European Journal of Operational Research 318 (2024) 768–777

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Discrete optimization

Some new clique inequalities in four-index hub location models

Mercedes Landete [a], Juanjo Peiró [b,*]

[a] Departamento de Estadística, Matemáticas e Informática. Instituto Centro de Investigación Operativa, Universidad Miguel Hernández de Elche, Spain
[b] Departament d'Estadística i Investigació Operativa. Facultat de Ciències Matemàtiques, Universitat de València, Spain

ARTICLE INFO

Keywords:
Location
Combinatorial optimization
Set packing
Conflict graph
Branch and cut

ABSTRACT

Hub location problems can be modeled in several ways, one of which is the path-based family of models that make use of four-index variables. Clique inequalities are frequently used to describe solution characteristics for optimization problems with binary variables. In this study, new valid inequalities of the clique type are introduced for the path-based family of hub location models. Some of their properties and corresponding lifting are discussed, and a separation heuristic algorithm is proposed. We also report a set of computational experiments to evaluate the usefulness of the proposals when embedded in a commercial solver.

1. Introduction

Hub location problems (HLPs), core problems in Location Science (see Drezner and Hamacher 2002, Eiselt and Marianov 2011, 2015, and Laporte, Nickel, and da Gama 2019), arise in transportation when it is desirable to consolidate and disseminate flows at certain centralized locations known as *hubs*. In these problems, These problems require finding ideal locations of one or more hub facilities for: (a) switching, sorting, and connecting flows that enter hubs via one link and depart via another link, allowing redirection of flows with fewer links than would be required with direct connections; and (b) consolidating flows to be aggregated and disaggregated in an attempt to reduce transportation costs by taking advantage of economies of scale (see, e.g., Campbell 2013, Kara and Tansel 2000, 2003, O'Kelly, Campbell, de Camargo, and de Miranda 2015). Applications of hub location models classically arise in passenger travel and commodity (cargo) deliveries, as well as in computer and telecommunications network settings. Lately, these models are also being applied for solving public transportation planning and humanitarian logistic problems. A detailed list of up-to-date applications of hub location models can be found in Alumur, Campbell, Contreras, Kara, Marianov, and O'Kelly (2021).

The basics of a hub location problem can be described in the following way: There is a set V of n nodes. Associated with any pair of these nodes, say $i, j \in V$, there is a non-negative amount of flow t_{ij}, also called *traffic*, to be transported between them. Some of the nodes in V can act as hub facilities and need to be determined (*located*) as part of the optimization process. To transport the flows, there is a set of induced arcs between the nodes in V that may be used for this purpose, so flows are allowed to go through hub facilities or directly from the origin nodes to the destination nodes, although there are advantages (in terms of incentives) in routing flows via hubs. The generic objective is to optimize a function that depends on the location of hub facilities and the routing of flows. In this way, several intertwined optimization decisions need to be made: selecting the nodes that should become hubs, allocating nodes to hubs, and selecting the traffic routes between nodes through the hub network. These challenging problems are computationally classified as $\mathcal{NP}$-hard (see Ernst, Hamacher, Jiang, Krishnamoorthy, and Woeginger 2009, Kara and Tansel 2000, 2003, Sohn and Park 2000).

There is a vast literature dealing with the taxonomy and analysis of different hub location problems. Some reviews and surveys on this topic are those by Alumur and Kara (2008), Campbell and O'Kelly (2012), Farahani, Hekmatfar, Arabani, and Nikbakhsh (2013) and, more recently, Alumur (2019) and Contreras and O'Kelly (2019). According to many of them, three-index formulations, widely referred to as *flow-based* formulations (see Ernst and Krishnamoorthy 1996, 1998) and four-index formulations, also referred to as *path-based* formulations (see Campbell 1992, 1994), are among the most used. In this study, we will focus on the latter, as they can easily be applied to various allocation criteria such as the single and multiple patterns, as well as for center, equitable, and median objectives, and also allow different cost expressions to be taken into account in the optimization decision, such as inter-hub fixed costs.

We now present a basic formulation to illustrate the structure of a hub location problem in which the path-based model is used and introduce the four-index variables and characteristics. In this model, we use two typical families of decision variables in Location Science

* Corresponding author.
E-mail addresses: landete@umh.es (M. Landete), juanjo.peiro@uv.es (J. Peiró).

https://doi.org/10.1016/j.ejor.2024.06.008
Received 22 July 2023; Accepted 6 June 2024
Available online 10 June 2024

problems: location–allocation variables, denoted by y_{ik} for $i,k \in V$, and path variables x_{ijkm}, for $i,j,k,m \in V$, with the following meaning:

- Variable $y_{ik} = \begin{cases} 1, & \text{if node } i \text{ can use hub node } k \text{ to route some of its traffic,} \\ 0, & \text{otherwise,} \end{cases}$
  where $y_{kk} = 1$ implies that k has been selected as hub node for the system.
- Variable $x_{ijkm} = \begin{cases} 1, & \text{if the traffic traveling from } i \text{ to } j \text{ traverses hubs } k \text{ and } m \text{ in this order,} \\ 0, & \text{otherwise,} \end{cases}$
  where $x_{ijim} = 1$ ($x_{ijkj} = 1$) implies i (j) is a hub of itself, and $x_{ijii} = 1$ ($x_{ijjj} = 1$) implies that the flow between i and j traverses only hub i (j). In these x-variables, the pair of indices (i,j) is referred to as the origin–destination (OD) pair of a path, whereas pair (k,m) is known as the transshipment hub pair. Note that variable x_{ijkm} does not necessarily take the same value as x_{jimk}, i.e., symmetry is not assumed.

With these variables, a standard formulation for the hub location problem with basic characteristics consists of:

$$\begin{aligned}
\min \quad & \text{a cost function,} \\
\text{s.t.:} \quad & \sum_{k\in V}\sum_{m\in V} x_{ijkm} = 1 && i,j \in V, && (1)\\
& y_{ik} \le y_{kk} && i,k \in V, && (2)\\
& x_{ijkm} \le y_{ik} && i,j,k,m \in V, && (3)\\
& x_{ijkm} \le y_{jm} && i,j,k,m \in V, && (4)\\
& x_{ijkm}, y_{jm} \in \mathcal{F} && i,j,k,m \in V, && (5)\\
& x_{ijkm} \in \{0,1\} && i,j,k,m \in V, && (6)\\
& y_{ik} \in \{0,1\} && i,k \in V. && (7)
\end{aligned}$$

The objective is to minimize a generic cost function, for instance, the sum of costs for the transportation of traffic or the maximum of these costs. This function may also include other aspects, e.g., those related to installing/opening hubs for the network. Equalities (1) guarantee that all the traffic between nodes is routed. Constraints (2) ensure that if node i is allocated to node k, then k acts as a hub node. Likewise, constraints (3) and (4) ensure that if the traffic to be sent from i to j is routed via hubs k and m, then i has been allocated to hub k and j to hub m. Expression (5), which we state generically, includes in $\mathcal{F}$ all additional constraints that may be needed to complete the description of the feasible region of each particular HLP, e.g., a fixed number of hubs to be opened, different allocation strategies, etc. Finally, constraints (6) and (7) define the domain of the decision variables.

Path-based models usually assume two non-naive considerations: the transportation costs must satisfy the triangle inequality and the hub-network is complete. However, if any of them fails, these models still can be considered general if $\mathcal{F}$ includes adequate constraints for limiting the number of hubs when there is no triangle inequality (see Marín, Cánovas, and Landete 2006 for details) and for restricting the hub connections when the network is not complete. Hereinafter, we suppose that $\mathcal{F}$ takes these aspects into account. Hence, a solution to a four-index hub location problem is a set of paths connecting nodes i and j for all $i,j \in V$ in such a way that each path traverses one or two hubs. More precisely:

(a) If an origin node and a destination node are not hubs, the flow must go through one or two intermediate hubs that are different from the origin and the destination.
(b) If either the origin or the destination is a hub, then flow between them can either be sent directly or through an additional hub.
(c) If both are hubs, the flow must go directly from the origin to the destination.

It is a well-known fact that the linear relaxation of hub location models typically results in it being weak, hence it is useful to derive valid inequalities that may cut off fractional solutions of such relaxations. In this paper, we are interested in deriving incompatibility conditions in the form of valid inequalities to provide better descriptions of the feasible regions of four-index hub location problems.

