Hub Interdiction & Hub Protection problems:
Model formulations & Exact Solution methods

Prasanna Ramamoorthy
Sachin Jayaswal
Ankur Sinha
Navneet Vidyarthi

Hub Interdiction & Hub Protection problems:
Model formulations & Exact Solution methods

Prasanna Ramamoorthy

Sachin Jayaswal

Ankur Sinha

Indian Institute of Management Ahmedabad, India.

Navneet Vidyarthi

John Molson School of Business
Concordia University, Canada.

Abstract

In this paper, we present computationally efficient formulations for the hub interdiction and hub protection problems, which are bi-level and tri-level mixed integer linear programs, respectively. In the hub interdiction problem, the aim is to identify a set of r critical hubs from an existing set of p hubs that when interdicted results in the greatest disruption cost to the hub-and-spoke network. Reduction of the bi-level interdiction model to single level is straightforward using Karush-Kuhn-Tucker (KKT) conditions corresponding to the lower level problem; however, this turns out to be computationally inefficient in this context. Therefore, we exploit the structure of the problem using various closest assignment constraints to reduce the hub interdiction problem to single level. The modifications lead to computational savings of almost an order of magnitude when compared against the only model existing in the literature. Further, our proposed modifications offer structural advantages for Benders decomposition, which lead to substantial savings, particularly for large problems. Finally, we study and solve the hub protection problem exactly by utilizing the ideas developed for the hub interdiction problem. The tri-level protection problem is otherwise intractable, and to our best knowledge, has not been solved in the literature.

Keywords: Location, hub-and-spoke network, interdiction, protection, Benders decomposition.

Hub Interdiction & Hub Protection problems:
Model formulations & Exact Solution methods

1 Introduction

Certain infrastructural assets are critical to the functioning of a nation's economy and societal well being. The United States' Department of Homeland Security[1] identifies 16 infrastructure sectors as critical, such that their incapacitation or destruction can be debilitating to the national security, economy, and public health. Three out of these sixteen critical infrastructure sectors, namely transportation systems, communications, and energy employ hub-and-spoke as a dominant network structure because of its operational advantage. Hub-and-spoke networks exploit the economies of scale arising from consolidating at hubs the traffic originating from different sources and/or those destined to different demand points, instead of serving each origin-destination pair directly. Flows from the same origin with different destinations in a hub-and-spoke network are consolidated on their route at the hub where they are combined with flows that have different origins but the same destination (Campbell, 1996). In multi-hub networks, traffic concentrated at a hub is directed to a second hub, which distributes it to the final destinations, thereby exploiting the economies of scale on the inter-hub flows. Another advantage of a a hub-and-spoke network is that it results in fewer links, which makes the network construction cheaper and its maintenance easier, compared to an alternate network with direct connections between all sources and destinations.

O'Kelly (1986) was the first paper to study locating hubs between interacting cities. Campbell (1994) gave the first integer programming formulations for the p-hub median problem, uncapacitated hub location problem, p-hub center and hub covering problems. These models are largely inspired by their facility location counterparts: p-median problem, facility location problem, p-center problem and maximal covering problem. Since then research papers have been published in both single allocation (a non-hub is allocated to only one hub) or multiple allocation (a non-hub is allocated to one or more than one hub), capacitated (hubs have a fixed capacity) or uncapacitated (no limit on hub capacity) median and location problems. Skorin-Kapov et al. (1996), Ebery et al. (2000), Ernst and Krishnamoorthy (1996), Hamacher et al. (2004) are some of the important works in this area. In recent years several variations of the hub location problems have also appeared in literature. Notable among those are, hubs with congestion (Elhedhli and Hu, 2005), (Jayaswal and Vidyarthi, 2013), cycle hub location problem, where hubs are connected in a cycle (Contreras et al., 2016), tree of hubs location problem, where hubs are connected by a tree structure (Contreras et al., 2010), flow dependent economies of scale (O'Kelly and Bryan, 1998), stochastic demands (Contreras et al., 2011b), and hub location over a time period (Contreras et al., 2011a). A review of research papers in hub location can be found in: Alumur and Kara (2008), Campbell and O'Kelly (2012) and Farahani et al. (2013).

While a hub-and-spoke network structure is attractive due to its cost effectiveness, it is prone to sever disruptions in the event of a failure of any of its hubs. This is because failure of any hub in the network disrupts the flows from all the origin and destination points that it serves. A study states that it is possible to disrupt the entire United States' air network by interdicting just 2% of its all

---

[1]https://www.dhs.gov/what-critical-infrastructure

airports (Lewis, 2006). A related real incident is the June'16 attack on Ataturk International airport in Turkey. Ataturk International airport is one of the busiest airports in the world and it serves as a hub for Turkish Airlines, Onur Air and Atlas Global. It was attacked on 28th June 2016 by some gunmen, which left 45 people dead and more than 230 injured. Following the attack, flights destined for Istanbul were diverted to other hubs in the vicinity. The Federation Aviation Administration (FAA) and the Transportation Security Administration (TSA) of the United states' government grounded all passenger and commercial flights to and from Turkey for several hours post the attack, which resulted in traffic disruptions throughout the world. Incidents like this, make it necessary to identify critical hubs in a hub-and-spoke network so that resources may be focused towards their fortification (protection), and this forms the motivation of our study. We study the problem of identifying the critical hubs, which when disrupted will cause the maximum disruption. We call this problem as the hub interdiction problem (HIP). We further study the hub protection problem (HPP), which identifies the hubs to protect, taking into account the reaction of the interdictor (attacker). The hub interdiction and protection problems are modeled as bi-level and tri-level mixed integer programs (MIPs), which are challenging to solve. We present efficient solution methods that are capable of solving large instances of the problems in a reasonable time.

Following are the major contributions of the paper:

- This is among the first few papers to study the bi-level interdiction and tri-level protection problems in the context of hub-and-spoke network design.

- We present alternate ways to exploit the structure of the bi-level interdiction problem to reduce it to a tractable single-level optimization problem.

- We further present Benders decomposition for the different single-level formulations to efficiently solve large instances of the hub interdiction problem.

- To the best of our knowledge, this is the only paper to efficiently solve large instances of the hub interdiction and protection problems to optimality.

In Section 2, we present a literature review on interdiction and protection problems. Section 3 presents the model formulation of the bi-level interdiction problem, followed by two alternate ways of reducing it to a single level problem to make it tractable. The first approach uses the well-known Karush-Kuhn-Tucker (KKT) conditions for the lower level problem, while the second approach exploits the structure of the solution to the lower level problem to replace it by what we call as the closest assignment constraints (CACs). We present alternate forms of CACs, and present their relative merits and computational performances in Sections 4 and 5, respectively. In Section 6, we exploit Benders decomposition of the reduced single level formulation of the hub interdiction problem to solve it more efficiently, and present its computational results. In Section 7, we present the tri-level hub protection problem, followed by its solution using Implicit enumeration in combination with Benders decomposition. We conclude by providing possible future research directions in Section 8.

2   Literature Review

Interdiction problems have been widely studied with respect to network flows (network interdiction) and facility location (facility interdiction) problems. The decision maker in an interdiction problem is interested in identifying the set of nodes/arcs (in network interdiction) or facilities (in facility

interdiction) that when interdicted causes the maximum loss to her. The problem is modeled as a Stackleberg game in which the *attacker* is the leader and the network operator (*defender*) is the follower.

2.1 Network Interdiction

Network interdiction problems identify critical nodes or arcs in a network. The defender operates on the network to optimize her objective that can be one of the following: (i) to pass through the network as fast as possible (shortest path network interdiction) (Corley and Sha, 1982; Israeli and Wood, 2002; Cappanera and Scaparra, 2011) (ii) to move through the network without getting caught (most reliable path interdiction) (Shimizu et al., 2012) or (iii) to maximize the amount of flow passing through the network (maximum flow network interdiction) (Wood, 1993; Cormican et al., 1998). The objectives of the attacker in these models are: (i) to intercept or destroy the arc(s)/node(s) so as to maximize the length of the shortest path, or (ii) to minimize the maximum flow in the network, or (iii) to maximize the probability of detection in the network. These models find applications in disrupting enemy flows (McMasters and Mustin, 1970), infectious disease control (Assimakopoulos, 1987), counter-terrorism (Farley, 2003), interception of nuclear material (Morton et al., 2007) and contraband smuggling (Washburn and Wood, 1995). A review of network interdiction models with applications can be found in Collado and Papp (2012).

2.2 Facility Interdiction and Protection

Facility interdiction problems study the identification of critical facilities in a supply network. Church et al. (2004) proposed the r-interdiction median problem (r-IMP) and r-interdiction covering problem (r-ICP) to study interdiction of facilities under different location scenarios. The r-IMP identifies the set of r facilities to remove from the existing ones to maximize the overall demand weighted transportation cost of serving customers from remaining facilities, whereas r-ICP identifies the set of r facilities that when removed minimizes the total demand that can be covered within a specific distance or time.

Different variants of r-IMP are studied in the literature. Church and Scaparra (2007a) studied an extension of the problem where the success of the attack is uncertain. The authors assumed that the attacks are successful with a given probability. Losada et al. (2012) studied another type of uncertainty in r-IMP which is the uncertainty of the degree of impact created by the attack. This problem identifies disruption scenarios that result in the maximum overall traveling distance for serving all customers when the impact on a facility after an attack is uncertain. A key assumption here is that the degree of interdiction impact on a facility is proportional to the amount of resources employed.

The problems described above assume no restrictions on the capacity of the facilities. Aksen et al. (2014) studied the partial interdiction of capacitated r-IMP, wherein facilities operate with a fixed capacity, which when interdicted, operate with a reduced capacity. The amount of capacity reduction is directly proportional to the interdiction resources deployed. Though various versions of r-IMP are studied (capacitated and uncapacitated, partial and full interdiction), its counterpart (r-ICPs) have received only limited attention in literature.

