alpha_{q-1})+2^{q-2}(1-\alpha_{q-2})} \leq \cdots \leq w_1^2 r^{2^{q-1}(1-\alpha_{q-1})+2^{q-2}(1-\alpha_{q-2})+\cdots+2(1-\alpha_1)} \\ &\leq \gamma r^{2^{q-1}(1-\alpha_{q-1})+2^{q-2}(1-\alpha_{q-2})+\cdots+2^1(1-\alpha_1)+2^0(1-\alpha_0)} = \gamma r^{2^q-1} r^{-\sum_{i=0}^{q-1}\alpha_i 2^i} = \gamma r^{2^q-1} r^{-(d-1)} \\ &= \gamma r^{2^q-d}. \end{aligned}$$

Thus, it implies that $r^d \leq \gamma$. □

Using the above result, one can easily represent volumes of the neighborhoods. For instance, for $d = 3$ and $d = 4$, the terms r_k^3 or r_k^4 can be represented by means of the following set of SOC inequalities and auxiliary variables:

$$(d=3)\begin{cases} \gamma_k r_k \geq \omega_1^2 \\ \omega_1 \geq r_k^2 \end{cases} \qquad (d=4)\begin{cases} \gamma_k \geq \omega_1^2 \\ \omega_1 \geq \omega_2^2 \\ \omega_2 r_k \geq r_k^2 \end{cases}.$$

The following result states a geometrical property of the optimal solutions of the UHLPN.

Proposition 3.1. *Optimal locations of the hubs (x-variables) must belong to the boundary of the selected neighborhoods.*

Proof. Let (x^*, r^*, z^*) be the optimal solution for UHLPN with objective value f^*. Assume that there is some $k \in N$ such that $z_{kk}^* = 1$ and $x_k^* \in \text{int}(\mathcal{N}_k(r_k^*))$. Thus, there exists $0 < \varepsilon_k < r_k^*$ such that $x_k^* \in \mathcal{N}(r_k^* - \varepsilon)$. Let us define $r' = (r_1^*, \dots, r_{k-1}^*, r_k^* - \varepsilon_k, r_{k+1}^*, \dots, r_n^*)$. Clearly, (x^*, r', z^*) is still feasible and its objective function is smaller than f^* since it only affects the term g_k and $g_k(r_k^*) \geq g_k(r_k')$, contradicting the optimality of (x^*, r^*, z^*). □

3.2. First reformulation

Next, we present a reformulation of the UHLPN described in the section above based on the linearization of the nonlinear terms of the objective function.

Indeed, the bilinear and trilinear terms in the objective function can be linearized by introducing auxiliary decision variables. In particular, for all $\{i, k\}, \{j, m\}, \{k, m\} \in E$ let

$$\eta_{ik}^C = \mathrm{d}^C(a_i, x_k) z_{ik}, \quad \eta_{ik}^D = \mathrm{d}^D(a_i, x_k) z_{ik} \text{ and } \nu_{ikjm} = \mathrm{d}^H(x_k, x_m) z_{ik} z_{jm}.$$

Using the new set of decision variables, the UHLPN reduces to

$$\min \sum_{i,k \in N} (O_i \eta_{ik}^C + D_i \eta_{ik}^D) + \sum_{i,j \in N} w_{ij} \sum_{\{i,k\},\{j,m\},\{k,m\} \in E} \nu_{ikjm} + \sum_{k \in N} f_k y_k + \sum_{k \in N} g_k(r_k)$$

s.t. (1), (2), (3),

$$\eta_{ik}^C \geq \mathrm{d}^C(a_i, x_k) - \hat{D}_{ik}^C(1 - z_{ik}), \quad \forall \{i, k\} \in E, \tag{4}$$

$$\eta_{ik}^D \geq \mathrm{d}^D(x_k, a_i) - \hat{D}_{ik}^D(1 - z_{ik}), \quad \forall \{i,k\} \in E, \tag{5}$$

$$\nu_{ikjm} \geq \mathrm{d}^H(x_k, x_m) - \hat{D}_{km}^H(2 - z_{ik} - z_{jm}), \quad \forall \{i,k\}, \{j,m\}, \{k,m\} \in E, \tag{6}$$

$$\eta_{ik}^C, \eta_{ik}^D, \nu_{ikjm} \geq 0, \quad \forall \{i,k\}, \{j,m\}, \{k,m\} \in E,$$

$$z_{ik}, y_k \in \{0,1\}, \quad \forall \{i,k\} \in E, k \in N,$$

$$0 \leq r_k \leq R_k, \quad \forall k \in N,$$

where constraints (4)–(6) are linearizations of the products $\mathrm{d}^C(a_i, x_k)z_{ik}$, $\mathrm{d}^D(a_i, x_k)z_{ik}$ and $\mathrm{d}^H(x_k, x_m)z_{ik}z_{jm}$, in which the constants $\hat{D}_{ik}^C$, $\hat{D}_{ik}^C$, and $\hat{D}_{ik}^H$ are upper bounds on the distance from nodes a_i to x_k for the collection, distribution and transportation cost functions, respectively. These bounds depend on the distances used for the cost functions. In particular, for Euclidean distances, one can choose $\hat{D}_{ik}^C = \mathrm{d}^C(a_i, a_k) + 2R_k$, $\hat{D}_{ik}^D = \mathrm{d}^D(a_i, a_k) + 2R_k$ and $\hat{D}_{km}^H = \mathrm{d}^H(a_k, a_m) + R_k + R_m$.

Observe that the set of constraints (6) can be alternatively rewritten as

$$\nu_{ikjm} \geq \mathrm{d}^H(x_k, x_m) - \hat{D}_{km}^H(1 - z_{ik}), \qquad \forall \{i,k\}, \{j,m\}, \{k,m\} \in E, \tag{7}$$

$$\nu_{ikjm} \geq \mathrm{d}^H(x_k, x_m) - \hat{D}_{km}^H(1 - z_{jm}), \qquad \forall \{i,k\}, \{j,m\}, \{k,m\} \in E. \tag{8}$$

However, the above formulation is not suitable to be solved *directly* by any of the available mixed integer SOC optimization (MISOCO) solvers (e.g., CPLEX, Gurobi, or FICO Xpress) since (4)–(6) are not rigorously speaking SOC constraints. Nevertheless, one can introduce adequate sets of nonnegative auxiliary variables for the distances, $\mathrm{d}^C(a_i, x_k)$, $\mathrm{d}^D(x_k, a_i)$ and $\mathrm{d}^H(x_k, x_m)$ in those constraints, d_{ik}^C, d_{ik}^D and d_{km}^H, respectively, for $\{i,k\}, \{k,m\} \in E$, and then rewrite (4)–(6) as

$$d_{ik}^C \geq \mathrm{d}^C(a_i, x_k), \forall \{i,k\} \in E, \tag{9}$$

$$d_{ik}^D \geq \mathrm{d}^D(a_i, x_k), \forall \{i,k\} \in E, \tag{10}$$

$$d_{km}^H \geq \mathrm{d}^H(x_k, x_m), \forall \{i,k\} \in E, \tag{11}$$

$$\eta_{ik}^C \geq \mathrm{d}_{ik}^C - \hat{D}_{ik}^C(1 - z_{ik}), \forall \{i,k\} \in E, \tag{12}$$

$$\eta_{ik}^D \geq \mathrm{d}_{ik}^D - \hat{D}_{ik}^D(1 - z_{ik}), \forall \{i,k\} \in E, \tag{13}$$

$$\nu_{ikjm} \geq \mathrm{d}_{km}^H - \hat{D}_{km}^H(2 - z_{ik} - z_{jm}), \forall \{i,k\}, \{j,m\}, \{k,m\} \in E, \tag{14}$$

$$d_{ik}^C, d_{ik}^D, d_{km}^H \geq 0, \forall \{i,k\}, \{k,m\} \in E.$$

where now, (9)–(11) are SOC-representable (see, e.g., Blanco et al., 2014) and (12)–(14) are linear constraints.

The above formulation has $O(|E|^2)$ variables, $O(|E|^2)$ linear constraints, and $O(|E|)$ nonlinear constraints.

Note that in case $\mathrm{d}^C = \mathrm{d}^D$, a single set of η-variables, instead of η^C and η^D, can be used to linearize the formulations.