We recall that two generic binary variables, say a and b, are said to be *incompatible* (or *in conflict*) if at most one can take the value 1 in a feasible solution, i.e., $a + b \le 1$. Clique inequalities (Edmonds, 1962) are a general class of 0–1 inequalities that state incompatibilities between binary variables. They were formally introduced as a class of facet-defining inequalities for the set-packing problem in Padberg (1973). It is well known that any clique inequality of a set-packing problem can be written by grouping in a single inequality a subset of pairwise incompatible variables. There is an extensive literature in which these inequalities are studied and deployed for modeling and solving various combinatorial optimization problems. In hub location models, inequalities of the form

$$\sum_{i\in V}\sum_{j\in V}\sum_{k\in V}\sum_{m\in V} a_{ijkm}x_{ijkm} \le 1,$$

with $a_{ijkm} \in \{0,1\}$ for all $i,j,k,m \in V$, are clique inequalities with x-variables. Works such as Marín (2005a, 2005b), Marín et al. (2006), and Corberán, Landete, Peiró, and Saldanha-da Gama (2019) present comprehensive studies of large sets of clique inequalities that improve the polyhedral description of hub location models, while others such as Calvete, Domínguez, Galé, Labbé, and Marín (2019), Letchford and Vu (2021), Marín and Pelegrín (2019), and Marzi, Rossi, and Smriglio (2019) address a variety of optimization problems by taking advantage of such inequalities in other contexts.

Our contribution

In this paper, we introduce new families of clique inequalities with four-index x-variables for path-based hub location models. The number of different OD pairs in clique inequalities for HLPs known until now is three. We prove that this number reaches a maximum of ten, and introduce a new family of clique inequalities with ten OD pairs. We use sequential-lifting theory to strengthen the new inequalities. These theoretical results are presented in Section 2. Then, we propose an algorithm to separate the new family of strengthened clique inequalities heuristically. We test the usefulness of our proposals when embedded in a commercial solver by performing a small set of computational experiments in Section 3. We finish the paper with some conclusions.

2. New clique inequalities for hub location models

A classical family of constraints in hub location is Eqs. (1), which state that, for any i and j, all the flow originating at i whose destination is j has to be transported. Typically, since the objective function of HLPs minimizes some costs, these constraints are in the form of equality. However, if this function is expressed in terms of profits, equalities (1) can be re-written as

$$\sum_{k\in V}\sum_{m\in V} x_{ijkm} \le 1 \qquad i,j \in V. \tag{8}$$

Apart from these, other clique inequalities with x-variables have been previously proposed in the literature for HLPs (see Corberán et al. 2019, Marín 2005b, Marín et al. 2006), e.g.:

$$x_{abij} + x_{abji} + \sum_{k,m\in V\,:\,(k,m)\neq(i,j)} x_{ijkm} \le 1 \qquad i,j,a,b \in V, i \neq j, (a,b) \neq (i,j), \tag{9}$$

$$x_{abij} + x_{abji} + x_{jijj} + \sum_{k\in V}\sum_{m\in V\,:\,m\neq j} x_{ijkm} \le 1 \qquad i,j,a,b \in V\,:\, i \neq j. \tag{10}$$

All these inequalities contain x-variables with some distinct OD pairs. Specifically, inequalities (8), (9) and (10) involve one, two and three different OD pairs, respectively: in the first case, only variables with the OD pair (i,j) come into play; in the second, only OD pairs (a,b) and (i,j) are involved; in the third, only the pairs (a,b), (j,i) and (i,j). To the best of our knowledge, the maximum number of OD pairs found in the literature on clique inequalities with four-index x-variables for HLP is three. We aim to introduce new families of clique inequalities that contain x-variables with as many different OD pairs as possible, ideally, more than three.

Notation 1. *From now on, let $*$ denote a generic element of set V.*

Proposition 1 (*Gathering results from Marín et al. 2006 and Corberán et al. 2019*). *In four-index hub location models, given $i,j,k,m \in V$, a generic x_{ijkm} variable is incompatible with any of the following x-variables:*

- x_{ij**}
- x_{km**}, $k \neq m \wedge (i,j) \neq (k,m)$, *excluding* x_{kmkm}
- x_{mk**}, $k \neq m \wedge (i,j) \neq (k,m)$, *excluding* x_{mkmk}
- x_{**ii}, $i \neq k$
- x_{**jj}, $j \neq m$
- x_{**ij}, $i \neq j \wedge (i,j) \neq (k,m)$
- x_{**ji}, $i \neq j \wedge (i,j) \neq (k,m)$
- x_{k***}, $k = m$, *excluding* x_{k*k*}
- x_{*k**}, $k = m$, *excluding* x_{*k*k}.

Assumption 1. We will assume throughout the paper that all elements in any subset of V are distinct, unless otherwise stated.

Observe now the following property of any feasible solutions for HLPs: if an OD pair, say i and j, uses hub nodes k and m that are different, respectively, from i and j, then no other OD pair can use either i or j as hubs. Next we state this property in terms of incompatibilities of x-variables.

Proposition 2. *Let $\{i_1,j_1,k_1,m_1\}$ and $\{i_2,j_2,k_2,m_2\} \subseteq V$ be two sets such that $\{i_1,j_1\} \neq \{i_2,j_2\}$. Suppose that these sets constitute the ordered set of indices of two x-variables, $x_{i_1j_1k_1m_1}$ and $x_{i_2j_2k_2m_2}$, respectively. If the OD pair of one of these variables, say $\{i_1,j_1\}$, and the transshipment hub pair of the other variable, say $\{k_2,m_2\}$, have at least one element in common, i.e., if $|\{i_1,j_1\} \cap \{k_2,m_2\}| \geq 1$, then these two x-variables are incompatible.*

Proof. Suppose $|\{i_1,j_1\} \cap \{k_2,m_2\}| \geq 1$. This implies $(i_1 = k_2) \vee (j_1 = k_2) \vee (i_1 = m_2) \vee (j_1 = m_2)$. If $i_1 = k_2$, then $j_1 \neq k_2$ and $i_1 \neq m_2$ by Assumption 1, and we only need to check whether $j_1 = m_2$ holds. Suppose $j_1 = m_2$, then the two variables become $x_{i_1j_1k_1m_1}$ and $x_{i_2j_2i_1j_1}$. In that case, $x_{i_1j_1k_1m_1} = 1$ implies that neither i_1 nor j_1 can be hubs, thus $x_{i_2j_2i_1j_1} = 0$, and $x_{i_2j_2i_1j_1} = 1$ implies that i_1 and j_1 are hubs, then they should be hubs of themselves, thus $x_{i_1j_1k_1m_1} = 0$, making these two variables incompatible. Now, suppose $j_1 \neq m_2$. A similar reasoning applies again but only for i_1, which also makes the two variables incompatible.

The cases where $j_1 = k_2$, $i_1 = m_2$, or $j_1 = m_2$ hold, which need to be checked to complete the proof, are analogous. □

Fig. 1 serves as an illustration of the process of proving Proposition 2. Any node in this figure represents a unique x-variable: an arbitrary variable x_{ijkm} is represented by the node at the kth row and mth column of the ith row and the jth column of blocks. To see the incompatibilities arising from Proposition 2, we can choose any variable with four different indices, say x_{1234} (marked as a black square in the figure), and verify that the gray nodes correspond to all the incompatibilities of x_{1234} stated in Proposition 2.

Proposition 2 extends Proposition 1: all the incompatibilities in Proposition 1 except those in the first bullet are included, among others, in Proposition 2. Hence, Proposition 2 allows us to derive new sets of valid inequalities for hub location problems that use four-index x-variables. For example, given $\{1,2,3,4\} \subseteq V$, we can apply this result with variables x_{1234} and x_{3412} and derive inequality $x_{1234} + x_{3412} \leq 1$. Nonetheless, we can add more x-variables that satisfy the conditions of Proposition 2 to this inequality and obtain $x_{1234} + x_{2341} + x_{3412} + x_{4123} + x_{1324} + x_{2413} \leq 1$. This example is now generalized.

Corollary 1. *Let $\{i,j,k,m\} \subseteq V$. Inequality*

$$x_{ijkm} + x_{jkmi} + x_{kmij} + x_{mijk} + x_{ikjm} + x_{jmik} \leq 1 \tag{11}$$

is a valid clique inequality for four-index HLP models.

Proof. The proof is straightforward since, as seen in Proposition 2, any of the six x-variables in the inequality is incompatible with all the others. □

The incompatible variables resulting from Propositions 1 and 2 induces a so-called *conflict* graph. This is an auxiliary graph that contains a node associated with every binary variable of a problem, and an edge between every pair of nodes corresponding to incompatible variables. For example, Fig. 1 shows the nodes of the conflict graph for an HLP when $|V| = 4$. It is well known (see, e.g., Borndörfer and Weismantel 2000, Cánovas, Landete, and Marín 2000, Conforti, Cornuéjols, and Rao 1995, Johnson and Padberg 1982, Nemhauser and Trotter 1974, Padberg 1973, Trotter 1975) that some sub-graphs of a conflict graph, for instance those of the clique type, induce facets of the set packing polytope. We recall that, in an undirected graph, a *clique* is a subset of nodes such that every two distinct nodes in the subset are adjacent (i.e., a complete subgraph). Any clique in a conflict graph induces an incompatibility condition between all its nodes that can be mathematically expressed as the summation of the corresponding variables (nodes in the clique) less than or equal to 1. Accordingly, by analyzing the conflict graph derived from all the incompatibility constraints of a hub location model we may render new conditions to be satisfied by the feasible solutions of HLP models with four-index x-variables.