Church and Scaparra (2007b) studied an extension of r-IMP, known as the r-interdiction median problem with fortification (r-IMF). This problem identifies optimal fortification/protection strategies against interdiction. According to this model, when a facility is fortified/protected, it is completely immune to an attack. Scaparra and Church (2008a) formulated the r-IMF as a bi-level MIP, which is solved using an Implicit enumeration algorithm. Scaparra and Church (2008b) proposed an alternate

method for r-IMF. The idea is to reformulate the problem as a maximal covering problem with precedence constraints, which is solved using an approximate heuristic. This helps in identifying the upper and lower bounds of the problem, which is used to reduce the size of the original problem. This reduced problem is then solved to optimality using a standard MIP solver. Losada et al. (2010), Scaparra and Church (2012), Aksen et al. (2010), Aksen and Aras (2012), Aksen et al. (2013) and Liberatore et al. (2012) are other related works in this area.

2.3 Hub interdiction

Interdiction of hubs in a hub-and-spoke network has received scarce attention in the literature, despite its many useful applications as discussed in Section 1. However, there have been a few studies in closely related areas. An et al. (2015) and Azizi et al. (2016), for example, studied the reliable hub-and-spoke network design problem, which includes the possibility of re-routing flows through backup hubs when the active hubs are disrupted. However, the objective in both these papers is to minimize the weighted sum of pre-disruption and the expected value (over all disruption scenarios) post-disruption transportation cost. HIPs by contrast, study the worst-case loss to the defender.

To the best of our knowledge, Lei (2013) is the only paper on HIP/HPP. The author presented bi-level and tri-level MIP for HIP and HPP, respectively. However, due to the complexity of the problem, computational results are presented only for small instances of HIP, whereas no solution method for HPP is presented. The objective of this paper is to present efficient solution methods capable of solving large instances of HIP and HPP.

3 Problem Description and Model formulation

We consider a hub-and-spoke network with a set of flows (W_{ij}) associated with every source node $i \in N$ and destination node $j \in N$. The flows are always routed through one or two of the hubs from the set $H \subseteq N$ of p hubs to benefit from economies of scale in transportation. The objective of the operator of the network (called defender) is to identify the set of r hubs, which when destroyed by an attacker causes her the maximum cost from rerouting of flows that are affected because of the interdicted hubs. We make a reasonable assumption of $r < p$ since the attacker (typically a terrorist organization) might not have resources to interdict all the hubs. This is the context of HIP, which is modeled as a Stackleberg game. In HIP, the attacker makes the first move by choosing the r hubs to interdict, followed by the defender who decides how to route the flows through the remaining $p - r$ hubs with minimum cost. This is represented as a bi-level MIP. The hierarchical structure of the problem is shown through Figure **??**.

3.1 Model Formulation

In this section, we provide a mathematical formulation for the HIP. To begin with, we introduce the notations used, and then move on to the formulation.

3.1.1 Notations

To model the problem, we define the following indices and parameters:

i :   Index for source nodes, $i \in N$;

j :   Index for destination nodes, $j \in N$;

k :   Index for hub which is connected to i, $k \in H$;

m :   Index for hub which is connected to j, $m \in H$;

α ;   Discount factor for collection (source to hub), $(i \to k)$

δ ;   Discount factor for transhipment (hub to hub), $(k \to m)$

χ ;   Discount factor for distribution (hub to destination), $(m \to j)$

H :   set of all hubs, $H \subseteq N$;

W_{ij} :   Flow from source i to destination j;

d_{ij} :   Cost of travelling from node i to node j;

d_{ijkm} :   Cost of traversing from source i to destination j; $d_{ijkm} = \alpha d_{ik} + \delta d_{km} + \chi d_{mj}$;

p :   No. of hubs present in the system;

r :   No. of hubs to interdict:

The decision variables are defined as follows:

X_{ijkm} :   Fraction of flows from source i to destination j through hubs k and m after interdiction;

z_k :   1, if hub k remains open after interdiction, 0 otherwise.

With the above notation, the hub interdiction problem can mathematically be stated as the following bi-level MIP.

$$[HIP_{2L}] : \max_{z_k} Z \tag{1}$$

$$\text{s.t.} \sum_{k \in H} z_k = p - r \tag{2}$$

$$z_k \in \{0, 1\} \qquad\qquad \forall k \in H \tag{3}$$

$$Z = \min_{X_{ijkm}} \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \sum_{m \in H} W_{ij} d_{ijkm} X_{ijkm} \tag{4}$$

$$\text{s.t.} \sum_{k \in H} \sum_{m \in H} X_{ijkm} = 1 \qquad\qquad \forall i, j \in N \tag{5}$$

$$\sum_{m \in H} X_{ijkm} + \sum_{m \in H/(k)} X_{ijmk} \leqslant z_k \qquad\qquad \forall i, j \in N; k \in H \tag{6}$$

$$X_{ijkm} \geqslant 0 \qquad\qquad \forall i, j \in N; k, m \in H \tag{7}$$

Attacker's objective function (1) maximizes the defender's objective of minimizing the weighted transportation cost. Constraint (2) ensures that $p - r$ hubs remain open after interdiction. Constraints (4) to (7) form the lower level routing problem. Constraint (5) states that the fractional sum of flows between source i and destination j through all possible combinations of hubs k and m should be equal to 1. Constraint (6) models the condition that a flow can happen through and out of the hub only if the hub remains open. The same condition can be alternatively represented by the following set of constraints, as done by Lei (2013).

$$\sum_{k \in H} X_{ijkm} \leqslant z_m \qquad\qquad \forall \, i, j \in N; m \in H$$

$$\sum_{m \in H} X_{ijkm} \leqslant z_k \qquad\qquad \forall \, i, j \in N; k \in H$$

However, the constraint set of the form (6) is proven to be facet defining (Hamacher et al., 2004). Hence, constraint set (6) provides a tighter LP relaxation, which is effective in solving large instances of the hub interdiction problem.

3.2   Reduction to Single level

Bi-level problems, even with linear programs at both levels, are NP-hard problems (Audet et al., 1997; Frangioni, 1995). HIP, which is a bi-level MIP, is even more difficult. Bi-level problems in the literature are traditionally solved by reducing the problem to single level using reduction techniques. We present two alternate ways of reducing HIP to a single level MIP to make it tractable. The first approach is based on the use of the well-known KKT conditions for the lower level problem, while the second approach exploits the structure of the solution to the lower level problem to replace it by CACs.

3.2.1   Single level reduction using lower level KKT conditions

The lower level problem in HIP is a linear program with continuous variables. This makes its reduction to a single level using KKT conditions straightforward. For a given upper level variable $\bar{z}_k$, taking dual variables ϕ_{ij} and λ_{ijk} for constraints (5), (6) we get the following Lagrangian relaxation:

$$\sum_{i\in N}\sum_{j\in N}\sum_{k\in H}\sum_{m\in H} W_{ij}d_{ijkm}X_{ijkm} + \sum_{i\in N}\sum_{j\in N}\phi_{ij}(\sum_{k\in H}\sum_{m\in H} X_{ijkm} - 1)+$$

$$\sum_{i\in N}\sum_{j\in N}\sum_{k\in H}\lambda_{ijk}(\sum_{m\in H} X_{ijkm} + \sum_{m\in H/(k)} X_{ijmk} - \bar{z}_k)$$

Differentiating the expression with respect to X_{ijkm} we get:

$$W_{ij}d_{ijkk} + \phi_{ij} + \lambda_{ijk} \geqslant 0 \qquad\qquad \forall\, i,j \in N, k,m \in H, k = m$$
$$W_{ij}d_{ijkm} + \phi_{ij} + \lambda_{ijk} + \lambda_{ijm} \geqslant 0 \qquad\qquad \forall\, i,j \in N, k,m \in H, k \neq m$$

The single level problem with KKT conditions can be written as:

$$[HIP_{KKT}]: \max_{z_k, X_{ijkm}} \sum_{i\in N}\sum_{j\in N}\sum_{k\in H}\sum_{m\in H} W_{ij}d_{ijkm}X_{ijkm} \tag{8}$$

$$\text{s.t.} \sum_{k\in H} z_k = p - r \tag{9}$$

$$W_{ij}d_{ijkm} + \phi_{ij} + \lambda_{ijk} \geqslant 0 \qquad\qquad \forall\, i,j \in N; k,m \in H, k = m \tag{10}$$

$$W_{ij}d_{ijkm} + \phi_{ij} + \lambda_{ijk} + \lambda_{ijm} \geqslant 0 \qquad\qquad \forall\, i,j \in N; k,m \in H, k \neq m \tag{11}$$

$$\lambda_{ijk}(\sum_{m\in H} X_{ijkm} + \sum_{m\in H/(k)} X_{ijmk} - z_k) = 0 \qquad\qquad \forall\, i,j \in N; k \in H \tag{12}$$

$$\sum_{k\in H}\sum_{m\in H} X_{ijkm} = 1 \qquad\qquad \forall\, i,j \in N \tag{13}$$

$$\sum_{m\in H} X_{ijkm} + \sum_{m\in H/(k)} X_{ijmk} \leqslant z_k \qquad\qquad \forall\, i,j \in N; k \in H \tag{14}$$

$$X_{ijkm}, \lambda_{ijk} \geqslant 0, -\infty \leqslant \phi_{ij} \leqslant \infty \qquad\qquad \forall\, i,j \in N; k,m \in H \tag{15}$$

This resulting single level problem contains non-linear complementary slackness constraint (12), that is linearized using the standard method (Fortuny-Amat and McCarl, 1981). The linearized constraints are:

$$\lambda_{ijk} \leqslant M * \alpha_{ijk} \qquad\qquad \forall\, i,j \in N; k \in H \qquad (16)$$

$$\sum_{m \in H} X_{ijkm} + \sum_{m \in H/(k)} X_{ijmk} - z_k \geqslant -M(1 - \alpha_{ijk}) \qquad \forall\, i,j \in N; k \in H \qquad (17)$$

$$\alpha_{ijk} \in \{0,1\} \qquad\qquad \forall\, i,j \in N; k \in H \qquad (18)$$

The objective function (8) along with constraints (9) - (11), (13) - (15) and (16) - (18) forms the linearized problem. The problem contains $n^2p^2 + 3n^2p + n^2 + 1$ constraints and $n^2p^2 + 2n^2p + n^2 + p$ variables, out of which $p + n^2p$ are binary variables. For a 25-node 10-hub problem, this results in 81,876 constraints and 6,260 binary variables out of a total of 75,635 variables, which makes it a fairly difficult problem to solve. This enormous size is due to the addition of binary variables to convert the mixed integer non-linear program to mixed integer linear program. Given that KKT based reduction might not be suitable to solve large scale HIPs, we look at an alternative formulation that exploits the properties of the problem to come up with a more tractable formulation.