In Abdinnour-Helm and Venkataramanan (1998), the authors propose a multicomodity flow based formulation for the uncapacitated hub location problem together with a branch-and-bound strategy to solve it. This formulation directly applies to our problem after introducing the following set of binary variables:

$$p_{ijkm} = \begin{cases} 1 & \text{if the arc } (k, m) \text{ is used to route the flow between } i \text{ and } j, \\ 0 & \text{otherwise,} \end{cases}$$

for $\{k, m\} \in E$.

Thus, inducing a formulation with $O(|E|^2)$ binary variables in contrast to the $O(|E|)$ binary variables in our formulation.

3.3. Second reformulation

As in the previous formulation, we consider, again, variables η^C_{ik} and η^D_{ik} to represent the products of the collection/distribution distances and the allocation variables. Now, instead of the 4-index variables ν_{ikjm}, we consider 2-index variables μ_{km}, which represent the aggregated value of the overall cost flow on a given interhub arc (k, m), that is,

$$\mu_{km} = \sum_{\substack{i,j \in N: \\ \{i,k\},\{j,m\} \in E}} w_{ij} \mathrm{d}^H(x_k, x_m) z_{ik} z_{jm},$$

for all $\{k, m\} \in E$.

The resulting equivalent reformulation is as follows:

$$\begin{aligned}
\min \; & \sum_{\{i,k\} \in E} (O_i \eta^C_{ik} + D_i \eta^D_{ik}) + \sum_{\{k,m\} \in E} \mu_{km} + \sum_{k \in N} f_k y_k + \sum_{k \in N} g_k(r_k) \\
\text{s.t. } & (1), (2), (3), (4), (5), \\
& \mu_{km} \geq \sum_{i,j \in N} w_{ij} \mathrm{d}^H(x_k, x_m) z_{ik} z_{jm}, \quad \forall \{k, m\} \in E, \qquad (15) \\
& \eta^C_{ik}, \eta^D_{ik} \geq 0, \quad \forall \{i, k\}, \{j, m\} \in E, \\
& z_{ik}, y_k \in \{0, 1\}, \quad \forall \{i, k\} \in E, k \in N, \\
& 0 \leq r_k \leq R_k, \quad \forall k \in N, \\
& \mu_{km} \geq 0, \quad \forall \{k, m\} \in E.
\end{aligned}$$

Observe that, taking into account the nonnegativity of variables and coefficients, and the minimization objective function, it is guaranteed that for optimal solutions equality will hold for constraints (15), which establishes the objective function value for the routing of interhub flows.

Note also that the trilinear terms of constraints (15) make them difficult to handle algorithmically. Next, we propose an equivalent set of linear constraints, which can be used for modeling the values of variables μ_{km}. First, in Proposition 3.2 we give a linearization of the bilinear products of z-variables.

Proposition 3.2. *The set of nonlinear constraints (15) is equivalent to the following set of inequalities:*

$$\mu_{km} \geq \mathrm{d}^H(x_k, x_m) \sum_{(i,j)\in S} w_{ij}(z_{ik} + z_{jm} - 1), \quad \forall S \subseteq N_{km}, \forall \{k, m\} \in E. \tag{16}$$

where $N_{km} = \{(i, j) \in N \times N : \{i, k\}, \{j, m\} \in E\}$.

Proof. Let $(\bar{z}, \bar{\mu}) \in \{0, 1\}^{|E|} \times \mathbb{R}_+^{|E|}$. Assume first that $(\bar{z}, \bar{\mu})$ satisfies (15). Let $\{k, m\} \in E$ and $S \subseteq N_{km}$. Then

$$\begin{aligned}
\bar{\mu}_{km} &\overset{(15)}{\geq} \mathrm{d}^H(x_k, x_m) \sum_{(i,j)\in N_{km}} w_{ij}\bar{z}_{ik}\bar{z}_{jm} = \mathrm{d}^H(x_k, x_m) \sum_{\substack{(i,j)\in N_{km}:\\ \bar{z}_{ik}=\bar{z}_{jm}=1}} w_{ij}(\bar{z}_{ik} + \bar{z}_{jm} - 1)\\
&\geq \mathrm{d}^H(x_k, x_m) \sum_{\substack{(i,j)\in S:\\ \bar{z}_{ik}=\bar{z}_{jm}=1}} w_{ij}\bar{z}_{ik}\bar{z}_{jm} \geq \mathrm{d}^H(x_k, x_m) \sum_{(i,j)\in S} w_{ij}(\bar{z}_{ik} + \bar{z}_{jm} - 1).
\end{aligned}$$

Hence, (16) is verified.

On the other hand, if $(\bar{z}, \bar{\mu})$ satisfies (16), for all $\{k, m\} \in E$ and $S \subseteq N_{km}$, then, in particular, choosing $\overline{S} = \{(i, j) \in N_{km} : \bar{z}_{ik} = \bar{z}_{km} = 1\}$, we have that

$$\begin{aligned}
\bar{\mu}_{km} &\overset{(16)}{\geq} \mathrm{d}^H(x_k, x_m) \sum_{(i,j)\in \overline{S}} w_{ij}(\bar{z}_{ik} + \bar{z}_{jm} - 1) = \mathrm{d}^H(x_k, x_m) \sum_{(i,j)\in \overline{S}} w_{ij}\\
&= \mathrm{d}^H(x_k, x_m) \sum_{(i,j)\in N_{km}} w_{ij}\bar{z}_{ik}\bar{z}_{jm},
\end{aligned}$$

so (15) is also verified. □

Observe that family (16) may have an exponential number of constraints (as in complete hub-and-spoke networks), namely $|E| \times 2^{|E|}$ inequalities, which are nonlinear because of the products $\mathrm{d}^H(x_k, x_m) \times z_{ik}$. Note that since constraint (16) is only active in case the sets S are adequately chosen as the whole pairs of users, i and j in N, linked to k and m, respectively, for $\{k, m\} \in E$ (which are unknown), one cannot avoid the use of the exponentially many constraints.

As stated in the following result, those terms can be suitably linearized by introducing additional decision variables and applying the McCormick transformation (McCormick, 1976).

Proposition 3.3. *Nonlinear constraints (16) can be replaced by the following set of inequalities:*

$$\mu_{km} \geq \sum_{(i,j)\in S} w_{ij}(\theta_{ikm} + \theta_{jmk} - \mathrm{d}^H(x_k, x_m)), \quad \forall \{k, m\} \in E, \forall S \subseteq N_{km}, \tag{17}$$

$$\theta_{ikm} \geq \mathrm{d}^H(x_k, x_m) - \hat{D}^H_{km}(1 - z_{ik}), \forall \{i, k\}, \{k, m\} \in E, \tag{18}$$

$$\theta_{ikm} \geq 0, \forall \{i, k\}, \{k, m\} \in E, \tag{19}$$

where the new variables θ_{ikm} model the routing cost of the flow through the interhub arc (k, m) with origin at i and the coefficient $\hat{D}^H_{km}$ is an upper bound on the distance between the positions of hubs k and m.

This reformulation has $O(|N||E|)$ variables, $O(|E|)$ nonlinear constraints and exponentially many linear constraints. Then, to take advantage of the smaller number of nonlinear constraints of this representation we propose an incomplete formulation, in which constraints (16) are added on-the-fly, embedded in a branch-and-cut scheme, as described in the next section.

4. Branch-and-cut solution algorithm

In this section, we develop a branch-and-cut solution algorithm for the UHLPN, based on the reformulation introduced in Section 3.3 where the routing costs of access and distribution flows are linearized via constraints (4) and (5), and the costs of interhub flows are linearized using constraints (17)–(19). Given that the set of constraints (17) is of exponential size, only a small number of them is considered initially, and the remaining ones are handled as lazy constraints, so they are only separated at the nodes of the enumeration tree where a solution is found with binary values for the z.

Initial set of constraints: We incorporate initially the following constraints in the form of (4):

$$\mu_{km} \geq w_{ij}(\theta_{ikm} + \theta_{jmk} - \mathrm{d}^H(x_k, x_m)), \quad \forall \{k, m\} \in E, \forall i, j \in N : \{i, k\}, \{j, m\} \in E,$$

that is, choosing singleton S-sets in the form $S = \{(i, j)\}$.

