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

Prasanna Ramamoorthy
Sachin Jayaswal
Ankur Sinha
Navneet Vidyarthi

W.P. No.2016-10-01
October 2016

The main objective of the working paper series of the IIMA is to help faculty members, research staff and doctoral students to speedily share their research findings with professional colleagues and test their research findings at the pre-publication stage. IIMA is committed to maintain academic freedom. The opinion(s), view(s) and conclusion(s) expressed in the working paper are those of the authors and not that of IIMA.
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 problem. The problem 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 for the hub-and-spoke network owner. To begin with, the problem is modeled as a bilevel mixed integer linear program. We explore two ways to reduce this bilevel program to single level by replacing the lower level problem with constraints obtained i) using KKT conditions and ii) by exploiting the structure of the problem. Reduction using KKT conditions is straightforward but computationally inefficient in this context. Exploiting the structure of the problem, we propose two alternate forms of closest assignment constraints and study their computational effectiveness while solving the problem. We also show the dominance relationship between our proposed closest assignment constraints and the only other version studied in the literature. Our computational results suggest that with one form of our proposed closest assignment constraint the resulting model is solved on an average seven times faster than the proposed one in literature. We further propose refinements to these alternate forms of closest assignment constraints which are computationally faster than their original constraints. We also solve the single level hub interdiction problem using a Benders’ decomposition method to fully exploit the potential of our proposed closest assignment constraint. The computational efficiency gained using the closest assignment constraints, makes the trilevel protection problem tractable. We reduce the trilevel hub protection problem to a bilevel problem, and solve it using an Implicit enumeration + Benders’ decomposition procedure.

Keywords: Closest assignment constraints, Hub Interdiction, Hub Protection, Bilevel programs, Constraint dominance, Presolve procedure
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 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 (DHS 2016). In a hub-and-spoke network, hubs act as interim nodes for the flow between the source and destination nodes in the network. Each node in the network is connected to a set of hubs through which they receive or send flows from/to other nodes in the network. Flows from the same origin with different destinations are consolidated on their route at hubs, where they are combined with flows that have different origins but same destinations. These flows are either then routed directly or through other hubs to their respective destination. Since flows between hubs are the collected flows between different nodes, this gives better utilization of the carrier resulting in economies of scale (Campbell 1996). Another advantage is that a hub-and-spoke network results in fewer links when compared to direct connections between all sources and destinations in a network. Fewer links make network maintenance easier and helps bring down network construction costs.

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, 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), multiple allocation (a non-hub is allocated to one or more than one hub), capacitated (hubs have a fixed capacity), 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), (Azizi et al. 2011), 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 (OKelly 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).

Though a Hub-and-spoke network structure is cost effective in operation, the impact associated with the failure of hubs is very severe. This is because, hubs carry inward and outward flow of various nodes and when a hub become dysfunctional, flow of all these nodes gets affected, causing severe disruption in the network. A recent study states that by interdicting just 2% of the airports (64 in number) it is possible to disrupt the entire United States’ air network (Lewis 2006). A related real life example is the June’16 attack on Ataturk International airport in Turkey. Ataturk International air-
port 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 gunmen which resulted in the death of 45 people and more than 230 injuries. 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 that resulted in traffic disruptions throughout the world. Incidents like this makes 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 present a model for identifying critical hubs known as hub interdiction problem, and a model for identifying which hubs to protect in case of an worst case interdiction known as the Hub protection problem. The Hub interdiction problem and Hub protection problem are bilevel and trilevel programs respectively. These problems have binary variables in all the levels except their lower most level which makes them intractable. We study efficient ways of solving these problems using model reduction and exact solution methods. We prove that with the use of these techniques, we are able to solve very large instances of the problems in reasonable time.