3.2.2 Single level reduction using closest assignment constraints

The attacker in the upper level of the bi-level HIP decides upon his optimal set of r hubs to attack, after which the defender routes the disrupted flows optimally through the remaining hubs. Since the lower level problem contains just the routing decision, we can reduce the lower level problem using a closest assignment constraint by which the flows are allocated to the cheapest cost routes.

Closest assignment constraints are used in facility location problems to allocate customers to their nearby facilities. It is because in facility location problems cost is not always proportional to the distance between customer and the facility. The system might assign a customer to some facility farther from him, while he might want to be assigned to the nearest open facility. The closest assignment constraint captures this requirement. Espejo et al. (2012) and Gerrard and Church (1996) compare different closest assignment constraints used in location problems and study their theoretical properties. These constraints find applications in hazardous facility location (Song et al., 2013), facility location under competition (Dobson and Karmarkar, 1987), and interdiction problems (Liberatore et al., 2011). Lei (2013) converted the bi-level hub interdiction problem to single level using the following set of closest assignment constraints:

$$\sum_{qs \in C_{ijkm}} X_{ijqs} + X_{ijkm} \geqslant z_k + z_m - 1 \qquad\qquad \forall\, i,j \in N; k,m \in H \qquad (CAC1)$$

where, $C_{ijkm} = \{(q,s)|\ d_{ijqs} < d_{ijkm} \text{ or } (d_{ijqs} = d_{ijkm} \text{ and } (q < k \text{ or } (q = k \text{ and } s < m)))\}$.

For a given source - destination (s - d) pair (i,j), $CAC1$ ensures that the flow between them happens only through a path that is no costlier than the path via hubs k and m as long as they are open. This is an extension of its counterpart for facility location problems given by Church and Cohon (1976). $CAC1$ arbitrarily breaks any tie between paths having the same cost. Breaking ties for HIP is not necessary, unlike in facility location problem without which it becomes infeasible. For ease of discussion, we redefine $CAC1$ as:

$$\sum_{qs|d_{ijqs} \leqslant d_{ijkm}} X_{ijqs} + X_{ijkm} \geqslant z_k + z_m - 1 \qquad\qquad \forall\, i,j \in N; k,m \in H \qquad (CAC1')$$

It is noteworthy that the use of $CAC1'$ in place of $CAC1$ makes little computational difference.

We propose two alternative CACs, which are presented below. These constraints are designated as $CAC2$ and $CAC3$. $CAC2$ forbids assignment of flows from any s - d pair (i, j) to a path costlier than the path X_{ijkm} ($i \to k \to m \to j$) when hubs k and m are open (z_k and $z_m = 1$). $CAC2$ is given below:

$$\sum_{qs \in E_{ijkm}} X_{ijqs} \leqslant 2 - z_k - z_m \qquad\qquad \forall\, i, j \in N; k, m \in H \qquad\qquad \text{(CAC2)}$$

where, $E_{ijkm} = \{(q, s)|\ d_{ijqs} > d_{ijkm}\}$. $CAC2$ is similar to the constraint devised by Wagner and Falkson (1975) for facility center problems.

$CAC3$ ensures closest assignment by allocating flows from any s - d pair (i, j) through all the paths X_{ijqs} ($i \to q \to s \to j$) not greater than the current path X_{ijkm} ($i \to k \to m \to j$) when hubs k and m are not interdicted. This is presented below:

$$\sum_{q \in H} \sum_{s \in H} d_{ijqs} X_{ijqs} + (M - d_{ijkm})(z_k + z_m - 1) \leqslant M \qquad\qquad \forall\, i, j \in N; k, m \in H \qquad\qquad \text{(CAC3)}$$

where, $M = \max \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \sum_{m \in H} d_{ijkm}$.

In the above inequality, by fixing z_k and z_m to 1, the allocations X_{ijqs} will be on paths shorter than d_{ijkm}. $CAC3$ is an adaptation of the closest assignment constraint from Berman et al. (2009) for hub location problems.

The single level HIP with the addition of closest assignment constraint takes the following form:

$$[HIP_{1L}]: \max_{y_k, X_{ijkm}} \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \sum_{m \in H} W_{ij} d_{ijkm} X_{ijkm}$$
$$\text{s.t. } (2), (3), (5) - (7)$$
$$CAC1 \quad or \quad CAC2 \quad or \quad CAC3$$

This single level problem is smaller in size compared to the single level problem formulation obtained using KKT conditions. The current problem has the same number of variables as the original bi-level problem, while the latter one has a very large model size due to the extra binary variables that were required to linearize it. For a 25-node 10-hub problem, the reduced single level problem using the closest assignment constraint contains 62,510 variables and 69,376 constraints compared to 75,635 variables and 81,876 constraints for the single level problem using KKT conditions. This makes the KKT approach computationally inefficient when compared with the CACs. Hence going forward, we focus on reduction using closest assignment constraints for solving the HIP.

3.3   Dominance relationship between CACs

In order to find the best closest assignment constraint among the proposed constraints for reduction, we study the dominance relationships between the constraints. A constraint which dominates all the other alternate constraints is the one with the tightest LP relaxation for the problem. Espejo et al. (2012) proposed the rules for dominance relationship between constraints as follows: A constraint dominates the other if the former constraint implies the latter. If both constraints imply one another we say that the constraints are equivalent.

In the following, we state dominance relationships between the closest assignment constraints introduced above.

Proposition 1. *CAC2 is equivalent to CAC1'*

Proof. *CAC2* can be written as:

$$1 - \sum_{(qs)\,|\,d_{ijqs} \leqslant d_{ijkm}} X_{ijqs} \leqslant 2 - z_k - z_m \quad \forall\, i, j \in N; k, m \in H$$

$$\implies \sum_{(qs)\,|\,d_{ijqs} \leqslant d_{ijkm}} X_{ijqs} \geqslant z_k + z_m - 1 \quad \forall\, i, j \in N; k, m \in H$$

Separating X_{ijkm} term we get:

$$\sum_{qs\,|\,d_{ijqs} \leqslant d_{ijkm}} X_{ijqs} + X_{ijkm} \geqslant z_k + z_m - 1 \qquad \forall\, i, j \in N; k, m \in H$$

Hence, $CAC2 \implies CAC1'$. Similarly $CAC1' \implies CAC2$ can be proved. Therefore, $CAC2$ is equivalent to $CAC1'$. $\qquad\square$

Proposition 2. *CAC2 dominates CAC3*

Proof. *CAC2* can be written as:

$$\sum_{(qs)\in E_{ijkm}} X_{ijqs} + z_k + z_m \leqslant 2 \qquad \forall\, i, j \in N; k, m \in H$$

$CAC2$ can be relaxed and written as:

$$X_{ijqs} + z_k + z_m \leqslant 2 \qquad \forall i, j \in N; k, m \in H; (q, s) \in E_{ijkm}. \qquad (CAC2\text{-rel})$$

Now it is evident that, $CAC2 \implies CAC2$-rel, while $CAC2$-rel $\not\implies CAC2$. Therefore $CAC2$ dominates $CAC2$-rel. In order to show that $CAC2$ dominates $CAC3$, we just need to prove that $CAC2$-rel implies $CAC3$.

Multiplying by d_{ijqs} on both sides of $CAC2$-rel and summing it up over $(q, s) \in E_{ijkm}$ we get,

$$\sum_{(qs)\in E_{ijkm}} d_{ijqs}X_{ijqs} + \sum_{(qs)\in E_{ijkm}} d_{ijqs}z_k + \sum_{(qs)\in E_{ijkm}} d_{ijqs}z_m \leqslant 2\sum_{(qs)\in E_{ijkm}} d_{ijqs} \quad \forall\, i, j \in N; k, m \in H$$

Adding $\sum_{(qs)\,|\,d_{ijqs}\leqslant d_{ijkm}} d_{ijqs}X_{ijqs} + (M - d_{ijkm} - \sum_{(qs)\in E_{ijkm}} d_{ijqs})(z_k + z_m - 1)$ to both sides of the above inequality, we get

$$\sum_{q\in H}\sum_{s\in H} d_{ijqs}X_{ijqs} + (M - d_{ijkm})(z_k + z_m - 1) \leqslant \sum_{(qs)\in E_{ijkm}} d_{ijqs} + \sum_{(qs)\,|\,d_{ijqs}\leqslant d_{ijkm}} d_{ijqs}X_{ijqs}$$

$$+ (M - d_{ijkm})(z_k + z_m - 1) - \sum_{(qs)\in E_{ijkm}} d_{ijqs}(z_k + z_m - 1) \quad \forall\, i, j \in N; k, m \in H.$$

In the above constraint, the right hand side takes the maximum value when both z_k and z_m are one. This value is always bounded by M since $\sum_{(qs)\,|\,d_{ijqs}\leqslant d_{ijkm}} d_{ijqs}X_{ijqs} \leqslant d_{ijkm}$. Therefore, we get the following

$$\sum_{q\in H}\sum_{s\in H} d_{ijqs}X_{ijqs} + (M - d_{ijkm})(z_k + z_m - 1) \leqslant M \qquad \forall\, i, j \in N; k, m \in H,$$

which proves that $CAC2$-rel $\implies CAC3$. Hence, $CAC2$ dominates $CAC3$. $\qquad\square$

In the following section, we suggest refinements of $CAC1$ and $CAC2$, and present two additional CAC sets that lead to fewer constraints.