Lazy constraints: As we show next, once an integer feasible solution for the z-variables is found in the branch-and-bound search tree, separation can be carried out by inspection. Moreover, it is also possible to apply a more sophisticated strategy in which, instead of generating all violated constraints, only the most violated constraint is identified and added.

For all $S \subseteq N \times N$, we use the following notation in our approach:

- $S^+ = \{i \in N : \exists j \in N \text{ with } (i, j) \in S\}$,
- $S^+(i) = \{j \in N : (i, j) \in S\}$ for all $i \in S^+$,
- $S^- = \{j \in N : \exists i \in N \text{ with } (i, j) \in S\}$,
- $S^-(j) = \{i \in N : (i, j) \in S\}$ for all $j \in S^-$.
- $O_i(S) \sum_{j\in S^-(i)} w_{ij}$ is the overall flow with origin i and destination in S, for all $i \in S^+$,
- $D_j(S) \sum_{i\in S^+(j)} w_{ij}$ is the overall flow with origin in S and destination j, for all $j \in S^-$,
- $w(S) = \sum_{(i,j)\in S} w_{ij}$ is the overall flow over the connections in S.

With such a notation, for two potential hubs $k, m \in N$ with $\{k, m\} \in E$, and $S \subseteq N_{km}$, constraint (17) reads as

$$\mu_{km} \geq \sum_{i\in S^+} O_i(S)\theta_{ikm} + \sum_{j\in S^-} D_j(S)\theta_{jmk} - w(S)\mathrm{d}^H(x_k, x_m). \tag{20}$$

Algorithm 1. Branch-and-cut solution approach

Data: $\mathcal{S} = \emptyset$, `violation=true`
while `violation=true` **do**
  `violation=false`
  Solve UHLPN($\mathcal{S}$): $(\bar{z}, \bar{r}, \bar{\mu}, \bar{x})$.
  for $\{k, m\} \in E$ *with* $\bar{y}_k = \bar{y}_m = 1$ **do**
    $S_{km} = \{(i, j) \in N_{km} : \bar{z}_{ik} = \bar{z}_{jm} = 1\}$.
    if $(\{k, m\}, S_{km})$ *violates* (16) **then**
      $\mathcal{S} \to \mathcal{S} \cup \{S_{km}\}$
      `violation=true`.

In Algorithm 1 we describe the pseudo-code of the proposed solution scheme. There, we denote by UHLPN($\mathcal{S}$) the problem formulated as in Section 3.3 but where only the constraints (16) involving the sets in the pool $\mathcal{S} \subseteq \{S : S \subset N \times N\}$ are added.

Observe that to check the violation of constraint (20) for fixed $\{k, m\} \in E$ with $\bar{y}_k = \bar{y}_m = 1$ for a given feasible solution $(\bar{x}, \bar{z}, \bar{r}; \bar{\theta}, \bar{\mu})$, we consider the following choice for the S-set:

$$S = \{(i, j) \in N_{km} : \bar{z}_{ik} = \bar{z}_{jm} = 1\}.$$

Then, (20) is active only whenever $\sum_{i\in S^+} O_i(S)\bar{\theta}_{ikm} = \sum_{j\in S^-} D_j(S)\bar{\theta}_{jmk} = w(S)\mathrm{d}^H(\bar{x}_k, \bar{x}_m)$ (in that case $\bar{\mu}_{km} = w(S)\mathrm{d}^H(\bar{x}_k, \bar{x}_m)$) or if $\bar{x}_k$ and $\bar{x}_m$ coincide. Otherwise, since $\bar{\theta}_{ikm} = \mathrm{d}^H(\bar{x}_k, \bar{x}_m)\bar{z}_{ik}$ and $\bar{\theta}_{jmk} = \mathrm{d}^H(\bar{x}_k, \bar{x}_m)\bar{z}_{jm}$ one would have that $\sum_{i\in S^+} O_i(S) \neq \sum_{j\in S^-} D_j(S)$ being then the overall flow generated at the origins of S different from the overall flow with destination at the destinations of S. Furthermore, in case $\sum_{i\in S^+} O_i(S) = \sum_{j\in S^-} D_j(S)$, this flow must coincide with the overall flow given by the origins and destinations in S.

Clearly, constraint (17) with a maximum right-hand side value for the solution $(\bar{x}, \bar{z}, \bar{r}; \bar{\theta}, \bar{\mu})$, is the one associated with set S. Hence, in order to solve the separation problem one only has to check whether constraint (17) associated with S is violated, or equivalently, whether:

$$\bar{\mu}_{km} < \bar{d}_{km} w(S),$$

where $\bar{d}_{km} = \mathrm{d}^H(\bar{x}_k, \bar{x}_m)$ is the routing cost of the interhub flows through arc $(\bar{x}_k, \bar{x}_m)$.

If the above condition is not met, then constraint (17) associated with S is violated, so the following cut is added:

$$\mu_{km} \geq \sum_{i\in S^+} O_i(S)\theta_{ikm} + \sum_{j\in S^-} D_j(S)\theta_{jmk} - w(S)\mathrm{d}^H(x_k, x_m).$$

One can also easily check for the most violated inequality, that is, find $\{\bar{k}, \bar{m}\}$ in $\arg\max_{k,m\in N} \{\bar{d}_{km} w(S) - \bar{\mu}_{km}\}$ with $d_{\bar{k}\bar{m}} w(S) > \bar{\mu}_{\bar{k}\bar{m}}$.

5. Experiments

We have performed a series of experiments to test and compare the formulations. We have used the most common datasets in hub location: Australia Post (AP) (Ernst and Krishnamoorthy, 1996) and Civil Aeronautics Board (CAB) (O'Kelly, 1987) with $n \in \{10, 20\}$ in $\mathbb{R}^2$ in which the network is complete. The coordinates, a, the OD flows, w, and the setup costs, f, are taken from these instances. We have implemented the HLPN formulation with distances d^C, d^D, and d^H induced by the ℓ_1, ℓ_2, and the ℓ_∞ norms such that $\mathrm{d}^H \leq \mathrm{d}^C = d^D$, that is, the combination of norms for $(\mathrm{d}^C = \mathrm{d}^D, \mathrm{d}^H)$ in $\{(\ell_p, \ell_q) : p \leq q, p, q \in \{1, 2, \infty\}\}$. We also apply an economy of scale factor $\alpha \in \{0.2, 0.5, 0.8, 1\}$. In case $\alpha = 1$, we use only the combination of hub-to-hub distances and hub-to-spoke distances such that $\mathrm{d}^H < \mathrm{d}^D$, that is, a pair of norms in the form (ℓ_p, ℓ_q) with $p > q$.

We consider unit-ball norm-based neighborhoods. In particular, we choose as basic neighborhoods the sets $\{z \in \mathbb{R}^2 : \|z\|_p \leq 1\}$ for $p \in \{1, 2, \infty\}$. Different upper bounds have been considered for the dilations of the neighborhoods. In particular, we fix $R_k = \tau \min_{\substack{i \in N:\\ i \neq k}} \mathrm{d}^C(a_i, a_k)$ for $k \in N$, with $\tau \in \{0.25, 0.5\}$. The radius-dependent setup costs are linear and such that $f_{\text{set-up}}(y, r) = \sum_{k \in N} f_k y_k + \sum_{k \in N} \Lambda_k r_k$, where $\Lambda_k = \rho f_k$ with $\rho \in \{0.01, 0.1, 1\}$.

With these choices, for each value of n we solved 864 instances. Thus, moving n and considering the two datasets, we solved 5184 instances, each of them with the two reformulations. A time limit of two hours was set for all the problems.

The two approaches were coded in Python, and solved using Gurobi 8.0 in a Mac OSX Mojave with an Intel Core i7 processor at 3.3 GHz and 16 GB of RAM. For the branch-and-cut approach, we use the default lazy callbacks implemented in Gurobi. We denote by $(F1)$ the solution approach based on solving the compact formulation described in Section 3.2 and by $(F2)$ the branch-and-cut approach proposed in Section 3.3.