Notation 2. *Let G be the conflict graph defined by the incompatibilities between x-variables in HLPs. Let Q be a clique in G with the maximum number of different OD pairs and with at most one x-variable for each OD pair.*

Definition 1. The *neighborhood* of a node x_{ijkm} in G, denoted by $N(x_{ijkm})$, is the set of nodes in G connected to x_{ijkm} with an edge.

Lemma 1. *In all x-variables in Q all four indices are different.*

Proof. We prove this by showing that an x-variable with repeated indices results in a potential Q with a smaller cardinality than that resulting from an x-variable with four different indices. To do so, it is sufficient to check that $N(x_{ijkk}) \subsetneq N(x_{ijkm})$, since x_{mabc} with $a,b,c \notin \{i,j,k,m\}$ belongs to $N(x_{ijkm})$ and not to $N(x_{ijkk})$. □

Lemma 2. *There are no two x-variables with the same transshipment hub pair in Q, i.e., if $x_{ijkm}, x_{abcd} \in Q$, then $\{k,m\} \neq \{c,d\}$.*

Proof. The proof proceeds by assuming $\{k,m\} = \{c,d\}$ and derives a contradiction. Suppose that $\{k,m\} = \{c,d\}$, so the two variables would become $x_{ijkm}, x_{abkm} \in Q$. As they belong to Q, they are connected by sharing at least one of the indices in the OD pair of one of them with one of the indices of the transhipment pair of the other, so:

- $\{i,j\} \cap \{k,m\} \neq \emptyset$, which contradicts Lemma 1 since x_{ijkm} would not have four different indices,

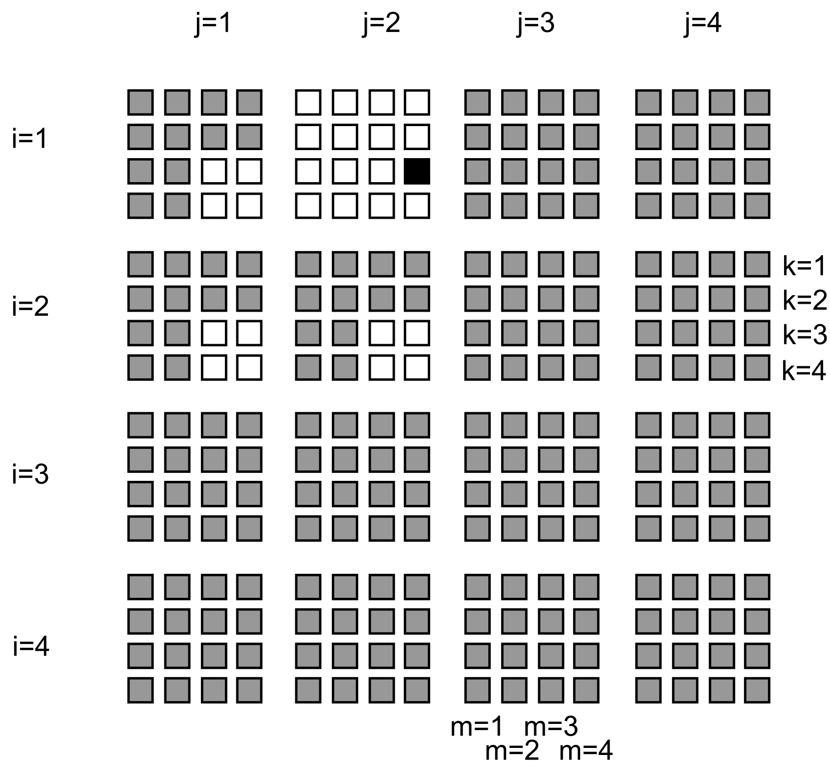

Fig. 1. Illustration of the incompatibilities in Proposition 2.

- $\{a,b\} \cap \{k,m\} \neq \emptyset$, which also contradicts Lemma 1 since x_{abkm} would not have four different indices. □

Continuing with the previous discussion, it is interesting to note that inequalities (11) contain six different OD pairs and, thus, are more diverse than any other clique inequality previously presented in the academic literature for four-index hub location models. In fact, $\binom{4}{2} = 6$ is the number of different pairs that can be chosen from a list of four elements. It would be of interest to know whether, by considering $\ell > 4$ pairwise disjoint indices in V, the maximum number of different OD pairs in a clique inequality is also $\binom{\ell}{2}$ or smaller (it cannot be larger).

In order to answer such a question, an optimization model could be solved. Let B be the set of all combinations of two elements of V, i.e., $B = \{\tau : \tau = (i,j), i,j \in V, i \neq j\}$. Let θ_{ts} be a parameter that indicates whether t and s in B share an element in the following form:

$$\theta_{(i_1,j_1)(i_2,j_2)} = \begin{cases} 1, & \text{if } i_1 \text{ or } j_1 \in \{i_2, j_2\} \\ 0, & \text{otherwise.} \end{cases}$$

For instance, $\theta_{(1,2)(3,4)} = 0$, $\theta_{(1,3)(3,4)} = 1$ and, obviously, $\theta_{ts} = \theta_{st}$. Let $z_{(i_1,j_1)(i_2j_2)}$ be a binary variable that indicates whether any of the x-variables $x_{i_1j_1i_2j_2}$, $x_{i_1j_1j_2i_2}$, $x_{j_1i_1i_2j_2}$ or $x_{j_1i_1j_2i_2}$ is in a clique inequality or not. The problem of computing the maximum number of distinct OD pairs possibly involved in a valid clique inequality can be modeled as the optimization problem of computing the maximum number of z-variables that can simultaneously take value one. It can be stated as:

$$(\mathrm{P}_1) \quad \max \sum_{t \in B} \sum_{s \in B : \theta_{ts} = 0} z_{ts}, \tag{12}$$

$$\text{s.t.: } z_{ts} + z_{uv} \leq 1 \qquad t,s,u,v \in B : (t,s) \neq (u,v), \theta_{tv} = \theta_{su} = 0, \tag{13}$$

$$z_{ts} \in \{0,1\} \qquad t,s \in B : \theta_{ts} = 0. \tag{14}$$

In this model, constraints (13) state the incompatibilities of the x-variables: variables x_{ts} and x_{uv} can belong to the same clique inequality, i.e., are incompatible, only if $\theta_{tv} = 1$ or $\theta_{su} = 1$. Constraints (14) state that the four indices of the x-variable in the clique inequality are pairwise disjoint as we have explained in Lemma 1.

Table 1
Optimal values of (P_1) for different $|V|$.

$\lvert V\rvert$	4	5	6	7	8	9	10	11	12	13	14	15
$v((\mathrm{P}_1))$	6	10	10	10	10	10	10	10	10	10	10	10

Notice that (P_1) is a model for solving the well-known maximum clique problem (MCP) in a graph $\mathcal{G}(\mathrm{P}_1)$ in which there is a node associated with each z-variable and an edge between each pair of variables z_{ts} and z_{uv} such that $\theta_{tv} + \theta_{su} \geq 1$. In particular, $\mathcal{G}(\mathrm{P}_1)$ has the same incidence degree of all nodes and, to the best of our knowledge, it does not belong to any special class of graphs for which the MCP can be solved in polynomial time: it is not bipartite, because its nodes are not decomposed into two disjoint sets, nor a web or an anti-web, as can be easily verified by drawing a small example, nor a planar graph, nor a perfect graph. Despite these facts, we can use (P_1) to compute—by brute force using standard integer programming techniques—the maximum number of different OD pairs that a clique inequality with x-variables may contain for a given value of $|V|$. Table 1 summarizes, for several values of $|V|$, the objective value of (P_1), which we denote by $v((\mathrm{P}_1))$.

As can be seen from Table 1, if $|V| = 4$, then $v((\mathrm{P}_1)) = 6$, which corroborates the discussion after Corollary 1. Interestingly, $v((\mathrm{P}_1))$ equals 10 for all other values of $|V|$ that we tested. We prove later that, for $|V| > 4$, ten is the maximum number of OD pairs that can appear together in a valid clique inequality for four-index HLP models, providing a better upper bound for the specific MCP given by (P_1).

Corollary 2. *Inequalities*

$$\sum_{t \in B} z_{ts} \leq 1 \qquad s \in B \tag{15}$$

and

$$\sum_{s \in B} z_{ts} \leq 1 \qquad t \in B \tag{16}$$

are valid for (P_1).

Proof. Inequalities (15) are a direct implication from Lemma 2, which states that if $x_{ijkm} = 1$, then $x_{abkm} = 0$ for all i, j, k, m, a, b, i.e., $x_{ijkm} + x_{abkm} \leq 1$. Inequalities (16) follow from the fact that (P_1) maximizes the diversification of OD pairs. Both families of inequalities follow from constraints (13). □

Lemma 3. *If $x_{ijkm} \in Q$, then the elements in the family x_{ji**} do not belong to Q.*

Proof. The proof follows from Lemma 1. An element in x_{ij**} cannot be connected to an element in x_{ji**} unless some x-variable has a repeated index, contradicting Lemma 1. □

Proposition 3. *For $|V| > 4$, the maximum number of different OD pairs in x-variables that belong to Q is 10.*

Proof. We will proceed by checking that the optimal value of (P_1) cannot be greater than 10.