3.4   Reduced formulation

In this section, we propose a reduced formulation for $CAC1$ and $CAC2$ based on constraint dominance principles.

Proposition 3. *For a given s - d pair (i,j) and hubs $(k, m \neq k)$ between them, $CAC1_{ijkm}$ dominates $CAC1_{ijmk}$ when $d_{ijkm} < d_{ijmk}$.*

Proof. For given $CAC1_{ijkm}$, $CAC1_{ijmk}$ and $d_{ijkm} < d_{ijmk}$, comparing $CAC1_{ijkm}$ and $CAC1_{ijmk}$ we see that RHS of both the constraints are the same and LHS of $CAC1_{ijmk}$ contains the terms in the LHS of $CAC1_{ijkm}$ (since $d_{ijkm} < d_{ijmk}$) and additional X_{ijqs} variables.

   $CAC1_{ijkm}$ and CAC_{ijmk} constraints are binding when z_k and $z_m = 1$. The additional X_{ijqs} variables in $CAC1_{ijmk}$ are set to zero because LHS of $CAC1_{ijkm}$ is equal to 1, thereby making $CAC1_{ijmk}$ redundant. Therefore, $CAC1_{ijkm}$ dominates $CAC1_{ijmk}$. $\square$

   Based on Proposition 3, we propose a new formulation for *CAC1* which is given below: We define a set, $H'_{ijkm} = \{(k,m)|d_{ijkm} \leqslant d_{ijmk};$ or $(m,k)|d_{ijmk} < d_{ijkm}$ and $\forall\, i,j \in N, k, m \geqslant k \in H\}$ Next we define the set C'_{ijkm} which eliminates the closest assignment constraint corresponding to the longest path as follows: $C'_{ijkm} = \{(q,s)|\ d_{ijqs} < d_{ijkm}$ or $d_{ijqs} = d_{ijkm}$ and $(q < k$ or$(q = k$ and $s < m));$ $\forall\, i,j \in N, (k,m) \in H'_{ijkm}\}$. The new closest assignment constraint can then be written as follows:

$$\sum_{qs \in C'_{ijkm}} X_{ijqs} + X_{ijkm} \geqslant z_k + z_m - 1 \ \forall\, i,j \in N; k, m \in H \tag{rCAC1}$$

Proposition 4. *For a given s - d pair (i,j) and hubs $(k, m \neq k)$ between them, $CAC2_{ijkm}$ dominates $CAC2_{ijmk}$ when $d_{ijkm} < d_{ijmk}$.*

Proof. Given $CAC2_{ijkm}$ and $CAC2_{ijmk}$ and $d_{ijkm} < d_{ijmk}$, comparing $CAC2_{ijkm}$ and $CAC2_{ijmk}$ we see RHS of both the constraints are same and LHS of $CAC2_{ijkm}$ contains the terms in the LHS of $CAC2_{ijmk}$ (since $d_{ijkm} < d_{ijmk}$) and additional X_{ijqs} variables.

   $CAC2_{ijkm}$ and $CAC2_{ijmk}$ are binding when z_k and $z_m = 1$. $CAC2_{ijkm}$ sets the variables in LHS of $CAC2_{ijmk}$ and additional X_{ijqs} variables to zero, making $CAC2_{ijmk}$ redundant. Therefore, $CAC2_{ijkm}$ dominates $CAC2_{ijmk}$. $\square$

   Based on Proposition 4, we propose a new formulation for $CAC2$ which is given below: We define a set $S'_{ijkm} = \{(k,m)|d_{ijkm} \leqslant d_{ijmk}$ or $(m,k)|d_{ijmk} < d_{ijkm} \ \forall\, i,j \in N, k, m \geqslant k \in H\}$. Next we define the set E'_{ijkm} which eliminates the closest assignment constraint corresponding to the longest path as follows: $E'_{ijkm} = \{(q,s)|d_{ijqs} > d_{ijkm}; \forall i,j \in N, (k,m) \in S'_{ijkm}\}$. The new closest assignment constraint can then be written as:

$$\sum_{qs \in E'_{ijkm}} X_{ijqs} \leqslant 2 - z_k - z_m \qquad \qquad \forall\, i,j \in N; k, m \geqslant k \in H. \tag{rCAC2}$$

   The reduced constraint sets $rCAC1$ and $rCAC2$ have $|N|^2((p^2 + p)/2)$ constraints, while their parents $CAC1$ and $CAC2$ have $|N|^2 p^2$ constraints.

4 Advantages of CAC2 over CAC1

In this section, we outline the advantages of $CAC2$ over $CAC1$ in different stages of the solution process. $CAC2$ has certain structural properties that help in solving the single level problem faster than the one with $CAC1$. These properties are also valid for $rCAC2$ since it is a tighter version of $CAC2$.

4.1 Advantage at Presolve

Presolve procedure is executed by the solver, prior to solving the optimization problem to reduce the size of the given problem by removing the redundant variables and constraints. Probing is a process that is carried out at the presolve step wherein logical consequences are investigated by setting the binary variables at their bounds (Savelsbergh, 1994). In this subsection, we show that $CAC2$ and $rCAC2$ together with constraint (6) eliminate a lot of variables by probing.

Proposition 5. *For a given s - d pair (i,j) and hub k, X_{ijkm} variables that appear common in constraint (6) and $CAC2_{ijkk}$ can be fixed to zero.*

Proof. Given $CAC2_{ijkk}$, let the set A = $\{X_{ijkm}|X_{ijkm} \in CAC2_{ijkk}; (6) \text{ for } i,j,k\}$
Case 1: $z_k = 0$ Variables in set A are reduced to zero by constraint (6) for i, j, k.
Case 2: $z_k = 1$ Variables in set A are reduced to zero by $CAC2_{ijkk}$.
Since the variables in set A are reduced to zero eitherwise, they can be eliminated from the model. □

Thus, $CAC2$ and $rCAC2$ formulations eliminate a lot of variables by probing procedure. Despite the constraints being equivalent, probing reduction using $CAC1$ formulation is not straightforward. It will be obvious from the results provided in the later section, where we will observe that there is little advantage at the presolve stage with the $CAC1$ formulation. In the following subsection, we present the advantage provided by $CAC2$ in a branch-and-bound procedure.

4.2 Advantage at Branch-and-Bound step

In a branch-and-bound procedure, the given MIP is relaxed and the linear relaxation is solved at the root node. Further branching is done by setting the integer variables to its bounds that have taken a fractional value in the optimal solution to the relaxed problem. In our problem, branching is done by setting z_k variables to zero and one. When a z_k variable is set to one, some X_{ijkm} variables are set to zero because of the $CAC2$ formulation. These variables can be eliminated from the model to reduce the model size. Alternatively, when z_k is set to zero some X_{ijkm} variables are eliminated because of constraint (6) which again reduces the model size. This is elaborated in the example below:

Consider a hub interdiction problem with $N = 5$, $p = 3$ and $r = 1$. Let the located hubs be 1,2 and 3. For a given s - d pair $(0,4)$ of the problem we have the following distance matrix: $d_{0411} = 373.8127, d_{0412} = 1006.071, d_{0413} = 2642.653, d_{0421} = 1527.739, d_{0422} = 1375.603, d_{0423} = 1696.158, d_{0431} = 2266.804, d_{0432} = 1745.126, d_{0433} = 1696.158$

Now writing constraint 6, and $CAC2$ constraint for the path 0411 we have:

$$X_{0411} + X_{0412} + X_{0413} + X_{0421} + X_{0431} <= z_1$$
$$X_{0412} + X_{0413} + X_{0421} + X_{0422} + X_{0423} + X_{0431} + X_{0432} + X_{0433} <= 2 - 2z_1$$

As discussed above, in the branch-and-cut tree two branches are created with $z_1 = 1$ and $z_1 = 0$. For the branch with $z_1 = 0$, The variables $X_{0411}, X_{0412}, X_{0413}, X_{0421}, X_{0431}$ are set to zero. For the branch with $z_1 = 1$, The variables $X_{0412}, X_{0413}, X_{0421}, X_{0422}, X_{0423}, X_{0431}, X_{0432}, X_{0433}$ are set to zero. Thus $CAC2$ and $rCAC2$ formulations reduce the size of the problem further for each sub problem in the branch-and-cut tree. This advantage of $CAC2$ and $rCAC2$ further results in improved performance while solving the hub interdiction problem.

5   Computational Results

In the previous section, we studied the relative merits of different CACs in solving the hub interdiction problem. In this section, we present the results of our computational experiments to highlight the degree of computational advantage gained through the use of one CAC vis-a-vis others. For our experiments, we use instances derived from the Civil Aeronautics Board (CAB) dataset containing $|N| = 25$ nodes, and Australian Post (AP) dataset containing $|N| = 200$ nodes. The hub locations used for each of these instances are the optimal hub locations obtained by solving a corresponding uncapacitated p-hub median problem (Ebery et al., 2000). All the computational experiments are performed on a workstation with a 2.60GHz Intel Xeon - e5 processor and 24GB memory, and all the instances are solved using Cplex 12.6.