In Tables 1 and 2, we report the average results of our computational experience in the following layout:

- Measures of the required CPU time (in seconds) required for solving the problem with the two approaches: minimum (Min_1 and Min_2), mean (Av_1 and Av_2), maximum (Max_1 and Max_2), and standard deviation (Dst_1 and Dst_2) of the CPU times of the instances in the row.
- Average percentage gaps with respect to the relaxation obtained in the root node (GapR_1 and GapR_2).

Each row in Tables 1 and 2 summarizes the results obtained for the six instances in combination of the τ and ρ parameters described above.

From the results, one can observe that except in a few of the small instances, the branch-and-cut approach (F2) requires less CPU time to solve the problems for polyhedral neighborhoods (ℓ_1 and ℓ_∞ neighborhoods). For disk-shaped neighborhoods, we note that the small instances are solved in smaller times using the compact approach (F1), but as the sizes of the instances increase, the branch-and-cut performs slightly better in CPU time. While all the small ($n = 10$) instances were easily solved for ℓ_1- and ℓ_∞-norm-based neighborhoods in a few seconds with both procedures (a maximum of 42 seconds was required), the ℓ_2-norm-based neighborhoods needed much more CPU time, in particular with the branch-and-cut approach in which close to 30 minutes were needed to solve a single instance. For these small instances and polyhedral neighborhoods, the branch-and-cut

Table 1
Average CPU times for $n = 10$ instances

				ℓ_1										ℓ_2										ℓ_∞									
Dataset	α	d^C	d^H	Min$_1$	Av$_1$	Max$_1$	Dst$_1$	Min$_2$	Av$_2$	Max$_2$	Dst$_2$	GapR$_1$	GapR$_2$	Min$_1$	Av$_1$	Max$_1$	Dst$_1$	Min$_2$	Av$_2$	Max$_2$	Dst$_2$	GapR$_1$	GapR$_2$	Min$_1$	Av$_1$	Max$_1$	Dst$_1$	Min$_2$	Av$_2$	Max$_2$	Dst$_2$	GapR$_1$	GapR$_2$
AP	0.2	ℓ_1	ℓ_2	3.7	8.0	17.1	4.9	1.0	5.7	20.9	7.7	80.45	83.56	10.7	19.3	39.9	10.8	14.5	160.7	338.5	141.3	85.47	88.46	3.8	9.0	18.1	6.1	1.2	5.1	19.3	7.0	80.02	83.04
			ℓ_∞	2.2	6.6	11.8	3.3	1.3	3.2	8.3	2.7	80.38	83.50	6.0	9.7	16.2	4.0	17.8	66.4	120.8	46.6	85.42	88.43	2.9	7.1	11.7	2.9	1.5	2.5	4.3	1.1	79.95	82.98
		ℓ_2	ℓ_∞	3.2	5.7	11.6	3.2	1.3	6.5	16.8	6.1	82.64	81.71	5.6	13.0	26.0	8.6	23.3	158.7	428.0	171.4	84.76	84.52	3.1	4.7	6.8	1.6	1.2	3.0	5.3	1.6	82.55	81.62
	0.5	ℓ_1	ℓ_2	4.7	9.3	18.2	4.8	1.3	13.1	45.9	17.6	81.29	84.27	13.5	21.7	33.1	7.2	18.2	378.8	1402.5	542.5	86.10	88.96	6.4	11.0	18.6	5.0	1.3	7.5	18.5	6.5	80.96	83.84
			ℓ_∞	4.9	9.3	15.8	4.3	1.3	4.4	11.3	3.9	81.14	84.14	6.3	11.5	18.5	5.4	12.9	89.4	171.9	69.0	85.98	88.88	6.0	9.2	13.6	3.3	1.4	3.3	6.6	1.9	80.80	83.70
		ℓ_2	ℓ_∞	4.0	6.2	8.8	2.1	1.6	9.0	30.0	10.6	83.29	82.39	6.6	13.7	25.6	7.7	32.8	214.5	566.6	214.6	85.32	85.09	4.3	6.7	9.1	2.2	1.7	5.3	9.1	2.6	83.21	82.31
	0.8	ℓ_1	ℓ_2	5.9	26.3	99.7	36.2	1.9	15.0	42.5	15.3	82	84.87	21.4	45.2	100.8	29.4	37.1	507.4	1813.4	667.3	86.63	89.39	6.1	14.8	31.0	8.9	1.8	12.3	30.3	9.7	81.73	84.49
			ℓ_∞	6.2	11.1	18.2	5.0	1.9	7.2	18.1	6.2	81.80	84.70	8.8	18.2	35.4	10.9	16.9	131.9	321.9	117.9	86.47	89.27	6.6	10.3	17.6	4.4	2.0	6.3	10.5	3.6	81.51	84.30
		ℓ_2	ℓ_∞	4.6	7.4	11.8	2.9	2.2	17.9	56.1	20.5	83.81	82.94	7.5	23.6	55.9	19.2	48.8	293.6	812.3	304.7	85.80	85.57	5.1	7.9	13.5	3.4	2.2	9.7	22.5	7.5	83.73	82.86
	1	ℓ_1	ℓ_2	6.2	25.0	92.4	33.3	2.1	21.4	42.6	16.5	82.39	85.20	21.9	47.2	103.5	29.9	77.3	618.4	1765.5	665.7	86.92	89.62	7.1	29.1	100.0	35.3	2.2	25.0	55.6	20.1	82.16	84.86
			ℓ_∞	6.0	11.1	18.8	5.3	2.2	8.3	19.7	6.8	82.15	84.99	9.4	19.4	29.8	10.0	26.9	129.1	249.6	100.2	86.74	89.47	7.0	10.8	19.7	5.0	2.4	6.6	12.0	3.7	81.91	84.64
		ℓ_2	ℓ_∞	5.6	8.9	14.1	3.5	2.6	17.6	44.8	17.6	84.12	83.27	8.5	24.9	45.9	16.8	69.0	542.6	1591.4	619.5	86.07	85.85	5.4	9.0	17.5	4.7	2.5	15.0	42.1	15.1	84.03	83.17
CAB	0.2	ℓ_1	ℓ_2	2.2	3.2	3.6	0.5	0.6	0.9	1.2	0.3	52.67	60.99	4.4	6.2	8.0	1.2	3.1	4.5	7.0	1.5	66.76	78.78	3.2	4.0	4.5	0.6	0.7	0.8	0.9	0.1	52.95	60.08
			ℓ_∞	1.6	2.3	4.2	1.0	0.8	0.9	1.2	0.1	50.51	60.97	3.6	4.8	6.8	1.3	2.6	4.6	7.9	2.1	66.13	78.77	1.6	2.2	3.0	0.5	0.8	0.9	1.1	0.1	51.38	60.03
		ℓ_2	ℓ_∞	0.9	1.8	3.0	0.9	0.6	1.1	1.6	0.4	57.54	52.58	2.1	2.8	3.7	0.6	2.4	3.6	5.3	1.2	61.07	61.13	1.1	1.8	2.6	0.6	0.9	1.3	2.0	0.5	59.16	53.18
	0.5	ℓ_1	ℓ_2	2.3	3.4	4.1	0.7	0.6	0.9	1.5	0.3	53.42	61.58	4.4	6.8	11.1	2.3	3.9	7.1	10.3	2.5	67.39	79.16	3.4	4.0	4.6	0.5	0.6	0.9	1.0	0.1	54.50	61.39
			ℓ_∞	1.7	3.2	5.8	2.0	0.8	1.0	1.4	0.2	51.26	61.54	3.5	4.9	6.6	1.5	3.2	6.3	11.9	3.9	66.76	79.14	1.6	2.2	2.7	0.4	0.8	1.0	1.3	0.2	52.88	61.28
		ℓ_2	ℓ_∞	1.0	1.9	2.7	0.8	0.6	1.0	1.4	0.4	58.01	53.11	2.1	2.9	3.9	0.6	2.1	3.7	6.0	1.5	61.50	61.56	1.2	1.8	2.6	0.7	1.0	1.6	3.1	0.9	59.85	53.99
	0.8	ℓ_1	ℓ_2	3.1	3.9	5.2	0.8	0.8	1.0	1.4	0.2	54.14	62.14	5.0	7.3	12.1	2.8	4.3	13.0	29.4	10.5	67.76	79.39	4.3	4.8	5.5	0.5	0.8	1.2	1.7	0.3	55.84	62.52
			ℓ_∞	1.8	3.8	6.9	2.2	0.9	1.2	1.8	0.3	51.97	62.08	3.6	5.8	8.2	2.1	4.7	8.5	17.0	5.0	67.12	79.36	2.1	3.7	4.9	1.1	1.1	1.3	1.5	0.2	54.18	62.36
		ℓ_2	ℓ_∞	1.1	2.2	3.7	1.1	0.7	1.1	1.6	0.3	58.47	53.62	2.2	3.0	4.0	0.8	2.3	3.8	6.0	1.5	61.92	61.98	1.5	2.5	3.7	1.0	1.1	1.8	2.3	0.5	60.37	54.58
	1	ℓ_1	ℓ_2	2.5	4.0	5.5	1.1	0.8	1.3	2.1	0.5	54.58	62.49	5.1	7.9	11.4	2.7	5.0	24.9	86.4	31.3	67.99	79.52	4.1	5.1	6.8	1.0	0.8	1.3	1.6	0.3	56.57	63.12
			ℓ_∞	2.0	4.0	7.4	2.5	0.9	1.4	2.3	0.5	52.42	62.43	3.2	5.9	8.4	2.1	5.0	8.4	13.7	3.8	67.34	79.49	2.3	5.2	9.1	2.3	1.2	1.4	1.6	0.1	54.92	62.96
		ℓ_2	ℓ_∞	1.3	3.1	4.9	1.3	0.8	1.4	2.2	0.5	58.76	53.95	2.6	3.6	4.7	0.9	2.9	5.0	7.6	2.0	62.19	62.25	1.5	2.3	3.2	0.8	1.0	1.7	2.6	0.7	60.66	54.93