In the following section, we present a literature review of interdiction problems. Section 3 presents the model formulation of the bilevel interdiction problem and we study the strategies for conversion of this bilevel problem to a single level problem using KKT conditions and closest assignment constraints. Following this, we study the dominance relationship between the closest assignment constraints and also the advantages provided by them in terms of solution procedure. Section 4 provides the computational results comparing the different closest assignment constraints. In Section 5, we study the benders decomposition procedure for solving the Hub interdiction problem and present computational results for the same. In Section 6, we present the trilevel hub protection problem. We reduce the problem to bilevel using the efficient closest assignment constraint and solve the problem using an Implicit enumeration + Benders decomposition procedure. Finally, we conclude by providing the computational results for the protection problem.

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 nodes/ars (network interdiction) or facilities (facility interdiction) which when interdicted causes the maximum loss to her. The problem is modeled as a Stackleberg (non co-operative) game in which the attacker is the leader and the network operator (defender) is the follower.

2.1 Network Interdiction

A network interdiction problem identifies 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] (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 objective of the attacker in these models is: (i) to intercept or destroy the arc(s)/node(s) so as to maximize the length of the shortest path.
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 in facility median and covering problems. 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 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 an other type of uncertainty in r-IMP which is 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 is that the degree of impact on a facility by its interdiction is proportional to the amount of resources employed.

The problems described above assume no restrictions on the capacity of the facilities which implies that after an interdiction event, the remaining facilities can accommodate any demand. This assumption is not always realistic as facilities operate with fixed capacities in many of the cases. Aksen et al. (2014) studied the partial interdiction of capacitated r-IMP where facilities operate with a fixed capacity and when interdicted, they are not fully destroyed but operate with a reduced capacity. The amount of capacity reduction is directly proportional to the interdiction resources deployed. Though various versions of the r-interdiction median problem are studied (capacitated and uncapacitated, partial and full interdiction), its counterpart (r-interdiction covering problem) has received only limited attention in literature.

Church and Scaparra (2007b) studied an extension of the r-interdiction problem known as the r-interdiction median problem with fortification (r-IMF) which identifies optimal fortification strategies against the worst case interdiction. A key assumption of the fortification problem is that a protected facility cannot be attacked. Scaparra and Church (2008a) formulated the r-IMF as a bilevel mixed integer program. In their version the defender protects a certain number of facilities to mitigate the worst case attack by the attacker. The attacker then interdicts a set of \( r \) facilities from the remaining set of unprotected facilities. The authors formulated an Implicit enumeration algorithm to solve the problem. The same authors proposed an exact method to solve the r-IMF (Scaparra and Church, 2008b). The idea is to reformulate the problem as a maximal covering problem with precedence constraints and solve it using a new approximate heuristic. This new approach helps in identifying the upper and lower bounds of the problem which can be used to reduce the size of the original problem. This reduced problem is then solved to optimality. 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 important papers in this area.
2.3 Hub interdiction

In this paper we study a hub interdiction problem which identifies critical hubs from an existing hub and spoke system. Interdiction of hubs in a hub-and-spoke network has received limited attention in the literature despite its many useful applications. To the best of our knowledge, Azizi et al. (2016) and Lei (2013) are the only two papers studying hub interdiction in literature.

Here, we present a new bilevel formulation for the hub interdiction problem. We explore the reduction of the bilevel problem to single level using: i) KKT conditions and ii) Closest assignment constraints. We propose four new closest assignment constraints and study the performance of these constraints computationally. We also study a hub protection problem which is modeled as a trilevel mixed integer program. We further reduce this trilevel problem to a bilevel problem, without which the problem becomes intractable. We solve the protection problem using an implicit enumeration + Benders decomposition procedure. We also present extensive computational results for both interdiction and protection problems.

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 \(H \subset N\) 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 affected by the interdicted hubs. This is the context of the hub interdiction median (HIM) problem which is modeled as a Stackelberg game. In HIM, 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 bilevel mixed integer linear program.