In Table 1, we present the results of the experiments for CAB dataset with $|N| = 25$ nodes and $p \in \{7, 10\}$ hubs. The discount factors for collection (α) and distribution (χ) are set at 1.0 in the CAB dataset, while the discount factor for transhipment (δ) is varied in the experiments. For $N = 25$ and $p = 7$ hubs, HIP with $CAC1$ or $CAC2$ or $CAC3$ contains 35,626 constraints and 30,633 variables. Similarly, $N = 25$ and $p = 10$ hubs, the number of constraints and variables are 69,376 and 62,511, respectively, which are reported under the column "Original size". The number of constraints and variables remaining in the model after the presolve operation and the CPU time to solve the model are reported under the columns "Cons.", "Vars." and "CPU", respectively for the different CACs. These results clearly show the computational inferiority of $CAC3$ compared to the other CACs, as highlighted by its relatively high CPU times. This is mainly due to a weak LP relaxation of the resulting single-level MIP model, besides relatively very few variables and constraints getting eliminated at the presolve stage. Between $CAC1$ and $CAC2$, the former speeds up the computations by a factor of 2 to 4 for $p = 7$ and by a factor of 2 to almost 7 for $p = 10$, as highlighted by their relative CPU times. This gain comes largely from the elimination of a substantially larger proportions of the variables and constraints in the model using $CAC2$ compared to $CAC1$. With the reduced version of $CAC1$, namely $rCAC1$, the presolve operation is further able to eliminate a significant proportion of the variables remaining in the model with $CAC1$, leading to further savings in CPU times. $rCAC2$, on the other hand, is able to further eliminate only constraints but not variables remaining in the model with $CAC2$. Still, use of $rCAC2$ results in savings in CPU time by a factor of 2 to 4 compared to $rCAC1$, which itself provides savings in CPU time by a factor of 1.5 to 2 compared to $CAC1$. However, compared to $CAC2$, its reduced version, namely $rCAC2$, gives only marginal savings in CPU times since post presolve, $CAC2$ and $rCAC2$ have almost the same problem size.

In Table 2, we present additional results from our computational experiments with the best two CACs, namely $CAC2$ and $rCAC2$, for larger instances of the problem derived using the data related to the first 100, and all 200 nodes in the AP dataset. The observations based on the comparison between the computational performance of $CAC2$ and $rCAC2$ largely carry through from Table 1. However, for many of the AP dataset instances reported in Table 2, Cplex runs out of memory.

Table 1: Computational performance of HIP with different CACs using CAB dataset with $|N| = 25$.

Parameters			Original size		CAC1			CAC2			CAC3			rCAC1			rCAC2		
p	r	δ	Cons.	Vars.	Cons.	Vars.	CPU	Cons.	Vars.	CPU	Cons.	Vars.	CPU	Cons.	Vars.	CPU	Cons.	Vars.	CPU
7	1	0.9	35626	30633	9911	8619	4	1329	845	2	34986	30058	30	6611	8619	3	1329	845	2
	2				27593	23775	15	10383	4655	5	35094	30058	47	18377	23479	8	10365	4655	4
	3				27593	23775	17	10383	4655	6	35094	30058	53	18377	23479	12	10365	4655	5
	4				27593	23775	22	10383	4655	5	35094	30058	65	18377	23479	16	10365	4655	5
	5				27593	23775	13	10383	4655	3	35094	30058	155	18377	23479	10	10365	4655	2
	1	0.5			19519	17015	7	3713	2405	3	34964	30058	45	12815	17015	5	3713	2405	3
	2				29809	25835	18	11220	6567	7	35094	30058	48	19645	25529	12	10904	6567	6
	3				29809	25835	30	11220	6567	7	35094	30058	51	19645	25529	19	10904	6567	8
	4				29809	25835	28	11220	6567	11	35094	30058	80	19645	25529	19	10904	6567	8
	5				29809	25835	9	11220	6567	3	35094	30058	88	19645	25529	6	7817	4922	3
	1	0.1			26581	23197	11	6303	4143	3	34996	30058	47	17061	23197	7	6303	4143	3
	2				31579	27481	28	13176	8669	12	35094	30058	52	20280	27167	16	12956	8669	9
	3				31579	27481	43	13176	8669	10	35094	30058	70	20280	27167	25	12956	8669	10
	4				31579	27481	36	13176	8669	10	35094	30058	73	20280	27167	20	12956	8669	11
	5				31579	27481	8	13176	8669	3	35094	30058	38	20280	27167	5	12956	8669	3
10	1	0.9	69376	62511	28475	25786	14	1743	1102	7	68135	61285	90	17441	25786	9	1743	1102	7
	2				58805	53083	48	14045	6741	12	68241	61285	163	35957	52397	25	14027	6741	11
	3				58805	53083	61	14045	6741	13	68241	61285	393	35957	52397	34	14027	6741	12
	4				58805	53083	74	14045	6741	16	68241	61285	425	35957	52397	45	14027	6741	13
	5				58805	53083	67	14045	6741	15	68241	61285	520	35957	52397	40	14027	6741	13
	6				58805	53083	82	14045	6741	12	68241	61285	659	35957	52397	44	14027	6741	12
	7				58805	53083	96	14045	6741	14	68241	61285	2504	35957	52397	33	14027	6741	11
	8				58805	53083	49	14045	6741	8	68241	61285	914	35957	52397	40	14027	6741	7
	1	0.5			45559	41340	22	5301	3430	8	68099	61285	83	27655	41340	14	5301	3430	7
	2				62389	56487	64	17091	9895	18	68099	61285	150	37917	55765	33	16807	9895	17
	3				62389	56487	84	17091	9895	22	68241	61285	288	37917	55765	52	16807	9895	19
	4				62389	56487	115	17091	9895	25	68241	61285	325	37917	55765	58	16807	9895	23
	5				62389	56487	104	17091	9895	21	68241	61285	572	37917	55765	64	16807	9895	20
	6				62389	56487	109	17091	9895	19	68241	61285	526	37917	55765	59	16807	9895	20
	7				62389	56487	115	17091	9895	18	68241	61285	738	37917	55765	60	16807	9895	18
	8				62389	56487	47	17091	9895	9	68241	61285	1198	37917	55765	28	16807	9895	8
	1	0.1			56371	51118	36	9639	6468	9	68203	61283	96	34049	51118	20	9639	6468	8
	2				64741	58721	62	21443	14681	25	68241	61285	137	39119	57964	41	21055	14681	21
	3				64741	58721	83	21443	14681	26	68241	61285	250	39119	57964	53	21055	14681	27
	4				64741	58721	178	21443	14681	37	68241	61285	358	39119	57964	93	21055	14681	35
	5				64741	58721	152	21443	14681	45	68241	61285	526	39119	57964	66	21055	14681	45
	6				64741	58721	127	21443	14681	39	68241	61285	518	39119	57964	71	21055	14681	39
	7				64741	58721	99	21443	14681	31	68241	61285	468	39119	57964	53	21055	14681	30
	8				64741	58721	31	21443	14681	11	68241	61285	299	39119	57964	19	21055	14681	10

Hence, in the following section, we further exploit the structure of the resulting single-level model with $CAC2/rCAC2$ by solving it using Benders decomposition. This is also motivated by the fact that Benders decomposition has been successfully applied to solve large instances of hub location problems (de Camargo et al., 2009, 2008).

6 Benders Decomposition

In this section, we present the Benders decomposition algorithm for the single level MIP model for the hub interdiction problem, obtained through the use of $CAC2/rCAC2$. Using the usual trick used in Benders decomposition of separating the MIP model into a master problem consisting of constraints only involving integer variables, and a sub-problem consisting of rest of the constraints leads to the following master problem:

$$\max \theta \tag{19}$$

$$s.t. \sum_{k \in H} z_k = p - r \tag{20}$$

$$z_k \in \{0, 1\} \tag{21}$$

Solving the master problem provides the values of the z variables, say $\bar{z}$, and an upper bound to the original problem. For values of z fixed at $\bar{z}$ by the master problem, the sub-problem for the hub interdiction problem with $rCAC2$ can be written as:

$$\max \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \sum_{m \in H} W_{ij} d_{ijkm} X_{ijkm} \tag{22}$$

$$s.t. \sum_{i \in N} \sum_{j \in N} X_{ijkm} = 1 \qquad \forall \, k, m \in H \tag{23}$$

$$\sum_{m \in H} X_{ijkm} + \sum_{m \in H/k} X_{ijmk} \leqslant \bar{z}_k \qquad \forall \, i, j \in N, k \in H \tag{24}$$

$$\sum_{qs \in E'_{ijkm}} X_{ijqs} \leqslant 2 - \bar{z}_k - \bar{z}_m \qquad \forall \, i, j \in N, k, m \in H \tag{25}$$

$$X_{ijkm} \geqslant 0 \qquad \forall \, i, j \in N, k, m \in H \tag{26}$$

Associating ϕ_{ij}, λ_{ijk} and β_{ijkm} as the dual variables with the constraints (23),(24) and (25), respectively, we get the following dual of the sub-problem:

$$\min \sum_{i \in N} \sum_{j \in N} \phi_{ij} + \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \lambda_{ijk} \bar{z}_k + \sum_{i \in N} \sum_{j \in N} \sum_{(k, m \geqslant k) \in B^1_{ijkm}} \beta_{ijkm} (2 - \bar{z}_k - \bar{z}_m) \tag{27}$$

$$s.t \sum_{(q,s) \in B^2_{ijkm}} \beta_{ijqs} + \lambda_{ijk} + \phi_{ij} \geqslant W_{ij} d_{ijkm} \qquad \forall \, i, j \in N, k, m \in H, k = m \tag{28}$$

$$\sum_{(q,s) \in B^2_{ijkm}} \beta_{ijqs} + \lambda_{ijk} + \lambda_{ijm} + \phi_{ij} \geqslant W_{ij} d_{ijkm} \qquad \forall \, i, j \in N, k, m \in H, k \neq m \tag{29}$$

$$\beta_{ijkm} \geqslant 0 \qquad \forall \, i, j \in N, (k, m \geqslant k) \in B^1_{ijkm} \tag{30}$$

$$\phi_{ij}, \lambda_{ijk} \geqslant 0 \qquad \forall \, i, j \in N, k \in H \tag{31}$$

where, $B^1_{ijkm} = \{(k,m)|d_{ijkm} \leqslant d_{ijmk} \text{ or } (m,k)|d_{ijmk} < d_{ijkm}\}$.