Table 2
Average CPU times for $n = 20$ instances

ℓ_1

Dataset	α	d^C	d^H	Min_1	Av_1	Max_1	Dst_1	Min_2	Av_2	Max_2	Dst_2	$GapR_1$	$GapR_2$
AP	0.2	ℓ_1	ℓ_2	244.1	2978.2	5754.1	2651.6	27.4	196.9	469.4	173.0	84.57	87.37
			ℓ_∞	378.6	766.3	1703.5	512.6	31.8	125.3	327.6	118.7	84.51	87.34
		ℓ_2	ℓ_∞	413.3	1633.1	4077.5	1485.5	50.2	343.9	883.4	334.3	85.99	85.86
	0.5	ℓ_1	ℓ_2	427.3	1247.7	2068.0	1160.2	342.3	1061.4	2686.7	855.3	85.30	87.86
			ℓ_∞	367.6	1400.9	4249.8	1455.4	36.4	173.4	495.3	175.5	85.07	87.81
		ℓ_2	ℓ_∞	295.3	1340.5	2790.0	1149.7	41.0	216.4	562.5	209.2	86.47	86.34
	0.8	ℓ_1	ℓ_2	446.6	3048.2	7049.9	3517.3	294.9	949.9	1493.9	506.0	85.93	88.29
			ℓ_∞	452.4	2172.5	6129.7	2182.4	100.6	475.2	1344.2	516.9	85.59	88.23
		ℓ_2	ℓ_∞	461.7	1418.6	3073.0	1124.3	51.6	646.6	2729.7	1041.3	86.90	86.77
	1	ℓ_1	ℓ_2	1545.5	1546.6	1547.6	1.5	374.5	2109.7	4641.8	1978.3	86.04	88.51
			ℓ_∞	626.5	1953.5	4653.0	1601.7	106.5	876.3	2831.7	1033.6	85.90	88.48
		ℓ_2	ℓ_∞	396.1	2003.5	4344.9	1457.4	89.7	683.1	2327.5	834.3	87.14	87.01
CAB	0.2	ℓ_1	ℓ_2	1169.3	3556.5	5360.9	1589.2	28.8	197.6	439.4	180.6	86.56	87.31
			ℓ_∞	157.7	358.9	704.6	252.2	22.9	101.5	212.1	82.4	86.50	87.25
		ℓ_2	ℓ_∞	126.9	663.5	1439.5	578.8	28.5	178.5	389.9	154.1	87.41	85.14
	0.5	ℓ_1	ℓ_2	3906.9	5379.5	6663.3	990.0	301.0	974.0	3116.1	1086.0	87.45	88.13
			ℓ_∞	210.5	654.3	1540.2	550.1	60.9	212.7	475.4	200.4	87.29	88.01
		ℓ_2	ℓ_∞	183.3	1212.4	3268.2	1274.3	55.1	495.7	1241.7	450.3	88.25	86.14
	0.8	ℓ_1	ℓ_2	3560.4	4778.3	6840.7	1426.7	499.4	1612.2	4701.9	1754.1	88.25	88.88
			ℓ_∞	285.0	1267.0	3468.0	1395.6	135.8	473.7	1025.4	389.7	88	88.68
		ℓ_2	ℓ_∞	289.5	1601.6	3812.9	1415.7	211.7	1233.3	2830.8	1182.8	88.95	86.97
	1	ℓ_1	ℓ_2	4784.9	5349.0	5913.2	797.9	1052.4	2650.5	4761.8	1907.2	88.67	89.41
			ℓ_∞	369.2	1819.1	4608.6	2073.3	274.9	1111.1	3222.8	1231.8	88.43	89.09
		ℓ_2	ℓ_∞	964.7	1759.3	3254.3	903.7	345.9	2832.4	6770.0	2948.7	89.37	87.46

ℓ_2

Dataset	α	d^C	d^H	Min_1	Av_1	Max_1	Dst_1	Min_2	Av_2	Max_2	Dst_2	$GapR_1$	$GapR_2$
AP	0.2	ℓ_1	ℓ_2	TL	TL	TL	-	823.7	2338.1	4862.8	1639.0	86.96	89.25
			ℓ_∞	778.4	2987.6	6954.0	2832.5	407.6	1163.6	2385.2	885.4	86.44	89.24
		ℓ_2	ℓ_∞	867.9	1572.0	3025.7	992.4	748.1	1070.3	1696.4	542.3	85.49	85.77
	0.5	ℓ_1	ℓ_2	TL	TL	TL		2010.6	3567.2	4693.1	1392.1	87.41	89.75
			ℓ_∞	1134.9	2286.2	3358.5	1188.6	722.4	2580.1	6553.0	2363.0	87.03	89.63
		ℓ_2	ℓ_∞	522.9	1715.8	4786.8	2052.5	644.3	1144.6	1977.2	726.0	85.97	86.06
	0.8	ℓ_1	ℓ_2	TL	TL	TL	-	6823.3	6823.3	6823.3	-	88.02	90.18
			ℓ_∞	3155.3	4493.5	6091.5	1208.5	2203.5	3378.2	6012.2	1770.1	87.43	90
		ℓ_2	ℓ_∞	937.5	2191.3	4873.4	1827.1	1728.7	4331.2	7163.1	2724.4	86.39	86.58
	1	ℓ_1	ℓ_2	TL	TL	TL	-	6090.3	6589.8	7089.2	706.3	88.26	90.40
			ℓ_∞	1921.2	4698.7	5934.3	1888.8	5056.5	5476.9	5879.0	411.6	87.67	90.25
		ℓ_2	ℓ_∞	1182.6	2342.1	5713.2	2247.6	4038.0	5375.7	6713.4	1891.8	86.65	87.05
CAB	0.2	ℓ_1	ℓ_2	4091.2	4794.6	5361.1	645.9	367.7	930.0	1640.2	465.9	92.80	94.41
			ℓ_∞	343.9	1029.8	2430.1	853.4	211.2	500.3	973.7	319.5	92.75	94.39
		ℓ_2	ℓ_∞	290.6	1061.5	2348.3	998.9	282.7	599.5	1191.5	359.8	88.19	86.41
	0.5	ℓ_1	ℓ_2	6090.8	6090.8	6090.8	TL	1656.9	4266.9	7198.6	2276.7	93.29	94.79
			ℓ_∞	436.2	2180.2	5239.9	2330.7	435.8	1298.6	2468.2	864.2	93.18	94.72
		ℓ_2	ℓ_∞	582.3	2213.6	4342.1	1926.6	759.6	2203.8	4910.3	1861.8	88.96	87.35
	0.8	ℓ_1	ℓ_2	7045.2	7045.2	7045.2	-	6602.7	6602.7	6602.7	-	93.80	95.29
			ℓ_∞	698.7	1619.3	4199.5	1720.6	1295.9	3391.8	6489.9	2368.6	93.57	95.02
		ℓ_2	ℓ_∞	1041.8	2992.7	6977.6	2727.9	4300.1	4878.3	5456.4	817.7	89.63	88.20
	1	ℓ_1	ℓ_2	TL	TL	TL	-	TL	TL	TL	-	94.02	-
			ℓ_∞	823.5	2661.0	6830.1	2807.7	3262.5	3567.6	4222.6	441.0	93.81	95.20
		ℓ_2	ℓ_∞	2220.6	3100.3	3985.1	882.3	6687.3	6687.3	6687.3	TL	90.03	88.74