From model (P_1), we know that z_{ts} and z_{uv} are incompatible if $(t, s) \neq (u, v)$ and $\theta_{tv} = \theta_{su} = 0$. Complementarily, we will say that they are *compatible* if they are not incompatible. Let z^* be an optimal solution of (P_1). W.l.o.g., let us suppose that $z^*_{(i,j)(k,m)} = 1$. Hence, the optimal value of (P_1) will be the maximum number of variables that are compatible with $z_{(i,j)(k,m)}$ as well as being pairwise compatible. We need to show that they are at most nine.

All compatible variables with $z_{(i,j)(k,m)}$ are those of the form z_{st} such that $i \in t$ or $j \in t$ or $k \in s$ or $m \in s$. The continuation of the proof consists of getting a maximal set of compatible variables by iteratively adding compatible variables to the set $Q = \{z_{(i,j)(k,m)}\}$. We do so by, first, adding to Q all z_{st} variables with $j \in t$ that are pairwise compatible; second, adding variables with $k \in s$ that are pairwise compatible with variables in the updated Q; third, adding those with $i \in t$; and, finally, adding those with $m \in s$, always pairwise compatible.

Let $Q = Q \cup \{z_{(a,b)(c,j)}\}$. All variables with $j \in t$ that are compatible with the elements in $Q = \{z_{(i,j)(k,m)}, z_{(a,b)(c,j)}\}$ are of the form $z_{(*,*)(a,j)}$, $z_{(*,*)(b,j)}$ and $z_{(c,*)(*,j)}$. Since they must also be pairwise compatible, $z_{(*,*)(a,j)}$ becomes $z_{(b,d)(a,j)}$, $z_{(*,*)(b,j)}$ becomes $z_{(e,f)(b,j)}$, and $z_{(c,*)(*,j)}$ can be $z_{(a,c)(b,j)}$ or $z_{(b,c)(d,j)}$. These two cases are distinguished below:

- If $Q = \{z_{(i,j)(k,m)}, z_{(a,b)(c,j)}, z_{(b,d)(a,j)}, z_{(e,f)(b,j)}, z_{(a,c)(b,j)}\}$, there are three variables with $k \in s$ that are compatible with all the variables in Q and also pairwise compatible: $z_{(j,k)(b,e)}$, $z_{(e,k)(b,f)}$ and $z_{(b,k)(a,d)}$. Thus, we update Q by setting $Q = \{z_{(i,j)(k,m)}, z_{(a,b)(c,j)}, z_{(b,d)(a,j)}, z_{(e,f)(b,j)}, z_{(a,c)(b,j)}, z_{(j,k)(b,e)}, z_{(e,k)(b,f)}, z_{(b,k)(a,d)}\}$. Finally, there are two variables with $i \in t$ or $m \in s$ compatible with all the variables in Q and pairwise compatible, namely $z_{(b,j)(i,k)}$ and $z_{(j,m)(b,k)}$, and the maximal set of compatible variables becomes $Q = \{z_{(i,j)(k,m)}, z_{(a,b)(c,j)}, z_{(b,d)(a,j)}, z_{(e,f)(b,j)}, z_{(a,c)(b,j)}, z_{(j,k)(b,e)}, z_{(e,k)(b,f)}, z_{(b,k)(a,d)}, z_{(b,j)(i,k)}, z_{(j,m)(b,k)}\}$.
- If $Q = \{z_{(i,j)(k,m)}, z_{(a,b)(c,j)}, z_{(b,d)(a,j)}, z_{(e,f)(b,j)}, z_{(b,c)(d,j)}\}$, there are three variables with $k \in s$ that are compatible with all the variables in Q and also pairwise compatible: $z_{(j,k)(f,g)}$, $z_{(f,k)(b,e)}$ and $z_{(g,k)(b,f)}$. Thus, we update Q by setting $Q = \{z_{(i,j)(k,m)}, z_{(a,b)(c,j)}, z_{(b,d)(a,j)}, z_{(e,f)(b,j)}, z_{(b,c)(d,j)}, z_{(j,k)(f,g)}, z_{(f,k)(b,e)}, z_{(g,k)(b,f)}\}$. Finally, there are two variables with $i \in t$ or $m \in s$ compatible with all the variables in Q and pairwise compatible, namely $z_{(h,j)(i,k)}$ and $z_{(j,m)(h,k)}$, and the maximal set of compatible variables becomes $Q = \{z_{(i,j)(k,m)}, z_{(a,b)(c,j)}, z_{(b,d)(a,j)}, z_{(e,f)(b,j)}, z_{(b,c)(d,j)}, z_{(j,k)(f,g)}, z_{(f,k)(b,e)}, z_{(g,k)(b,f)}, z_{(h,j)(i,k)}, z_{(j,m)(h,k)}\}$.

In both cases, the number of elements in Q is ten. □

Fig. 2 helps to illustrate the initial proof steps of Proposition 3. Here, as in Fig. 1, each node represents a unique x-variable. In the first step of the proof an arbitrary variable, say x_{1356}, is chosen (thus, fixing $i = 1$, $j = 3$, $k = 5$ and $m = 6$). All the incompatible variables resulting from this selection are marked in gray. Among them, we next choose x_{2643} (fixing $a = 2$, $b = 6$, and $c = 4$), whose incompatible variables are the nodes marked with thick lines. The next variable to be chosen must belong to the incompatibility intersection of the two sets of incompatibilities mentioned before. We choose x_{2435}. Repeating this process, we sequentially choose x_{2536}, x_{2315}, x_{3546}, x_{3625}, x_{4526}, x_{4623} and x_{5624}. All these variables are marked with an "x" inside their corresponding node in the figure.

Notice that while $\mathcal{G}(\mathrm{P}_1)$ may not be a member of the classes of graphs known for which the MCP is solvable in polynomial time, by Proposition 3 we now know that the MCP on $\mathcal{G}(\mathrm{P}_1)$ is also easy.

Corollary 3. *The number of elements in V that are indices of the x-variables in Q with ten different OD pairs is greater than or equal to five and less than or equal to twelve.*

Proof. The result follows from case $z_{(c,*)(*,j)} = z_{(b,c)(d,j)}$ in the proof of Proposition 3. The twelve indices are $\{a, b, c, d, e, f, g, h, i, j, k, m\}$ and the corresponding elements of Q are: $z_{(i,j)(k,m)}$, $z_{(a,b)(c,j)}$, $z_{(b,d)(a,j)}$, $z_{(e,f)(b,j)}$, $z_{(b,c)(d,j)}$, $z_{(j,k)(f,g)}$, $z_{(f,k)(b,e)}$, $z_{(g,k)(b,f)}$, $z_{(h,j)(i,k)}$ and $z_{(j,m)(h,k)}$. If some of the indices coincide, then the number of elements in V can be reduced by up to five. Assuming that $V = \{1, 2, 3, \ldots\}$, some x-cliques with ten different OD pairs and $5, 6, 7, \ldots, 12$ different indices, respectively, are induced by the following variables:

- Five indices: x_{1234}, x_{1325}, x_{1435}, x_{1523}, x_{2345}, x_{2413}, x_{2514}, x_{3415}, x_{3524}, x_{4512}.
- Six indices: x_{1356}, x_{2315}, x_{2435}, x_{2536}, x_{2634}, x_{3546}, x_{3625}, x_{4526}, x_{4623}, x_{5624}.
- Seven indices: x_{1267}, x_{1324}, x_{1736}, x_{2317}, x_{2635}, x_{3427}, x_{3715}, x_{5637}, x_{5716}, x_{6723}.
- Eight indices: x_{1423}, x_{1634}, x_{2436}, x_{2613}, x_{2846}, x_{3478}, x_{3648}, x_{3768}, x_{3825}, x_{5812}.
- Nine indices: x_{1379}, x_{1758}, x_{2538}, x_{2837}, x_{3617}, x_{3714}, x_{3967}, x_{4523}, x_{4715}, x_{7834}.
- Ten indices: $x_{1,5,6,10}$, $x_{1,6,8,10}$, $x_{1,8,5,10}$, $x_{1,10,3,5}$, x_{2413}, x_{2514}, x_{3915}, $x_{3,10,5,9}$, x_{5712}, $x_{5,10,27}$.
- Eleven indices: $x_{1,10,3,8}$, $x_{1,11,2,10}$, $x_{2,11,9,10}$, $x_{3,10,4,8}$, $x_{4,10,1,8}$, $x_{5,8,6,11}$, $x_{6,7,10,11}$, $x_{6,8,7,11}$, $x_{8,10,5,6}$, $x_{9,11,1,10}$.
- Twelve indices: $x_{1,11,2,9}$, $x_{2,3,10,12}$, $x_{2,4,3,12}$, $x_{2,11,3,4}$, $x_{3,10,11,12}$, $x_{5,12,7,11}$, $x_{6,12,5,11}$, $x_{7,12,6,11}$, $x_{8,11,1,2}$, $x_{9,11,2,8}$. □

Corollary 4. *Let $\{a, b, c, d, e, f, g, h, i, j, k, m\} \subseteq V$ be twelve different elements in V. Then, inequality*

$$x_{akbi} + x_{bcjm} + x_{bdcm} + x_{bkcd} + x_{cjkm} + x_{emgk} + x_{fmek} + x_{gmfk} + x_{hkab} + x_{ikbh} \leq 1 \tag{17}$$

is a valid clique inequality for four-index HLP models.