$\quad B^2_{ijkm} = \{(q,s)|d_{ijqs} \leqslant d_{ijsq} \text{ and } d_{ijkm} > d_{ijqs} \text{ or } (s,q)|d_{ijsq} < d_{ijqs} \text{ and } d_{ijkm} > d_{ijqs} \ \ \forall q, s \geqslant q\}$.

$\quad$ The dual for the primal sub-problem with $CAC2$ added instead of $rCAC2$ can be similarly written as:

$$\min \sum_{i \in N} \sum_{j \in N} \phi_{ij} + \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \lambda_{ijk} z_k + \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \sum_{m \in H} \beta_{ijkm}(2 - z_k - z_m) \tag{32}$$

$$s.t \sum_{qs|d_{ijqs} < d_{ijkm}} \beta_{ijqs} + \lambda_{ijk} + \phi_{ij} \geqslant W_{ij} d_{ijkm} \qquad\qquad \forall\, i, j \in N, k, m \in H, k = m \tag{33}$$

$$\sum_{qs|d_{ijqs} < d_{ijkm}} \beta_{ijqs} + \lambda_{ijk} + \lambda_{ijm} + \phi_{ij} \geqslant W_{ij} d_{ijkm} \qquad\qquad \forall\, i, j \in N, k, m \in H \tag{34}$$

$$\beta_{ijkm}, \phi_{ij}, \lambda_{ijk} \geqslant 0 \qquad\qquad\qquad\qquad \forall\, i, j \in N, k, m \in H. \tag{35}$$

In Benders decomposition, the master and the sub problems are solved iteratively, with (optimality and feasibility) cuts derived from the dual of the subproblem at a given iteration added to the master problem in the subsequent iteration. With $\bar{\phi}_{ij}$, $\bar{\lambda}_{ijk}$ and $\bar{\beta}_{ijkm}$ as the optimal duals associated with the constraints (26), (27) and (28) at a given iteration, we get the following optimality cut:

$$\theta \leqslant \sum_{i \in N} \sum_{j \in N} \bar{\phi}_{ij} + \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \bar{\lambda}_{ijk} z_k + \sum_{i \in N} \sum_{j \in N} \sum_{(k,m) \in B^1_{ijkm}} \bar{\beta}_{ijkm}(2 - z_k - z_m). \tag{36}$$

(36) is the optimality cut when the bi-level hub interdiction model is reduced to a single level model using $rCAC2$. If instead, $CAC2$ is used, then the optimality cut added to the master problem is of the following form:

$$\theta \leqslant \sum_{i \in N} \sum_{j \in N} \bar{\phi}_{ij} + \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \bar{\lambda}_{ijk} z_k + \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \sum_{m \in H} \bar{\beta}_{ijkm}(2 - z_k - z_m). \tag{37}$$

Solving the master problem with the addition of Benders cut gives the upper bound, while the subproblem gives a lower bound to the original problem. The algorithm terminates when the difference between upper and and the best lower bound falls within a pre-specified tolerance for ϵ optimality gap.

$\quad$ **Remark:** As discussed in the previous section, $rCAC2$ does not enjoy much computational advantage over $CAC2$ when solving the resulting single-level MIP directly using Cplex since both of them result in almost the same model size (exactly the same number of variables and almost the same number of constraints) post-presolve, as evident from Table 1. However, when the MIP is decomposed into a master problem consisting of only the binary variables (z_k) and a subproblem consisting of only the continuous variables (X_{ijkm}), presolve loses its effect. Nonetheless, with Benders decomposition, the dual subproblem (27)-(31) resulting from the use of $rCAC2$ has fewer variables (β_{ijkm}) compared to the dual subproblem (32)-(35) resulting from the use of $CAC2$. With $|N|$ nodes and p hubs, the use of $rCAC2$ and $CAC2$ give $|N|^2(p^2 + p)/2$ and $|N|^2 p^2$ β_{ijkm} variables, respectively. For example, a problem instance with $|N| = 200$ and $p = 10$ results in 2,200,000 β_{ijkm} variables with the use of $rCAC2$, which is otherwise 4,000,000 with the use of $CAC2$. Thus, $rCAC2$ results in a reduction in the size of the dual subproblem by a factor of $0.5 + (0.5/p)$, which tends to 0.50 as p becomes large. We expect Benders decomposition to exploit this reduction in the size of the dual subproblem with $rCAC2$ to provide a computational advantage over $CAC2$.

6.1 Computational results

In this section, we present our computational results for larger instances using $CAC2$ and $rCAC2$. For each of the instances, we impose a CPU time limit of 36000 seconds (10 hours), and report the optimality gap for instances that are not solved to optimality within the prescribed time limit. Table 2 gives a comparison of the computation performances between Cplex and Benders decomposition with $CAC2$ and $rCAC2$. The results clearly highlight that for the instances that Cplex could solve within the prescribed 10 hour time limit, Benders decomposition is computationally faster than Cplex by a factor of $5-11$ and $5-14$ with the use of $CAC2$ and $rCAC2$, respectively. Of the remaining 7 instances that Cplex could not handle owing to memory restriction, Benders decomposition with $CAC2$ could solve only 2 of them to optimality within the prescribed time limit, while the Benders decomposition with $rCAC2$ could solve all of them. Further, for the 2 instances that Benders decomposition with $CAC2$ could solve to optimality within the prescribed time limit, Benders decomposition with $rCAC2$ is computationally faster by a factor of $7-8$. This observation is consistent with the remark we made in the previous section regarding the computational advantage enjoyed by Benders decomposition with $rCAC2$.

Table 2: Computational performance of HIP with CAC2 & rCAC2, with and without Benders decomposition

| Parameters | | | | CPLEX | | | | Benders Decomposition | | | |
| | | | | CAC2 | | rCAC2 | | CAC2 | | rCAC2 | |
$|N|$	p	r	α, χ, δ	Gap	CPU	Gap	CPU	Gap	CPU	Gap	CPU
100	10	5	3,0.75,2	0	8881	0	8315	0	1212	0	1100
100	10	6	3,0.75,2	0	6039	0	5070	0	1290	0	1034
100	10	7	3,0.75,2	0	4299	0	3996	0	562	0	444
100	10	8	3,0.75,2	0	3395	0	3255	0	310	0	233
100	15	5	3,0.75,2	*	*	*	*	87%	36000	0	21080
100	15	6	3,0.75,2	*	*	*	*	86%	36000	0	24091
100	15	7	3,0.75,2	*	*	*	*	94%	36000	0	20606
100	15	8	3,0.75,2	*	*	*	*	85%	36000	0	14956
200	10	5	3,0.75,2	*	*	*	*	42%	36000	0	5779
200	10	6	3,0.75,2	*	*	*	*	26%	36000	0	3328
200	10	7	3,0.75,2	*	*	*	*	0	18572	0	2274
200	10	8	3,0.75,2	*	*	*	*	0	8109	0	1180

* indicates out of memory status

7 Extension to Protection problem

A natural extension of the HIP is the hub protection problem (HPP), wherein the defender has an option to fortify/protect only a subset of q hubs (due to its limited protection resources like budget, etc.) against interdiction. Figure 1 gives a schematic representation of the tri-level Hub protection problem. While protection problems have been widely studied in the context of facility location (Church and Scaparra, 2007b; Scaparra and Church, 2008a,b, 2012; Aksen et al., 2010; Aksen and Aras, 2012; Aksen et al., 2013), the same has not received much attention in the context of hub location. To the best of our knowledge, Lei (2013) is the only paper on hub protection. The author

Level 1: Defender's problem (Hub protection)

Objective: Minimize the attacker's interdiction cost

Decision: Which q hubs to protect?

Variable type: Binary

Constrained
optimization
problem

Level 2: Attacker's problem (Hub interdiction)

Objective: Maximize the defender's routing cost

Decision: Which r hubs to interdict out of the remaining p–q unprotected hubs?

Variable type: Binary

Level 3: Defender's problem (Routing problem)

Constrained
optimization
problem

Objective: Minimize routing cost after interdiction

Decision: How to route the flow through the remaining hubs?
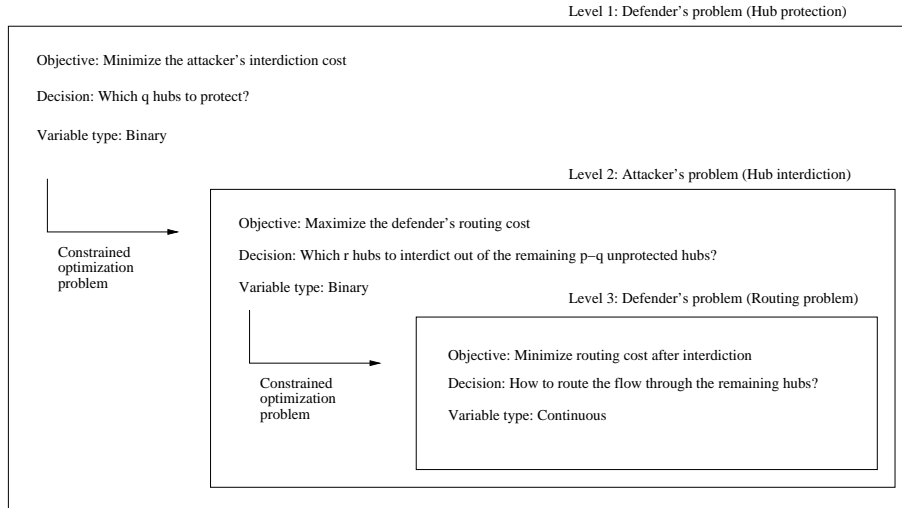
Variable type: Continuous

Figure 1: Hub protection problem as a tri-level MIP

presents a tri-level MIP formulation for HPP, which is reduced to a bi-level MIP using $CAC1$ (see Section 3.2.2). However, the author does not present any computational results in absence of any proposed solution method. In this section, we also present a tri-level MIP formulation for HPP. However, we reduce the tri-level MIP to bi-level MIP using $rCAC2$, which has been shown to perform the best among the different versions of CACs discussed in Section 3.2.2. This allows us to solve large instances of HPP using an efficient algorithm (Implicit enumeration + Benders decomposition) as described in the following subsection.