ℓ_∞

Dataset	α	d^C	d^H	Min_1	Av_1	Max_1	Dst_1	Min_2	Av_2	Max_2	Dst_2	$GapR_1$	$GapR_2$
AP	0.2	ℓ_1	ℓ_2	245.9	2949.5	5611.1	2479.0	23.0	153.8	299.8	127.8	84.03	86.93
			ℓ_∞	366.4	572.7	872.5	219.0	22.7	85.1	222.4	78.2	83.97	86.89
		ℓ_2	ℓ_∞	388.1	1382.0	3105.6	1230.7	45.0	148.6	419.0	141.1	85.74	85.48
	0.5	ℓ_1	ℓ_2	322.0	2512.6	5771.3	2877.4	85.8	429.1	757.5	252.0	84.77	87.47
			ℓ_∞	353.2	1034.3	2572.2	830.7	43.9	136.8	296.6	105.1	84.58	87.39
		ℓ_2	ℓ_∞	250.6	1084.9	2203.1	891.4	36.7	178.7	452.0	162.6	86.25	86.00
	0.8	ℓ_1	ℓ_2	519.7	1149.0	1778.3	890.0	281.9	1224.3	2593.3	786.8	85.33	87.92
			ℓ_∞	469.9	1691.7	4167.5	1468.7	77.3	225.6	462.7	158.9	85.11	87.82
		ℓ_2	ℓ_∞	261.6	1286.4	2956.6	1073.1	33.5	550.9	2080.4	768.8	86.72	86.48
	1	ℓ_1	ℓ_2	1960.3	2029.8	2099.3	98.3	536.2	2174.8	5616.0	1847.2	85.64	88.17
			ℓ_∞	519.8	2079.2	4631.4	1538.4	127.4	426.1	1224.4	423.9	85.41	88.07
		ℓ_2	ℓ_∞	422.3	1321.8	2514.8	981.8	58.8	637.0	1896.3	684.5	86.98	86.74
CAB	0.2	ℓ_1	ℓ_2	1442.5	4351.4	6844.6	1984.6	26.2	268.9	677.2	278.7	87.51	88.42
			ℓ_∞	141.3	372.6	701.4	220.5	19.2	94.9	221.5	84.3	87.42	88.37
		ℓ_2	ℓ_∞	139.6	389.8	1051.3	345.3	23.7	141.7	296.5	106.2	88.38	85.57
	0.5	ℓ_1	ℓ_2	2639.1	4456.0	6273.0	2569.5	310.2	1171.4	2970.0	1137.0	88.36	89.19
			ℓ_∞	232.5	673.5	1488.3	559.8	53.1	214.2	543.7	202.7	88.19	89.07
		ℓ_2	ℓ_∞	168.0	690.4	1371.5	517.4	58.5	348.9	562.3	193.9	89.16	86.56
	0.8	ℓ_1	ℓ_2	5190.2	5796.9	6250.6	450.9	599.9	2626.7	7127.1	2693.7	89.11	89.86
			ℓ_∞	270.4	1382.6	3717.4	1503.9	144.0	523.1	1158.1	465.3	88.85	89.68
		ℓ_2	ℓ_∞	232.3	1338.4	2653.0	899.6	263.1	1157.6	2144.0	642.9	89.84	87.41
	1	ℓ_1	ℓ_2	7113.8	7143.9	7174.0	42.6	1097.4	2300.9	3632.6	1084.2	89.52	90.31
			ℓ_∞	422.1	1968.4	4814.6	2041.4	203.1	982.1	2079.4	786.3	89.24	90.05
		ℓ_2	ℓ_∞	658.5	1926.6	3678.1	1126.4	503.2	2588.6	6239.7	2150.0	90.23	87.89

Table 3
Percentage of unsolved instances and MIP gaps for the $n = 20$ instances

				ℓ_1				ℓ_2				ℓ_∞			
Dataset	α	d^C	d^H	$MipGAP_1$	US_1	$MipGAP_2$	US_2	$MipGAP_1$	US_1	$MipGAP_2$	US_2	$MipGAP_1$	US_1	$MipGAP_2$	US_2
AP	0.2	ℓ_1	ℓ_2	1.96	16.67	-	-	13.81	100	-	-	3.29	16.67	-	-
		ℓ_2	ℓ_∞	-	-	-	-	-	-	10.95	25	-	-	-	-
	0.5	ℓ_1	ℓ_2	10.15	66.67	-	-	18.66	100	3.42	50	5.63	50	-	-
			ℓ_∞	-	-	-	-	2.09	33.33	-	-	-	-	-	-
		ℓ_2	ℓ_∞	-	-	-	-	-	-	8.63	25	-	-	-	-
	0.8	ℓ_1	ℓ_2	7.92	50	0.13	16.67	22.82	100	6.34	83.33	11.42	66.67	-	-
			ℓ_∞	-	-	-	-	2.90	33.33	2.14	33.33	-	-	-	-
		ℓ_2	ℓ_∞	-	-	-	-	-	-	9.41	25	-	-	-	-
	1	ℓ_1	ℓ_2	7.35	66.67	0.99	16.67	24.85	100	8.20	66.67	10.47	66.67	-	-
			ℓ_∞	0.60	16.67	-	-	3.57	33.33	4.97	50	-	-	-	-
		ℓ_2	ℓ_∞	-	-	-	-	-	-	12.55	50	-	-	-	-
	Average AP			2.33	18.06	0.09	2.78	8.32	46.88	4.95	34.38	2.57	16.67	-	-
CAB	0.2	ℓ_1	ℓ_2	-	16.67	-	-	4.89	50	-	-	2.39	16.67	-	-
	0.5	ℓ_1	ℓ_2	1.89	16.67	-	-	9.22	83.33	1.33	33.33	5.24	66.67	-	-
		ℓ_2	ℓ_∞	-	-	-	-	-	-	1.68	20	-	-	-	-
	0.8	ℓ_1	ℓ_2	5.26	33.33	1.34	16.67	14.39	83.33	11.73	83.33	7.18	33.33	1.05	16.67
			ℓ_∞	-	-	-	-	2.23	33.33	-	-	-	-	-	-
		ℓ_2	ℓ_∞	-	-	-	-	0.61	20	5.97	60	-	-	-	-
	1	ℓ_1	ℓ_2	5.62	66.67	3.67	50	18.32	100	30.35	100	6.48	66.67	4.18	33.33
			ℓ_∞	-	-	-	-	2.90	33.33	2.81	33.33	-	-	-	-
		ℓ_2	ℓ_∞	0.81	16.67	-	-	3.30	40	8.83	80	-	-	-	-
	Average CAB			1.13	12.50	0.42	5.56	4.87	38.24	5.29	33.82	1.78	15.28	0.44	4.17
Average				1.73	15.28	**0.26**	**4.17**	6.54	42.42	**5.12**	**34.09**	2.17	15.97	**0.22**	**2.08**