Proof. Direct from Proposition 3. □

Note that permuting the first two indices or the second two indices of any of the variables in Inequality (17) also results in a valid inequality.

Illustrative example

To illustrate the potential usefulness of the new clique inequalities in Corollaries 1 and 4, we show in the next example that they can cut off non-optimal solutions allowed by the clique inequalities known for four-index HLP models.

Example 1. Consider the following model:

$\max\ x_{1234} + x_{2345} + x_{3412} + x_{4123} + x_{1324} + x_{2413}$

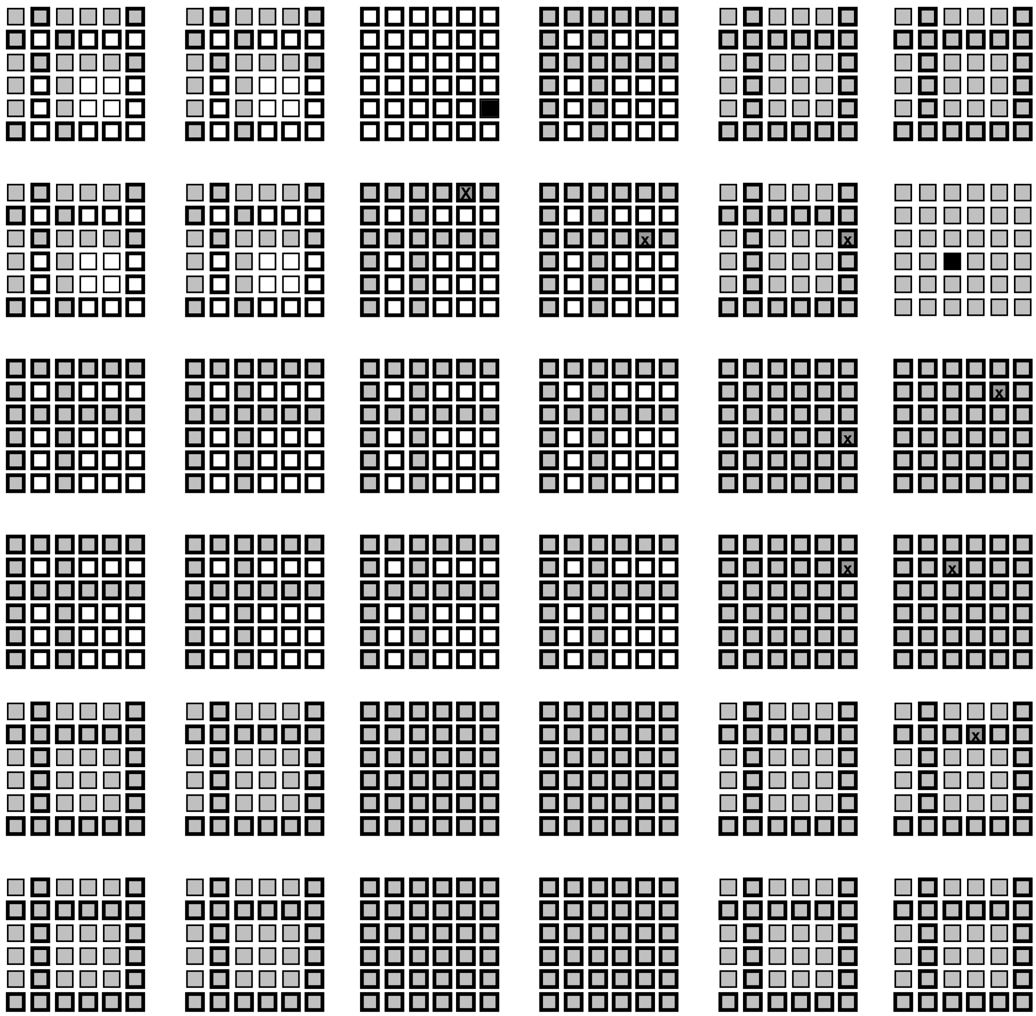

Fig. 2. Illustration of the incompatibilities in the proof of Proposition 3.

s.t.: (2),

x-set packing constraints,

$$x_{iiij} = x_{iiji} = 0 \qquad i,j \in V \,:\, j \neq i,$$
$$x_{ijki} = x_{jiik} = 0 \qquad i,j,k \in V \,:\, j,k \neq i,$$
$$0 \leq x_{ijkm} \leq 1 \qquad i,j,k,m \in V,$$

where "x-set packing constraints" are all the previously known clique inequalities for HLPs with x-variables (stated in Corberán et al. 2019, Proposition 9 after removing their y_{ij}- and y_{ii}-variables).

This model has an optimal solution that violates the clique inequality induced by its objective function, $x_{1234} + x_{2345} + x_{3412} + x_{4123} + x_{1324} + x_{2413} \leq 1$, while satisfying all the previous clique inequalities known so far: variables x_{2345}, x_{3412} and x_{2413} were not said to be incompatible, although it was already known that variable x_{2345} was incompatible with x_{4123}, variable x_{3412} was incompatible with x_{1234}, and variable x_{2413} with x_{1324}. The optimal value of the model is 3 for all $|V| \geq 5$, which corresponds to solution $x_{2345} = x_{3412} = x_{2413} = 1$, $x_{1234} = x_{4123} = x_{1324} = 0$. If we replace the objective function by $x_{1,2,3,5} + x_{7,8,1,9} + x_{10,14,1,8} + x_{11,8,1,7} + x_{9,15,1,10} + x_{9,12,1,14} + x_{9,4,1,11} + x_{1,6,2,13} + x_{1,16,2,6} + x_{1,13,2,16}$, the new problem objective value is 10, which indicates that none of the variables in this sum were incompatible in terms of Proposition 1.

Strengthening the new valid inequalities

Since our purpose in the previous sections was to diversify (maximize) the number of OD pairs, the obtained clique inequalities have at most one variable with a positive coefficient for each OD pair. However, there may be other x-variables with OD pairs already present in the inequality that are pairwise incompatible with those in Q and, thus, could be added to Q in order to strengthen the induced inequality. In other words, inequality

$$\sum_{x_{ijkm} \in Q} x_{ijkm} \leq 1, \tag{18}$$

where Q is a clique under the assumptions in Notation 2, can be strengthened if there is a variable x_{abcd} such that it is incompatible with all those variables in Q.

Sequential lifting is a well-known technique for strengthening valid inequalities in set packing models (see Padberg 1973 for more details). Its application in our context consists of checking if other x-variables with OD pairs already present in (18) can be added (i.e., if their coefficients can be lifted from 0 to 1). Applying this technique as explained in Padberg (1973) for strengthening the clique inequalities introduced above, two results follow.

Proposition 4. *Inequality*

$$x_{ijkm} + x_{jkmi} + x_{kmij} + x_{mijk} + x_{ikjm} + x_{jmik} +$$
$$x_{ijmk} + x_{jkim} + x_{kmji} + x_{mikj} + x_{ikmj} + x_{jmki} +$$
$$x_{ijkk} + x_{jkmm} + x_{kmii} + x_{mijj} \qquad \leq 1.$$

is a sequential lifting of Inequality (11) in Corollary 1.

Proof. From Corollary 1, we know that clique inequality $x_{ijkm} + x_{jkmi} + x_{kmij} + x_{mijk} + x_{ikjm} + x_{jmik} \leq 1$ is valid for the HLP. When we apply sequential lifting to variable x_{ijmk}, its lift coefficient is $1 - \max\{x_{ijkm} + x_{jkmi} + x_{kmij} + x_{mijk} + x_{ikjm} + x_{jmik} : x$ is feasible and $x_{abcd} = 0$ for all x_{abcd} incom- patible with $x_{ijmk}\}$. Variable x_{ijmk} is incompatible with $x_{ijkm}, x_{jkmi}, x_{kmij}, x_{mijk}, x_{ikjm}$ and x_{jmik}, thus the maximum is zero and the lift coefficient is one.