To model HPP, we define a new set of binary variables y_k which is 1, if the hub k is protected, 0 otherwise. With these new variables, the HPP can be modeled as a tri-level MIP.

7.1 Model Formulation

$$[HPP_{3L}] : \min_{y_k} Z_1 \tag{38}$$

$$\text{s.t.} \sum_{k \in H} y_k = q \tag{39}$$

$$y_k \in \{0, 1\} \tag{40}$$

$$Z_1 = \max_{z_k} Z_2 \tag{41}$$

$$\text{s.t.} \sum_{k \in H} z_k = p - r \tag{42}$$

$$y_k \leqslant z_k \qquad \qquad \forall \, k \in H \tag{43}$$

$$z_k \in \{0, 1\} \qquad \qquad \forall \, k \in H \tag{44}$$

$$Z_2 = \min_{X_{ijkm}} \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \sum_{m \in H} W_{ij} d_{ijkm} X_{ijkm} \tag{45}$$

$$\text{s.t.} \sum_{k \in H} \sum_{m \in H} X_{ijkm} = 1 \qquad \qquad \forall \, i, j \in N \tag{46}$$

$$\sum_{m \in H} X_{ijkm} + \sum_{m \in H/(k)} X_{ijmk} \leqslant z_k \qquad\qquad \forall\, i,j \in N; k \in H \qquad (47)$$

$$X_{ijkm} \geqslant 0 \qquad\qquad \forall\, i,j \in N; k,m \in H \qquad (48)$$

In this HPP_{3L}, the defender makes the first move, protecting q hubs from a set of p hubs (39), followed by the attacker who attacks r hubs out of the remaining $p-q$ unprotected hubs (42). For feasibility, $q+r \leqslant p$. In the above model, the defender in the upper level problem minimizes the attacker's second level objective of maximizing the interdiction cost. The lower two levels form the bi-level HIP, as described in Section 3.1.1, with the additional constraint (43), which ensures that a protected hub cannot be attacked. As discussed in Section 3.2.2, the lower bi-level HIP can be reduced to a single level HIP using the $CACs$. Using the most efficient CAC, namely $rCAC2$, HPP can be restated as the following bi-level program:

$$[HPP_{2L}] : \min_{y_k} Z_1 \qquad\qquad (49)$$

$$\text{s.t.} \sum_{k \in H} y_k = q \qquad\qquad (50)$$

$$y_k \in \{0,1\} \qquad\qquad (51)$$

$$Z_1 = \max_{z_k, X_{ijkm}} \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \sum_{m \in H} W_{ij} d_{ijkm} X_{ijkm} \qquad\qquad (52)$$

$$\text{s.t.} \sum_{k \in H} \sum_{m \in H} X_{ijkm} = 1 \qquad\qquad \forall\, i,j \in N \qquad (53)$$

$$\sum_{k \in H} z_k = p - r \qquad\qquad (54)$$

$$\sum_{m \in H} X_{ijkm} + \sum_{m \in H/(k)} X_{ijmk} \leqslant z_k \qquad\qquad \forall\, i,j \in N; k \in H \qquad (55)$$

$$\sum_{qs \in E'_{ijkm}} X_{ijqs} \leqslant 2 - z_k - z_m \qquad\qquad \forall\, i,j \in N; k,m \geqslant k \in H \qquad (56)$$

$$y_k \leqslant z_k \qquad\qquad \forall\, k \in H \qquad (57)$$

$$z_k \in \{0,1\}, \quad X_{ijkm} \geqslant 0 \qquad\qquad \forall\, i,j \in N; k,m \in H \qquad (58)$$

7.2 Solution method

We present a solution method for HPP, which is inspired by the implicit enumeration algorithm for r-IMF (Scaparra and Church, 2008a). The algorithm is based on the proposition that the optimal solution to r-IMF will necessarily contain atleast one of the facilities interdicted in r-IMP since any other combination of protected facilities will not prevent the worst scenario for the defender.

Implicit enumeration procedure for HPP is described as follows: At the root of the search tree, the algorithm solves an HIP, giving r interdicted hubs. The root node is then branched into r children nodes, each corresponding to protection of a hub k (by setting $y_k = 1$) out of the r interdicted hubs. At each of these r nodes, it solves a conditional hub interdiction problem (CHIP), which is an HIP with the restriction that the protected hub k cannot be interdicted (imposed using constraint set (57)). The solution to each of these CHIPs gives r interdicted hubs. Each of these nodes is in turn branched into r children nodes, each corresponding to protection of a hub k (by setting $y_k = 1$) out of the r interdicted hubs, in addition to the hubs protected at its parent node. This procedure is repeated

till the number of protected hubs on the path starting from the root to the current node is q. Any node at which q hubs are protected is called a leaf node. When each of the paths from the root node terminates in a leaf node, then the node with the lowest objective function value to its corresponding CHIP provides the solution to HPP. At each node in the search tree, HIP/CHIP is reduced to a single level MIP using $rCAC2$, which is solved using Benders decomposition.

To summarize the above procedure, we use $\ddot{y}$ and $\ddot{z}$ to denote the optimal solution vector to the protection and interdiction variables, respectively. Further, $\ddot{\theta}$ denotes the optimal objective function value to HPP. Let r denote the root node of the search tree, and S denote the set of nodes in the tree to be visited. We define the following two sets associated with each node n:

C_n : set of candidate hubs to be protected in the subsequent nodes on the subpath starting from node n.

F_n : set of hubs protected on the path from root to node n.

We use CHIP(F_n) to denote CHIP with the additional restriction that the hubs in F_n cannot be interdicted. Using the above notation, the above procedure has been described in Algorithm 1.

Algorithm 1 Implicit enumeration

1: **procedure** IMPLICIT ENUMERATION
2:     $F_r \leftarrow \phi$
3:     $\ddot{y}_k \leftarrow 0 \quad \forall k \in H$.
4:     Solve CHIP(F_r). $\hat{z} \leftarrow \{k|\ z_k = 0\}$; $\hat{\theta} \leftarrow$ objective function value of CHIP(F_r)
5:     $\ddot{\theta} \leftarrow \hat{\theta}$; $C_r = \{k|\ \hat{z}_k = 0\}$; $S = \{r\}$
6:     **while** $S \neq \phi$ **do**
7:         select $n \in S$
8:         **while** $C_n \neq \phi$ **do**
9:             Select $k \in C_n$
10:             $C_n \leftarrow C_n\backslash\{k\}$
11:             Generate node n_1 with $F_{n_1} = F_n \cup \{k\}$
12:             solve CHIP(F_{n_1}). $\hat{z} \leftarrow \{k|\ z_k = 0\}$; $\hat{\theta} \leftarrow$ objective function value of CHIP(F_n)
13:             **if** $|F_{n_1}| = q$ **then**
14:                 **if** $\hat{\theta} < \ddot{\theta}$ **then**
15:                     $\ddot{z} \leftarrow \hat{z}$; $\ddot{\theta} \leftarrow \hat{\theta}$
16:                     **for** $k \in H$ **do**
17:                         **if** $k \in F_{n_1}$ **then**
18:                             $\ddot{y}_k = 1$
19:                         **else** $\ddot{y}_k = 0$
20:                         **end if**
21:                     **end for**
22:                 **end if**
23:             **else** $C_{n_1} = \{k|\ \hat{z}_k = 0\}$; $S = S \cup \{n_1\}$
24:             **end if**
25:         **end while**
26:     **end while**
27: **return** $\ddot{\theta}, \ddot{y}, \ddot{z}$
28: **end procedure**

The above Implicit enumeration procedure solves $r^0 + r^1 + r^2 + ... + r^q = (r^{(q+1)} - 1)/(r - 1) - 1$ HIP/CHIPs, as opposed $\binom{p}{q}$ CHIPs in complete enumeration of the set of q protected hubs.

7.2.1 Results for the protection problem

Table 3 provides a comparison of the computational performance of the Implicit enumeration algorithm versus complete enumeration for different instances derived from AP dataset. Computational results show that Implicit enumeration provides a computational saving of 6% - 50% compared to complete enumeration.

Table 3: Computational results of the protection problem, (time in seconds).

Parameters					Complete Enumeration	Implicit Enumeration
$|N|$	p	r	q	α, χ, δ	CPU	CPU
50	10	5	1	3,0.75,2	845	425
50	10	5	2	3,0.75,2	2922	1656
50	10	6	1	3,0.75,2	656	394
50	10	6	2	3,0.75,2	2039	1617
50	10	7	1	3,0.75,2	413	289
50	10	7	2	3,0.75,2	1320	1198
100	10	5	1	3,0.75,2	3820	1918
100	10	5	2	3,0.75,2	13176	8059
100	10	6	1	3,0.75,2	3103	1862
100	10	6	2	3,0.75,2	8908	7207
100	10	7	1	3,0.75,2	1956	1369
100	10	7	2	3,0.75,2	5356	5059

8 Conclusion and future research directions

In this paper, we studied hub interdiction problem (HIP) and hub protection problem (HPP), which are formulated as bi-level and tri-level MIPs. For the bi-level HIP, we explored two alternate ways to reduce it to a single level problem. The first approach used the well-known KKT conditions for the lower level problem, while the second approach exploited the structure of the solution to the lower level problem to replace it using closest assignment constraints (CACs). We further studied alternate forms of CACs and their relative computational performances. Our results indicated that the best among our proposed CAC provided a computational advantage (in terms of reduced CPU times) by a factor of 7 times compared to the CAC proposed in the literature. We further provided reduced versions of the alternate CACs, one of which in conjunction with Benders decomposition helped solve large instances of HIP 7-8 times faster than its parent CAC. The computational advantage gained for HIP with the use of CAC and Benders decomposition allowed us to further solve large instances of an otherwise intractable HPP.