Table 4
Average number of cuts

			$n = 10$						$n = 20$					
			AP			CAB			AP			CAB		
			Neighborhoods											
α	d^C	d^H	ℓ_1	ℓ_2	ℓ_∞	ℓ_1	ℓ_2	ℓ_∞	ℓ_1	ℓ_2	ℓ_∞	ℓ_1	ℓ_2	ℓ_∞
0.2	ℓ_1	ℓ_2	128	103	130	37	34	34	137	59	109	240	225	267
		ℓ_∞	134	185	135	34	49	33	132	140	116	357	353	350
	ℓ_2	ℓ_∞	124	79	107	17	30	26	78	44	60	259	160	220
0.5	ℓ_1	ℓ_2	179	151	172	31	34	35	204	127	174	323	459	503
		ℓ_∞	156	226	158	36	50	39	208	263	211	439	560	502
	ℓ_2	ℓ_∞	194	130	166	17	30	27	163	75	131	322	418	382
0.8	ℓ_1	ℓ_2	201	206	199	31	46	41	353	172	252	637	486	785
		ℓ_∞	200	289	190	44	61	54	354	435	241	781	826	838
	ℓ_2	ℓ_∞	226	174	195	19	27	29	254	150	196	570	725	695
1	ℓ_1	ℓ_2	201	194	215	36	49	40	376	141	327	972	463	891
		ℓ_∞	164	254	191	48	55	53	478	636	331	935	1324	1116
	ℓ_2	ℓ_∞	257	235	205	18	26	24	272	208	256	772	598	931

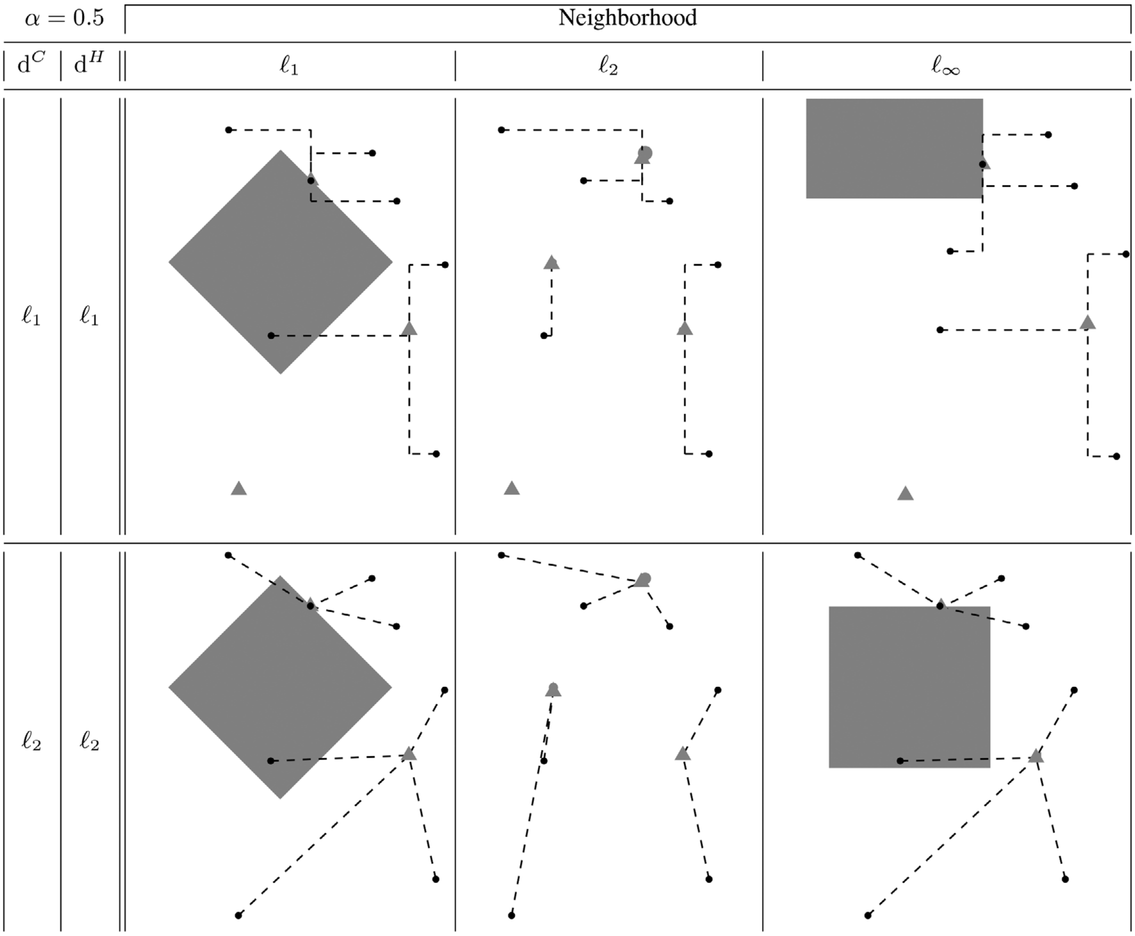

Fig. 4. Solutions for the same AP ($n = 10$) instance and different costs and neighborhood shapes.

approach required half of the time for solving the problems than the compact approach. However, for disk-shaped neighborhoods, the situation is the opposite. For the $n = 20$ instances, the branch and cut outperforms, in average, for any type of neighborhood, the compact approach in terms of the CPU times needed for solving the problem. For polyhedral neighborhoods, the compact approach need more than four times more CPU time to solve the problem than the branch and cut, and for Euclidean neighborhoods, the compact formulation needed twice the CPU time used by the branch and cut. This fact clearly justifies the use of formulation (F2) for solving the HLPN problem.

Regarding the gap with respect to the relaxation of the root node, we obtain similar results with both formulations, with slightly higher (less than 2% in average) bounds for the compact formulation (F1). Note that one may expect that (F2) is, at the root node, much weaker than (F1) since only a few cuts have been added at that node.

Furthermore, in Table 3 we report the results concerning the unsolved instances, for any of the formulations (F1) or (F2):

- Average percentage of MIP gaps (MIPGap_1 and MIPGap_2) obtained within the time limit.
- Percentage of unsolved instances with the two approaches: US_1 and US_2.

As can be seen from the average results shown in the table (where we have highlighted in boldface the best global average results for all the instances), the branch-and-cut approach was able to solve more instances than the compact formulation. The solutions obtained for the instances for which none of the approaches was able to certify optimality, the best obtained solutions were also better with the branch-and-cut approach.

Finally, in order to analyze the performance of the branch-and-cut approach, we show in Table 4 the number of cuts needed to solve the problem with such an algorithm. The first observation that comes from the results is that the number of cuts is small. Note that the complete formulation of the branch-and-cut approach requires an exponential (in $|E| = n^2$) number constraints. However, on average, the overall number of cuts is rather small (103 cuts for the $n = 10$ instances and 401 for the $n = 20$ instances). Moreover, although the shape of the neighborhoods seems to affect the difficulty of the problem, the number of cuts is similar for the three different neighborhoods considered in these experiments. Finally, we remark that the CAB dataset with $n = 10$ nodes requires, significantly, much less cuts than the AP dataset for the same number of points.

Finally, in Fig. 4 we draw some of the solutions for the AP dataset for $n = 10$ and different choices for the costs d^C and d^H and neighborhoods shapes. Observe that both the shapes of the neighborhoods and the distance-based costs affect the optimal position of the hubs and the allocation pattern of the hub-and-spoke network. In particular, in case that the neighborhood shapes are based on the ℓ_2-norm only two facilities are required, whereas for other shapes three facilities seem to be necessary.

6. Conclusions

We analyze in this paper, a new version of the UHLPFC where geographical flexibility is allowed for locating the hub nodes. We propose a general framework for the problem by modeling the geographical flexibility using neighborhood SOC-representable regions around the original positions of the nodes and measuring distribution, collection, and transportation costs by means of ℓ_p-norm distances. We propose an MINLP formulation for the problem and we provide two different formulations that make it tractable using commercial MISOCO solvers. The first formulation consists of a linearization of some bilinear and trilinear terms while the second one is based on introducing a novel set of exponentially many constraints for which an efficient separation oracle and a branch-and-cut approach is presented.

Future research on the topic includes the incorporation of preferences onto the neighborhoods allowing the decision maker to determine his most favorite regions. Also, some different models of hub location as those with incomplete backbone networks or covering objective functions can be extended to be analyzed with neighborhoods.

Acknowledgments

The authors were partially supported by research groups SEJ-584 and FQM-331 (Junta de Andalucía) and projects MTM2016-74983-C02-01 (Spanish Ministry of Education and Science/FEDER), FEDER-US-1256951, P18-FR-1422, P18-FR-2369 (Junta de Andalucía), CEI-3-FQM331 (Andalucía Tech), and *NetmeetData* (Fundación BBVA – Big Data 2019). We also would like to acknowledge Elena Fernández (Universidad de Cádiz) for her useful and detailed comments on previous versions of this manuscript.