3.1 Model 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 \( j \), \( m \in H \);

\( \alpha \): Discount factor for collection (source to hub), \((i - k)\)

\( \delta \): Discount factor for transhipment (hub to hub), \((k - m)\)

\( \chi \): Discount factor for distribution (hub to destination), \((m - j)\)

\( H \): set of all Hubs, \( H \subseteq N \);

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

\( d_{ijkm} \): Cost of traversing from source \( i \) to destination \( j \);

\( d_{ijkm} = \alpha \cdot dist_{ik} + \delta \cdot dist_{km} + \chi \cdot dist_{mj} \);

\( p \): no of hubs present in the system;

\( r \): no of hubs interdicted:

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.

**r-Hub Interdiction Median Problem - Multiple Allocation (r-HIMP-MA):**

\[
[M] : \text{max} \ Z
\]

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

\[
z_k \in \{0, 1\} \tag{3}
\]

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

\[
s.t. \sum_{k \in H} \sum_{m \in H} X_{ijkm} = 1 \quad \forall i, j \in N; \tag{5}
\]

\[
\sum_{m \in H} X_{ijkm} + \sum_{m \in H \setminus \{k\}} X_{ijmk} \leq z_k \quad \forall i, j \in N; k \in H \tag{6}
\]

\[
X_{ijkm} \geq 0 \quad \forall i, j \in N; k, m \in H \tag{7}
\]

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) states that a flow can happen through and out of the hub only if the hub remains open. Constraint set (6) is alternatively represented by the following set of constraints in Lei (2013):

\[
\sum_{k \in H} X_{ijkm} \leq z_m \quad \forall i, j \in N; \forall m \in H
\]
$$\sum_{m \in H} X_{ijkm} \leq z_k \quad \forall i, j \in N; \forall k \in H$$

However, constraint (6) is proven to be facet defining which makes its LP relaxation tighter [Hamacher et al. 2004]. Hence, we choose constraint set (6) since a tighter LP relaxation helps in solving the IP faster. This constraint is especially effective in solving large datasets of the hub interdiction problem.

### 3.2 Reduction to Single level

Bilevel problems in the literature are traditionally solved by reducing the problem to single level using various reduction techniques. We explore reducing our bilevel problem to single level either by, KKT (Karush - Kuhn - Tucker) conditions at the lower level or by using closest assignment constraints. The lower level problem in our bilevel interdiction problem is a linear program with continuous variables. This makes the reduction using KKT conditions straightforward. We study the resultant single level program obtained using both techniques and compare them in terms of their respective sizes.

#### 3.2.1 Reduction to single level by writing KKT conditions of the lower level problem

Taking dual variables $\phi_{ijkm}$ and $\lambda_{ijk}$ for constraints (5), (6) we get the following Lagrangean relaxation:

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

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

$$W_{ij} d_{ijkk} + \phi_{ij} + \lambda_{ijk} = 0 \quad \forall i, j \in N, k, m \in H, k = m$$

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

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

$$[M] : \max \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \sum_{m \in H} W_{ij} d_{ijkm} X_{ijkm} \quad (8)$$

s.t.  

$$\sum_{k \in H} z_k = p - r \quad (9)$$

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

$$W_{ij} d_{ijkm} + \phi_{ij} + \lambda_{ijk} + \lambda_{ijm} = 0 \quad \forall i, j \in N; \forall k, m \in H, k \neq m \quad (11)$$

$$\phi_{ij} (\sum_{k \in H} \sum_{m \in H} X_{ijkm} - 1) = 0 \quad \forall i, j \in N \quad (12)$$

$$\lambda_{ijk} (\sum_{m \in H} X_{ijkm} + \sum_{m \in H/(k)} X_{ijmk} - z_k) = 0 \quad \forall i, j \in N; \forall k \in H \quad (13)$$

$$\sum_{k \in H} \sum_{m \in H} X_{ijkm} = 1 \quad \forall i, j \in N \quad (14)$$
\[
\sum_{m \in H} X_{ijkm} + \sum_{m \in H/(k)} X_{ijmk} \leq z_k \quad \forall i, j \in N; \forall k \in H \\
X_{ijkm}, \lambda_{ijk} \geq 0, -\infty \leq \phi_{ij} \leq \infty \quad \forall i, j \in N; \forall k, m \in H 
\]

This resulting single level problem contains non-linear complementary slackness constraints (12) & (13), that are linearized using the method adopted in Fortuny-Amat and McCarl (1981).