The current work opens up a number of exciting possibilities for future research. In this paper, we studied protection of hubs as one approach used by the decision maker to safeguard against interdiction. Yet another approach against interdiction is to consider interdiction possibility at the design of the hub network itself. This, however, will be a much more challenging problem to solve than HPP. Another

interesting extension of HIP and HPP is to incorporate uncertainties in the problem parameters (like demand, etc.), leading to their robust counterparts (Bertsimas and Sim, 2004; Ben-Tal and Nemirovski, 1999). All these problems can further be extended to their capacitated versions, wherein the hubs have limited capacities.

References

Aksen, D., Akca, S. Ş., Aras, N., 2014. A bilevel partial interdiction problem with capacitated facilities and demand outsourcing. Computers & Operations Research 41, 346–358.

Aksen, D., Aras, N., 2012. A bilevel fixed charge location model for facilities under imminent attack. Computers & Operations Research 39 (7), 1364–1381.

Aksen, D., Aras, N., Piyade, N., 2013. A bilevel p-median model for the planning and protection of critical facilities. Journal of Heuristics 19 (2), 373–398.

Aksen, D., Piyade, N., Aras, N., 2010. The budget constrained r-interdiction median problem with capacity expansion. Central European Journal of Operations Research 18 (3), 269–291.

Alumur, S., Kara, B. Y., 2008. Network hub location problems: The state of the art. European journal of operational research 190 (1), 1–21.

An, Y., Zhang, Y., Zeng, B., 2015. The reliable hub-and-spoke design problem: Models and algorithms. Transportation Research Part B: Methodological 77, 103–122.

Assimakopoulos, N., 1987. A network interdiction model for hospital infection control. Computers in Biology and Medicine 17 (6), 413–422.

Audet, C., Hansen, P., Jaumard, B., Savard, G., 1997. Links between linear bilevel and mixed 0–1 programming problems. Journal of optimization theory and applications 93 (2), 273–300.

Azizi, N., Chauhan, S., Salhi, S., Vidyarthi, N., 2016. The impact of hub failure in hub-and-spoke networks: Mathematical formulations and solution techniques. Computers & Operations Research 65, 174–188.

Ben-Tal, A., Nemirovski, A., 1999. Robust solutions of uncertain linear programs. Operations research letters 25 (1), 1–13.

Berman, O., Drezner, Z., Tamir, A., Wesolowsky, G. O., 2009. Optimal location with equitable loads. Annals of Operations Research 167 (1), 307–325.

Bertsimas, D., Sim, M., 2004. The price of robustness. Operations research 52 (1), 35–53.

Campbell, J. F., 1994. Integer programming formulations of discrete hub location problems. European Journal of Operational Research 72 (2), 387–405.

Campbell, J. F., 1996. Hub location and the p-hub median problem. Operations Research 44 (6), 923–935.

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

Cappanera, P., Scaparra, M. P., 2011. Optimal allocation of protective resources in shortest-path networks. Transportation Science 45 (1), 64–80.

Church, R., Scaparra, M. P., 2007a. Analysis of facility systems reliability when subject to attack or a natural disaster. In: Critical Infrastructure. Springer Berlin Heidelberg, pp. 221–241.

Church, R. L., Cohon, J. L., 1976. Multiobjective location analysis of regional energy facility siting problems. Tech. rep., Brookhaven National Lab., Upton, NY (USA).

Church, R. L., Scaparra, M. P., 2007b. Protecting critical assets: the r-interdiction median problem with fortification. Geographical Analysis 39 (2), 129–146.

Church, R. L., Scaparra, M. P., Middleton, R. S., 2004. Identifying critical infrastructure: the median and covering facility interdiction problems. Annals of the Association of American Geographers 94 (3), 491–502.

Collado, R., Papp, D., 2012. Network interdiction - models, applications and unexplored directions. Research report RUTCOR 4 - 2012, Rutgers University.

Contreras, I., Cordeau, J.-F., Laporte, G., 2011a. The dynamic uncapacitated hub location problem. Transportation Science 45 (1), 18–32.

Contreras, I., Cordeau, J.-F., Laporte, G., 2011b. Stochastic uncapacitated hub location. European Journal of Operational Research 212 (3), 518–528.

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

Contreras, I., Tanash, M., Vidyarthi, N., 2016. Exact and heuristic approaches for the cycle hub location problem. Annals of Operations Research, 1–23.

Corley, H., Sha, D. Y., Sep. 1982. Most vital links and nodes in weighted networks. Oper. Res. Lett. 1 (4), 157–160.

Cormican, K. J., Morton, D. P., Wood, R. K., 1998. Stochastic network interdiction. Operations Research 46 (2), 184–197.

de Camargo, R. S., de Miranda Jr, G., Luna, H. P. L., 2008. Benders decomposition for the uncapacitated multiple allocation hub location problem. Computers & Operations Research 35 (4), 1047–1064.

de Camargo, R. S., de Miranda Jr, G., Luna, H. P. L., 2009. Benders decomposition for hub location problems with economies of scale. Transportation Science 43 (1), 86–97.

Dobson, G., Karmarkar, U. S., 1987. Competitive location on a network. Operations Research 35 (4), 565–574.

Ebery, J., Krishnamoorthy, M., Ernst, A., Boland, N., 2000. The capacitated multiple allocation hub location problem: Formulations and algorithms. European Journal of Operational Research 120 (3), 614–631.

Elhedhli, S., Hu, F. X., 2005. Hub-and-spoke network design with congestion. Computers & Operations Research 32 (6), 1615–1632.

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

Espejo, I., Marín, A., Rodríguez-Chía, A. M., 2012. Closest assignment constraints in discrete location problems. European Journal of Operational Research 219 (1), 49–58.

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.

Farley, J. D., 2003. Breaking al qaeda cells: A mathematical analysis of counterterrorism operations (a guide for risk assessment and decision making). Studies in Conflict & Terrorism 26 (6), 399–411.

Fortuny-Amat, J., McCarl, B., 1981. A representation and economic interpretation of a two-level programming problem. Journal of the operational Research Society, 783–792.

Frangioni, A., 1995. On a new class of bilevel programming problems and its use for reformulating mixed integer problems. European journal of operational research 82 (3), 615–646.

Gerrard, R. A., Church, R. L., 1996. Closest assignment constraints and location models: Properties and structure. Location Science 4 (4), 251 – 270.

Hamacher, H. W., Labbé, M., Nickel, S., Sonneborn, T., 2004. Adapting polyhedral properties from facility to hub location problems. Discrete Applied Mathematics 145 (1), 104–116.

Israeli, E., Wood, R. K., 2002. Shortest-path network interdiction. Networks 40 (2), 97–111.

Jayaswal, S., Vidyarthi, N., 2013. Capacitated multiple allocation hub location with service level constraints for multiple consignment classes. Tech. Rep. WP2013-11-02, Indian Institute of Management Ahmedabad, Research and Publication Department.

Lei, T. L., 2013. Identifying critical facilities in hub-and-spoke networks: A hub interdiction median problem. Geographical Analysis 45 (2), 105–122.

Lewis, T. G., 2006. Critical Infrastructure Protection in Homeland Security: Defending a Networked Nation. Wiley-Interscience.

Liberatore, F., Scaparra, M. P., Daskin, M. S., 2011. Analysis of facility protection strategies against an uncertain number of attacks: the stochastic r-interdiction median problem with fortification. Computers & Operations Research 38 (1), 357–366.

Liberatore, F., Scaparra, M. P., Daskin, M. S., 2012. Hedging against disruptions with ripple effects in location analysis. Omega 40 (1), 21–30.

Losada, C., Scaparra, M. P., Church, R. L., 2010. On a bi-level formulation to protect uncapacitated p-median systems with facility recovery time and frequent disruptions. Electronic Notes in Discrete Mathematics 36, 591–598.

Losada, C., Scaparra, M. P., Church, R. L., Daskin, M. S., 2012. The stochastic interdiction median problem with disruption intensity levels. Annals of Operations Research 201 (1), 345–365.

McMasters, A. W., Mustin, T. M., 1970. Optimal interdiction of a supply network. Naval Research Logistics Quarterly 17 (3), 261–268.

Morton, D. P., Pan, F., Saeger, K. J., 2007. Models for nuclear smuggling interdiction. IIE Transactions 39 (1), 3–14.

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

O'Kelly, M. E., Bryan, D., 1998. Hub location with flow economies of scale. Transportation Research Part B: Methodological 32 (8), 605–616.

Savelsbergh, M., 1994. Preprocessing and probing techniques for mixed integer programming problems. ORSA Journal on Computing 6 (4), 445–454.

Scaparra, M. P., Church, R., 2012. Protecting supply systems to mitigate potential disaster a model to fortify capacitated facilities. International Regional Science Review 35 (2), 188–210.

Scaparra, M. P., Church, R. L., 2008a. A bilevel mixed-integer program for critical infrastructure protection planning. Computers & Operations Research 35 (6), 1905–1923.

Scaparra, M. P., Church, R. L., 2008b. An exact solution approach for the interdiction median problem with fortification. European Journal of Operational Research 189 (1), 76–92.

Shimizu, K., Ishizuka, Y., Bard, J. F., 2012. Nondifferentiable and two-level mathematical programming. Springer Science & Business Media.

Skorin-Kapov, D., Skorin-Kapov, J., O'Kelly, M., 1996. Tight linear programming relaxations of uncapacitated p-hub median problems. European Journal of Operational Research 94 (3), 582–593.

Song, B. D., Morrison, J. R., Ko, Y. D., 2013. Efficient location and allocation strategies for undesirable facilities considering their fundamental properties. Computers & Industrial Engineering 65 (3), 475–484.

Wagner, J., Falkson, L., 1975. The optimal nodal location of public facilities with price-sensitive demand. Geographical Analysis 7 (1), 69–83.

Washburn, A., Wood, K., 1995. Two-person zero-sum games for network interdiction. Operations research 43 (2), 243–251.

Wood, R. K., 1993. Deterministic network interdiction. Mathematical and Computer Modelling 17 (2), 1–18.