References

Abdinnour-Helm, S., 1998. A hybrid heuristic for the uncapacitated hub location problem. *European Journal of Operational Research* 106, 2, 489–499.

Abdinnour-Helm, S., Venkataramanan, M., 1998. Solution approaches to hub location problems. *Annals of Operations Research* 78, 31–50.

Alumur, S., Kara, B.Y., 2008. Network hub location problems: the state of the art. *European Journal of Operational Research* 190, 1, 1–21.

Alumur, S.A., Campbell, J.F., Contreras, I., Kara, B.Y., Marianov, V., O'Kelly, M.E., 2021. Perspectives on modeling hub location problems. *European Journal of Operational Research* 291, 1–17.

Alumur, S.A., Nickel, S., Rohrbeck, B., da Gama, F.S., 2018. Modeling congestion and service time in hub location problems. *Applied Mathematical Modelling* 55, 13–32.

Aykin, T., 1988. On the location of hub facilities. *Transportation Science* 22, 2, 155–157.

Aykin, T., 1995. The hub location and routing problem. *European Journal of Operational Research* 83, 1, 200–219.

Aykin, T., Brown, G.F., 1992. Interacting new facilities and location-allocation problems. *Transportation Science* 26, 3, 212.

Blanco, V., 2019. Ordered p-median problems with neighbourhoods. *Computational Optimization and Applications* 73, 2, 603–645.

Blanco, V., Fernández, E., Puerto, J., 2017. Minimum spanning trees with neighborhoods: mathematical programming formulations and solution methods. *European Journal of Operational Research* 262, 3, 863–878.

Blanco, V., Marín, A., 2019. Upgrading nodes in tree-shaped hub location. *Computers & Operations Research* 102, 75–90.

Blanco, V., Puerto, J., 2021. Covering problems with polyellipsoids: a location analysis perspective. *European Journal of Operational Research* 1, 16, 44–58.

Blanco, V., Puerto, J., El Haj, S., 2014. Revisiting several problems and algorithms in continuous location with ℓ_τ-norms. *Computational Optimization and Applications* 58, 3, 563–595.

Brimberg, J., Mišković, S., Todosijević, R., Uroševic, D., 2020. The uncapacitated r-allocation p-hub center problem. *International Transactions in Operational Research*, https://doi.org/10.1111/itor.12801.

Campbell, J.F., O'Kelly, M.E., 2012. Twenty-five years of hub location research. *Transportation Science* 46, 2, 153–169.

Carrabs, F., Cerrone, C., Cerulli, R., Gaudioso, M., 2017. A novel discretization scheme for the close enough traveling salesman problem. *Computers & Operations Research* 78, 163–171.

Contreras, I., Fernández, E., Marín, A., 2009. Tight bounds from a path based formulation for the tree of hub location problem. *Computers & Operations Research* 36, 12, 3117–3127.

Contreras, I., Fernández, E., Marín, A., 2010. The tree of hubs location problem. *European Journal of Operational Research* 202, 2, 390–400.

Contreras, I., O'Kelly, M., 2019. Hub location problems. In Laporte, G., Nickel, S., Saldanha da Gama, F. (eds) *Location Science*. Springer, Cham, pp. 327–363.

Disser, Y., Mihalák, M., Montanari, S., Widmayer, P., 2014. Rectilinear shortest path and rectilinear minimum spanning tree with neighborhoods. International Symposium on Combinatorial Optimization. Springer, Heidelberg, pp. 208–220.

Dorrigiv, R., Fraser, R., He, M., Kamali, S., Kawamura, A., López-Ortiz, A., Seco, D., 2015. On minimum-and maximum-weight minimum spanning trees with neighborhoods. *Theory of Computing Systems* 56, 1, 220–250.

Ernst, A., Jiang, H., Krishanmoorthy, M., Baatar, D., 2017. Reformulations and computational results for the uncapacitated single allocation hub covering problem. *Data and Decision Sciences in Action* 1, 1, 133–148.

Ernst, A.T., Krishnamoorthy, M., 1996. Efficient algorithms for the uncapacitated single allocation p-hub median problem. *Location science* 4, 3, 139–154.

Farahani, R.Z., Hekmatfar, M., Arabani, A.B., Nikbakhsh, E., 2013. Hub location problems: a review of models, classification, solution techniques, and applications. *Computers & Industrial Engineering* 64, 4, 1096–1109.

Gentilini, I., Margot, F., Shimada, K., 2013. The travelling salesman problem with neighbourhoods: minlp solution. *Optimization Methods and Software* 28, 2, 364–378.

Habibzadeh Boukani, F., Farhang Moghaddam, B., Pishvaee, M.S., 2016. Robust optimization approach to capacitated single and multiple allocation hub location problems. *Computational and Applied Mathematics* 35, 1, 45–60.

Hoff, A., Peiró, J., Corberán, Á., Martí, R., 2017. Heuristics for the capacitated modular hub location problem. *Computers & Operations Research* 86, 94–109.

Labbé, M., Yaman, H., 2004. Projecting the flow variables for hub location problems. *Networks* 44, 2, 84–93.

Löffler, M., van Kreveld, M., 2010. Largest and smallest convex hulls for imprecise points. *Algorithmica* 56, 2, 235–269.

Martins de Sá, E., Contreras, I., Cordeau, J.F., Saraiva de Camargo, R., de Miranda, G., 2015. The hub line location problem. *Transportation Science* 49, 3, 500–518.

McCormick, G.P., 1976. Computability of global solutions to factorable nonconvex programs: part I—convex underestimating problems. *Mathematical programming* 10, 1, 147–175.

O'Kelly, M.E., 1986a. Activity levels at hub facilities in interacting networks. *Geographical Analysis* 18, 4, 343–356.

O'Kelly, M.E., 1986b. The location of interacting hub facilities. *Transportation science* 20, 2, 92–106.

O'Kelly, M.E., 1987. A quadratic integer program for the location of interacting hub facilities. *European Journal of Operational Research* 32, 3, 393–404.

O'Kelly, M.E., 1992. Hub facility location with fixed costs. *Papers in Regional Science* 71, 3, 293–306.

Peiró, J., Corberán, Á., Laguna, M., Martí, R., 2018. Models and solution methods for the uncapacitated r-allocation p-hub equitable center problem. *International Transactions in Operational Research* 25, 4, 1241–1267.

Peiró, J., Corberán, Á., Martí, R., Saldanha-da Gama, F., 2019. Heuristic solutions for a class of stochastic uncapacitated p-hub median problems. *Transportation Science* 53, 4, 1126–1149.

Puerto, J., Ramos, A., Rodríguez-Chía, A.M., 2011. Single-allocation ordered median hub location problems. *Computers & Operations Research* 38, 2, 559–570.

Puerto, J., Ramos, A., Rodríguez-Chía, A.M., 2013. A specialized branch & bound & cut for single-allocation ordered median hub location problems. *Discrete Applied Mathematics* 161, 16–17, 2624–2646.

Puerto, J., Ramos, A., Rodríguez-Chía, A.M., Sánchez-Gil, M.C., 2016. Ordered median hub location problems with capacity constraints. *Transportation Research Part C: Emerging Technologies* 70, 142–156.

Puerto, J., Valverde, C., 2020. Routing for unmanned aerial vehicles: touring dimensional sets. https://arxiv.org/abs/2012.02547.

Tanash, M., Contreras, I., Vidyarthi, N., 2017. An exact algorithm for the modular hub location problem with single assignments. *Computers & Operations Research* 85, 32–44.

Topcuoglu, H., Corut, F., Ermis, M., Yilmaz, G., 2005. Solving the uncapacitated hub location problem using genetic algorithms. *Computers & Operations Research* 32, 4, 967–984.

Váňa, P., Faigl, J., 2015. On the dubins traveling salesman problem with neighborhoods. 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, Piscataway, NJ, pp. 4029–4034.

Yang, Z., Xiao, M.Q., Ge, Y.W., Feng, D.L., Zhang, L., Song, H.F., Tang, X.L., 2018. A double-loop hybrid algorithm for the traveling salesman problem with arbitrary neighbourhoods. *European Journal of Operational Research* 265, 1, 65–80.