Next, the lift coefficient for x_{jkim} is $1 - \max\{x_{ijkm} + x_{jkmi} + x_{kmij} + x_{mijk} + x_{ikjm} + x_{jmik} + x_{ijmk} : x$ is feasible and $x_{abcd} = 0$ for all x_{abcd} incompatible with $x_{jkim}\}$. Again, x_{jkim} is incompatible with x_{ijkm}, $x_{jkmi}, x_{kmij}, x_{mijk}, x_{ikjm}, x_{jmik}$ and x_{ijmk}, hence the lift coefficient is also 1.

Sequentially proceeding with variables x_{kmji}, x_{mikj}, x_{ikmj}, x_{jmki}, x_{ijkk}, x_{jkmm}, x_{kmii} and x_{mijj} in this order, the inequality in the statement is obtained. □

Proposition 5. *Inequality*

$$x_{akbi} + x_{bcjm} + x_{bdcm} + x_{bkcd} + x_{cjkm} + x_{emgk} + x_{fmek} + x_{gmfk} + x_{hkab} + x_{ikbh} +$$

$$x_{akib} + x_{bcmj} + x_{bdmc} + x_{bkdc} + x_{cjmk} + x_{emkg} + x_{fmke} + x_{gmkf} + x_{hkba} + x_{ikhb} +$$

$$x_{akbb} + x_{akii} \leq 1.$$

is a sequential lifting of Inequality (17) in Corollary 4.

Proof. Similar to the proof of the previous proposition, sequential lifting is applied to lift the coefficients of $x_{akib}, x_{bcmj}, x_{bdmc}, x_{bkdc}, x_{cjmk}$, x_{emkg}, x_{fmke}, x_{gmkf}, $x_{hkba}, x_{ikhb}, x_{akbb}$ and x_{akii} in this order. □

3. Separation algorithms and some computational experiences

We devote this section, first, to a practical computational aspect: a proposal for the identification, from a set of exponentially-many new clique inequalities, of those that might be useful for helping off-the-shelf solvers. Then, some computational experiences are reported to evaluate the performance of the new proposals when tackling a set of instances of a hub location model.

3.1. Separating the new valid inequalities

Suppose we wish to solve a four-index HLP. Its continuous relaxation is a linear program (LP) yielding a solution $\hat{x}$. If $\hat{x}$ is integral, we have solved the integer linear program (ILP). If not, a way to help standard ILP solvers is to find some valid inequalities and add them to the formulation. Geometrically speaking, such inequalities *separate* the current fractional LP solution $\hat{x}$ from the feasible integer solutions. This separation could be performed in at least two ways:

- By means of an exact separation technique that, taking $\hat{x}$ as input, returns as output one or more violated inequalities, if any exist.
- Designing a heuristic separation procedure that takes $\hat{x}$ as input, and outputs either one or more violated inequalities, or a failure message if none is found.

Let $\alpha_{(i,j)(k,m)} = \max\{\hat{x}_{ijkm}, \hat{x}_{jikm}, \hat{x}_{ijmk}, \hat{x}_{jimk}\}$ for all $(i,j),(k,m) \in B : \theta_{(i,j)(k,m)} = 0$. The exact separation problem of the new clique inequalities can be reduced to the resolution of

$$\max \sum_{s \in B} \sum_{t \in B : \theta_{st} = 0} \alpha_{st} z_{st}$$

s.t.: (13)–(16).

As an alternative to solving this model, we also propose a heuristic procedure, whose steps are shown in Algorithm 1, to generically separate clique cuts induced from Proposition 2. The underlying idea of this proposal is to perform a greedy search of the most violated clique inequalities. To do so, we begin by sorting the x-variables that have positive values in the fractional solution in non-increasing order, and placing them in a sorted vector, say $\bar{P}$ (see line 3 of Algorithm 1). This sorted vector will be the one to be explored, from its first to last position, at any global iteration. In each of these iterations, its pth element will be the first to be included in a potentially violated clique, say Q. Then, in a process that we have called *forward search*, starting from the pth+1 position onwards up to the $|P|$-th position, we check whether there are other variables in $\bar{P}$ that are incompatible with all the elements in Q. Each of these occurrences is added to Q to enlarge it. In this way, when we check, at any step, whether a variable is incompatible with all elements in Q, we are verifying such a condition with the most updated version of Q. Later, in a second process that we have called *rolling search*, we also check whether variables in $\bar{P}$ from its first to its pth−1 position could also be added to Q in the same way. Once we finish these two searches, we check whether the variables in Q induce a violated clique inequality to be added as a new row of the linear program. If this is the case, before adding the new row, we try to enlarge Q even more with variables that took value zero in the linear relaxation, following the same strategy as in the other searches, i.e., by iteratively checking whether any of these variables with value zero are incompatible with all the variables in Q. The entire process can be repeated at any time in a row-generation algorithm based on linear relaxations.

1 Initialize sets P and Z as $P = \emptyset$ and $Z = \emptyset$.
2 Identify those x-variables with all different indices that take positive and zero values and place them, respectively, in sets P and Z.
3 Let $\bar{P}$ be the ordered set of the elements in P in non-increasing order of their linear relaxation values.
4 **for** ($p = 1$; $p \leq |\bar{P}|$; $p++$) **do**
5 Let x_p be the p-th element in the ordered set $\bar{P}$.
6 Initialize $Q = \{x_p\}$.
7 //Forward search:
8 **for** ($w = p + 1$; $w \leq |\bar{P}|$; $w++$) **do**
9 Let x_w be the w-th element in the ordered set $\bar{P}$.
10 **if** *x_w is connected with all $q \in Q$* **then**
11 Update $Q := Q \cup \{x_w\}$.
12 //Rolling search:
13 **for** ($r = 1$; $r < p$; $r++$) **do**
14 Let x_r be the r-th element in the ordered set $\bar{P}$.
15 **if** *x_r is connected with all $q \in Q$* **then**
16 Update $Q := Q \cup \{x_r\}$.
17 **if** *Q induces a violated clique inequality* **then**
18 //Zero's search:
19 **foreach** $z \in Z$ **do**
20 **if** *z is connected with all $q \in Q$* **then**
21 Update $Q := Q \cup \{z\}$.
22 Add the violated inequality as a new row to the linear program.

Algorithm 1: Heuristic separation procedure.

3.2. Some computational experiences

We now report some computational experiences to evaluate the performance of the new clique inequalities when tackling a hub location model, complementing the previous sections. For this purpose, six procedures were tested (see specific details in Table 2). They differ in the application (or not) of the automatic cut search identification procedures of the CPLEX solver, and in the separation routine employed

Table 2
Details of the procedures tested.

Procedure name	Separation routine	Automatic CPLEX cuts
Baseline	None	Yes
P1	Exhaustive	No
P2	Exhaustive	Yes
P3	Heuristic	No
P4	Heuristic	Yes
P5	Exhaustive & Heuristic	No
P6	Exhaustive & Heuristic	Yes

for finding the new clique inequalities violated, for which we proposed three alternatives:

- The exhaustive inspection of all four element subsets of V to check the complete family of inequalities from Proposition 4;
- The application of the heuristic separation procedure explained in Algorithm 1;
- The simultaneous application of the two previous methods.

All these procedures were implemented in the C language, making use of the IBM ILOG CPLEX solver version 12.10 through its Callable Library, and compiled with the `gcc` compiler version 11.2.0 using optimization flag `-O3`. This solver was used with the exploitation of multi-threading capabilities, i.e., all cores (eight in our case) are available for use. In the experiments we carried out, we deactivated its `presolve` and `aggregator` methods. The reason to do so was to avoid interactions that could blur the experimental results and conclusions. Other than that, the default solver parameters are assumed unless otherwise stated. Regarding the separation function, we used the `cutcallback` technology at each node of the branch-and-cut tree.

The results reported were performed on a machine with an Intel i7-6700 processor, 3.4 GHz, and 32 GB of RAM, running the Linux Ubuntu 22.04.1 LTS–64 bits operating system.

Computational results with the uncapacitated r-allocation p-hub median problem

The uncapacitated p-hub median problem is a well-known HLP. In it, given a set of nodes with pairwise traffic demands, a prefixed number of nodes, say p, from set V have to be chosen to act as a hub location, and the traffic needs to be routed through these hubs at minimum cost. This problem has been classically studied in two versions with respect to the allocation strategy: the single and the multiple allocation versions. In the single version, each node is assigned to only one hub, thus a node can only send and receive traffic through that hub, whereas in the multiple version there is no such condition and a node may send and receive its traffic through all p hubs.

It was Yaman (2011) who introduced and modeled the r-allocation version, in which each node can be connected to at most a pre-established number, say r, of the p hubs. This version generalizes the two aforementioned versions since its extreme cases of $r = 1$ and $r = p$ are, respectively, the single and multiple versions of the p-hub location problem, while allowing each node to connect up to r of the p hubs for the cases of $1 < r < p$ as intermediate possibilities. We note that no fixed costs for installing hubs are taken into account in this model since the decision on how many hubs to open is exogenously taken for a given p.

For $i \in V$ and $j \in V$, we use t_{ij} to denote the amount of traffic to be routed from i to j, and c_{ij} as its associated unit routing cost. Similarly, we can use the classical sets of variables for modeling the problem: variable y_{ik} is 1 if node $i \in V$ is allocated to node $k \in V$ and 0 otherwise. If $y_{kk} = 1$, then node k is allocated to itself, implying that a hub is opened at this node k. A flow variable x_{ijkm} is defined for each pair of nodes i and j and each pair of possible hubs k and m. This variable takes the value 1 if the flow from i to j travels using hubs k and m in this order, and 0 otherwise.

Using these variables, the uncapacitated r-allocation p-hub median problem (UrApHMP) can be easily modeled as follows:

$$
\begin{aligned}
\min \quad & \sum_{i\in V}\sum_{j\in V}\sum_{k\in V}\sum_{m\in V} t_{ij}(\chi c_{ik} + \alpha c_{km} + \delta c_{mj}) x_{ijkm}, && && (19)\\
\text{s.t.:} \quad & \sum_{k\in V} y_{kk} = p, && && (20)\\
& \sum_{k\in V} y_{ik} \le r, && i \in V && (21)\\
& y_{ik} \le y_{kk}, && i,k \in V && (22)\\
& \sum_{k\in V}\sum_{m\in V} x_{ijkm} = 1, && i,j \in V && (23)\\
& x_{ijkm} \le y_{ik}, && i,j,k,m \in V, && (24)\\
& x_{ijkm} \le y_{jm}, && i,j,k,m \in V, && (25)\\
& y_{ik} \in \{0,1\}, && i,k \in V, && (26)\\
& x_{ijkm} \in \{0,1\}, && i,j,k,m \in V. && (27)
\end{aligned}
$$

In this model, constraints (20) and (21) ensure that p hubs are open and nodes are allocated to at most r hubs. Constraints (22) force assignments to be valid only to hub nodes. Constraints (23) impose that all the traffic between any origin destination pair is routed, whereas constraints (24) and (25) ensure that if the flow of pair (i, j) travels from hub k to hub m, then i is allocated to k and j to m. Finally, constraints (26) and (27) define the domain of the variables, noting that the integrality of the x-variables can be relaxed, since the model always guarantees the existence of an optimal solution with entries 0 and 1 as each traffic demand is routed on a minimum-cost path in the resulting network.

We mentioned in Sections 1 and 2 that important results about valid inequalities and optimality conditions have been proposed for this hub location problem model in order to improve its polyhedral description and help tighten its linear relaxation. Nonetheless, we decided not to make use of them in the experiments we carried out as they could produce interactions and diffuse our conclusions about the computational usefulness of the new clique inequalities proposed in helping off-the-shelf solvers.

For our computational experiments we used some well-known sets of benchmark instances in hub location models: the CAB (O'Kelly, 1987) and the AP (Ernst & Krishnamoorthy, 1996) data sets, which can be obtained from the ORLIB[1] (Beasley, 1990). These datasets contain distance and cost matrices for up to 200 nodes, although we restricted our experiments to instances of 20 nodes, since they were challenging enough for our machine. We have checked the performance of the new clique inequalities in instances with values of r that are far from p, trying to move away from its multiple allocation version, to see if we can take advantage of the new inequalities in challenging allocation conditions.

Table 3 summarizes the computational effect of our proposals in some CAB and AP instances. The first set of columns gathers the instance information where we show the nature of the data and the tested values of parameters p, r, χ, α and δ. Several sets of columns follow, one for each solving procedure, in which we report the total CPU time needed (in seconds) to solve the instance and the number of violated inequalities that we added. For these experiments, a time limit of 3 h of CPU time was fixed for each instance and method. If a procedure exceeded this limit, its execution was aborted and this is reported in the table as "t.l.", shading its corresponding cells in gray. At the end of the table, we have computed the average CPU time needed for those instances in which all the methods have been able to solve them to optimality without exceeding the time limit ($\widetilde{\text{CPU}}$).

[1] http://people.brunel.ac.uk/mastjjb/jeb/info.html

Table 3
Results for the UrApHMP with each procedure.

Instance information						Baseline	P1		P2		P3		P4		P5			P6		
Name	p	r	χ	α	δ	CPU	CPU	Prop. 4	CPU	Prop. 4	CPU	Algo. 1	CPU	Algo. 1	CPU	Prop. 4	Algo. 1	CPU	Prop. 4	Algo. 1
CAB20	5	3	1	0.9	1	1764	1030	272	1989	412	1533	537	1435	447	1357	232	588	1444	752	436
	5	3	1	0.8	1	1936	1612	92	1699	112	1724	440	1778	880	1709	180	898	1854	252	440
	5	3	1	0.7	1	1372	1662	20	1469	24	1673	520	1039	404	1060	80	421	1767	12	490
	5	3	1	0.6	1	1080	971	0	1121	0	1351	448	1026	296	983	0	296	1370	16	450
	5	3	1	0.5	1	1125	1185	12	1301	8	1565	1182	1156	538	1154	0	538	1411	8	1025
	5	2	1	0.9	1	t.l.	6185	2258	t.l.	2434	6416	2313	10801	2638	7521	1934	2576	9864	1429	1545
	5	2	1	0.8	1	6517	5306	1090	6523	1016	3724	2124	9873	1263	3724	924	1921	9868	1202	1125
	5	2	1	0.7	1	3808	2783	666	4196	356	2677	908	10801	90	3112	196	764	t.l.	170	122
	5	2	1	0.6	1	6409	1646	52	5571	122	2043	685	2013	562	1836	92	721	2228	32	5482
	5	2	1	0.5	1	1902	1038	0	2144	100	1333	650	1927	625	1294	4	720	1854	56	441
	5	1	1	0.9	1	t.l.	t.l.	498	t.l.	492	t.l.	968	t.l.	318	t.l.	630	750	t.l.	901	161
	5	1	1	0.8	1	t.l.	10041	282	t.l.	1156	9399	288	t.l.	142	9268	2444	305	t.l.	2832	334
	5	1	1	0.7	1	7096	4526	168	7901	1040	4771	75	6484	452	4384	116	59	t.l.	1037	261
	5	1	1	0.6	1	8225	4039	256	5034	526	2420	14	6182	370	2151	128	26	8312	1068	664
	5	1	1	0.5	1	4891	2340	140	3462	388	1222	0	6372	124	1246	72	0	3796	502	62
	4	2	1	0.9	1	t.l.	4863	1080	t.l.	1017	5013	1144	8627	1242	5233	1864	1208	7713	1445	817
	4	2	1	0.8	1	8871	3695	652	10015	2922	2996	1451	10309	1295	2646	508	1747	t.l.	1042	1402
	4	2	1	0.7	1	t.l.	3013	92	t.l.	836	3098	1340	t.l.	233	3162	340	1329	t.l.	1253	1194
	4	2	1	0.6	1	4147	3391	100	3599	352	2803	1711	t.l.	1420	2887	116	1461	t.l.	564	1048
	4	2	1	0.5	1	5524	2802	76	5241	136	3064	942	3716	1067	3011	16	966	3668	16	1137
	4	1	1	0.9	1	t.l.	t.l.	2064	t.l.	459	t.l.	1444	t.l.	12	t.l.	972	1239	t.l.	214	0
	4	1	1	0.8	1	t.l.	t.l.	453	t.l.	457	t.l.	227	t.l.	179	t.l.	760	232	t.l.	237	7
	4	1	1	0.7	1	t.l.	7799	1158	t.l.	1302	7653	145	t.l.	203	6915	432	105	t.l.	820	315
	4	1	1	0.6	1	t.l.	3923	296	10151	814	3973	144	t.l.	775	3984	440	59	t.l.	1425	26
	4	1	1	0.5	1	t.l.	3274	336	t.l.	812	3973	144	t.l.	763	4906	536	113	t.l.	330	146
	3	2	1	0.9	1	t.l.	6307	752	t.l.	303	6431	1374	t.l.	0	7092	1292	1684	t.l.	1247	0
	3	2	1	0.8	1	t.l.	3970	72	t.l.	38	3565	3151	5994	0	3529	172	2793	5999	0	0
	3	2	1	0.7	1	t.l.	8376	420	t.l.	194	3582	581	t.l.	0	3358	28	379	t.l.	27	33
	3	2	1	0.6	1	t.l.	6825	80	t.l.	746	9420	2064	t.l.	0	8994	292	2958	t.l.	42	0
	3	2	1	0.5	1	t.l.	2228	24	t.l.	72	9574	8545	t.l.	6	8487	312	4607	t.l.	110	19
	3	1	1	0.9	1	t.l.	t.l.	914	t.l.	114	t.l.	110	t.l.	0	t.l.	947	250	t.l.	114	0
	3	1	1	0.8	1	t.l.	t.l.	1466	t.l.	149	t.l.	297	t.l.	0	10642	1446	331	t.l.	159	0
	3	1	1	0.7	1	t.l.	7813	958	t.l.	157	8433	80	t.l.	28	8873	1472	2027	t.l.	182	2
	3	1	1	0.6	1	t.l.	6747	694	t.l.	186	6549	86	t.l.	0	6678	682	50	t.l.	230	0
	3	1	1	0.5	1	t.l.	4740	572	t.l.	224	5018	12	t.l.	0	5540	796	0	t.l.	193	0
AP20	4	3	3	0.75	2	t.l.	4602	344	t.l.	710	4374	3173	t.l.	391	4661	320	2945	t.l.	432	479
	4	2	3	0.75	2	t.l.	6766	604	t.l.	0	8186	5514	t.l.	1	8199	554	5671	t.l.	0	1
	4	1	3	0.75	2	t.l.	2421	0	8207	378	3470	47	8232	89	3874	76	8	10262	981	131
	3	2	3	0.75	2	t.l.	7132	1104	t.l.	0	5473	2952	t.l.	1213 9	5738	678	2859	t.l.	1286	1293
	3	1	3	0.75	2	t.l.	5257	36	t.l.	1 08	3649	0	t.l.	0	3666	34	0	t.l.	38	0
						$\widetilde{\text{CPU}}$	*4295*				*4314*				*4349*					

Some observations drawn from the computational results in this table follow:

- By looking at the large shaded zones in the table indicating those cases that reached the three-hour CPU time limit, we can see that those procedures that make use of the automatic detection of cuts by CPLEX solver, i.e., `Baseline`, P2, P4 and P6, are those that needed significantly more computing time in this model. Hence, we can discard these procedures in favor of P1, P3 and P5.
- Regarding the number of violated inequalities, it is remarkable to find such a large number of violated cuts with both separating methods, either in isolation or in combination, that are separated by P1, P3 and P5. Despite discarding P6 as competitive, if we look at its results, we can see that the procedure detects many violated cuts simultaneously with its automatic detection routines. This suggests that these automatic routines do not implement the family of incompatibility inequalities that we presented in our main theoretical results.

In our opinion, these results indicate that our theoretical proposals could be potentially useful in helping off-the-shelf solvers when solving four-index hub location problems.

4. Conclusions

We have presented new theoretical results in this study that generalize previous polyhedral results on four-index hub location models. They have helped in deriving valid inequalities of the clique type with a maximum number of different OD pairs. We have proved that ten is the maximum number of different OD pairs that can be found together in these models. When the valid inequalities have been applied to several models of hub location problems, they have proved to be computationally useful embedded as cutting planes in solving procedures in reducing the computing time when compared to off-the-shelf solver options.

CRediT authorship contribution statement

Mercedes Landete: Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Writing – original draft, Writing – review & editing. **Juanjo Peiró:** Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no conflict of interest.

Acknowledgments

This work was supported by the Gobierno de España through Ministerio de Ciencia e Innovación and by ERDF–A way of making Europe under projects RED2018-102363-T, RED2022-134149-T and PID2021-122344NB-I00 (MCIN/ AEI/ 10.13039/501100011033), by the Generalitat Valenciana under projects PROMETEO/2021/063 and CIGE/2022/57, and by the Gobierno de España through Ministerio de Educación y Formación Profesional under travel grant CAS19/00076.

References

Alumur, S. (2019). Hub location and related models. In H. Eiselt, & V. Marianov (Eds.), *Contributions to location analysis* (pp. 237–252). Springer.

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

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

Beasley, J. (1990). OR-library: distributing test problems by electronic mail. *Journal of the Operational Research Society*, *41*, 1069–1072.

Borndörfer, R., & Weismantel, R. (2000). Set packing relaxations of some integer programs. *Mathematical Programming, Series B*, *88*, 425–450.

Calvete, H. L., Domínguez, C., Galé, C., Labbé, M., & Marín, A. (2019). The rank pricing problem: Models and branch-and-cut algorithms. *Computers & Operations Research*, *105*, 12–31.

Campbell, J. (1992). Location and allocation for distribution systems with transhipments and transportation economies of scale. *Annals of Operations Research*, *40*, 77–99.

Campbell, J. (1994). Integer programming formulations of discrete hub location problems. *European Journal of Operational Research*, *72*, 387–405.

Campbell, J. (2013). Modeling economies of scale in transportation hub networks. In *Proceedings of the annual Hawaii international conference on system sciences* (pp. 1154–1163).

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

Cánovas, L., Landete, M., & Marín, A. (2000). New facets for the set packing polytope. *Operations Research Letters*, *27*, 153–161.

Conforti, M., Cornuéjols, G., & Rao, M. R. (1995). Decomposition of wheel-and-parachute-free balanced bipartite graphs. *Discrete Applied Mathematics*, *62*, 103–117.

Contreras, I., & O'Kelly, M. (2019). Hub location problems. In G. Laporte, S. Nickel, & F. Saldanha-da-Gama (Eds.), *Location science* (pp. 327–363). Springer, http://dx.doi.org/10.1007/978-3-030-32177-2_12.

Corberán, A., Landete, M., Peiró, J., & Saldanha-da Gama, F. (2019). Improved polyhedral descriptions and exact procedures for a broad class of uncapacitated p-hub median problems. *Transportation Research, Part B (Methodological)*, *123*, 38–63.

Drezner, Z., & Hamacher, H. W. (2002). *Facility location*. Heidelberg-Berlin: Springer-Verlag.

Edmonds, J. (1962). Covers and packings in a family of sets. *American Mathematical Society. Bulletin*, 494–499.

Eiselt, H. A., & Marianov, V. (2011). *Foundations of location analysis*. Springer US.

Eiselt, H. A., & Marianov, V. (2015). *Applications of location analysis*. Springer International Publishing.

Ernst, A., Hamacher, H., Jiang, H., Krishnamoorthy, M., & Woeginger, G. (2009). Uncapacitated single and multiple allocation p-hub center problems. *Computers & Operations Research*, *36*(7), 2230–2241.

Ernst, A., & Krishnamoorthy, M. (1996). Efficient algorithms for the uncapacitated single allocation p-hub median problem. *Location Science*, *4*, 139–154.

Ernst, A., & Krishnamoorthy, M. (1998). Exact and heuristic algorithms for the uncapacitated multiple allocation p-hub median problem. *European Journal of Operational Research*, *104*, 100–112.

Farahani, R., Hekmatfar, M., Arabani, A., & Nikbakhsh, E. (2013). Hub location problems: a review of models, classification, solution techniques, and applicatons. *Computers & Industrial Engineering*, *64*, 1096–1106.

Johnson, E. L., & Padberg, M. W. (1982). Degree-two inequalities, clique facets, and biperfect graphs. *North-Holland Mathematics Studies*, *66*(C), 169–187.

Kara, B., & Tansel, B. (2000). On the single-assignment p-hub center problem. *European Journal of Operational Research*, *125*, 648–655.

Kara, B., & Tansel, B. (2003). The single-assignment hub covering problem: Models and linearizations. *Journal of the Operational Research Society*, *54*, 59–64.

Laporte, G., Nickel, S., & da Gama, F. S. (2019). Location science, (2nd ed.). Springer, http://dx.doi.org/10.1007/978-3-030-32177-2.

Letchford, A. N., & Vu, A. N. (2021). Facets from gadgets. *Mathematical Programming*, *185*, 297–314.

Marín, A. (2005a). Formulating and solving splittable capacitated multiple allocation hub location problems. *Computers & Operations Research*, *32*, 3093–3109.

Marín, A. (2005b). Uncapacitated euclidean hub location: Strengthened formulation, new facets and a relax-and-cut algorithm. *Journal of Global Optimization*, *33*, 393–422.

Marín, A., Cánovas, L., & Landete, M. (2006). New formulations for the uncapacitated multiple allocation hub location problem. *European Journal of Operational Research*, *172*, 274–292.

Marín, A., & Pelegrín, M. (2019). Adding incompatibilities to the simple plant location problem: Formulation, facets and computational experience. *Computers & Operations Research*, *104*, 174–190.

Marzi, F., Rossi, F., & Smriglio, S. (2019). Computational study of separation algorithms for clique inequalities. *Soft Computing*, *23*, 3013–3027.

Nemhauser, G., & Trotter, L. E. (1974). Properties of vertex packing and independence system polyhedra. *Mathematical Programming*, *6*, 48–61.

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

O'Kelly, M., Campbell, J. F., de Camargo, R. S., & de Miranda, G. (2015). Multiple allocation hub location model with fixed arc costs. *Geographical Analysis*, *47*(1), 73–96.

Padberg, M. (1973). On the facial structure of set packing polyhedra. *Mathematical Programming*, *5*, 199–215.

Sohn, J., & Park, S. (2000). The single allocation problem in the interacting three-hub network. *Networks*, *35*, 17–25.

Trotter, L. (1975). A class of facet producing graphs for vertex packing polyhedra. *Discrete Mathematics*, *12*, 373–388.

Yaman, H. (2011). Allocation strategies in hub networks. *European Journal of Operational Research*, *211*, 442–451.