The linearized constraints are:

\[
\phi_{ij} \leq M * \alpha_{ij}^1 \quad \forall i, j \in N \\
\sum_{k \in H} \sum_{m \in H} (X_{ijkm} - 1) \geq -M(1 - \alpha_{ij}^1) \quad \forall i, j \in N \\
\lambda_{ijk} \leq M * \alpha_{ijk}^2 \quad \forall i, j \in N; \forall k \in H \\
\sum_{m \in H} X_{ijkm} + \sum_{m \in H/(k)} X_{ijmk} - z_k \geq -M(1 - \alpha_{ijk}^2) \quad \forall i, j \in N; \forall k \in H \\
\alpha_{ij}^1, \alpha_{ijk}^2 \in \{0, 1\} \quad \forall i, j \in N; \forall k \in H
\]

The objective function (10) along with constraints ((9) - (11)), ((14) - (16)) and ((17) - (21)) forms the linearized problem. The problem contains \(n^2p^2 + 3n^2p + 3n^2 + 1\) constraints and \(p + 2n^2 + 2n^2p + n^2p^2\) variables, out of which \(p + n^2 + n^2p\) are binary variables. For a 25-node 10-hub problem, this results in 83,126 constraints and 6,885 binary variables out of a total of 76,260 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 hub interdiction problems, we look at an alternative formulation that exploits the structure of the problem to come up with a more tractable formulation.

3.2.2 reduction to single level using closest assignment constraints:

The attacker in the upper level of the bilevel hub interdiction problem decides upon the optimal set of \(r\) hubs to attack, while the defender routes the disrupted flows 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 were used in facility location problems to allocate customers to their nearby facilities. This is necessary 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 are used in hazardous facility location, facility location under competition, and interdiction problems (Liberatore et al., 2011), Lei (2013) converted the bilevel hub interdiction median problem (HIM) to single level using a closest assignment constraint. This closest assignment constraint is designated as CAC1 and is given below:

\[
\sum_{qs \in C_{ijkm}} X_{ijqs} + X_{ijkm} \geq z_k + z_m - 1 \quad \forall i, j \in N; \forall k, m \in H (CAC1)
\]
where, $C_{ijkm} = \{(q, s) | d_{iq} + \alpha * d_{qs} + d_{sj} < d_{ik} + \alpha * d_{km} + d_{mj} \text{ or } d_{iq} + \alpha * d_{qs} = d_{sj} < d_{ik} + \alpha * d_{km} + d_{mj} \text{ and } (q < k \text{ or } q = k \text{ and } s < m)\}$. 

For a flow between the source $i$, and destination $j$, through hubs $k$ and $m$, $C_A1$ allocates the flows through the path $(i - k - m - j)$ and the paths shorter than it when the hubs $k$ and $m$ are not interdicted ($z_k$ and $z_m = 1$).

We propose two alternative closest assignment constraints which are presented below. This constraints are designated as $C_A2$ and $C_A3$. $C_A2$ forbids assignment of flows to a path longer than $X_{ijkm}$ when hubs $k$ and $m$ are open ($z_k$ and $z_m = 1$) and is given as follows:

$$
\sum_{q,s \in E_{ijkm}} X_{ijqs} \leq 2 - z_k - z_m \forall i,j \in N; \forall k,m \in H \ (C_A2)
$$

where, $E_{ijkm} = \{(q, s) | d_{iq} + \alpha * d_{qs} + d_{sj} > d_{ik} + \alpha * d_{km} + d_{mj}\}$. $C_A3$ ensures closest assignment by allocating flows through all the paths $(i- > q- > s- > j)$ shorter than the path $(i- > k- > m- > j)$ when hubs $k$ and $m$ are not interdicted. This is presented below:

$$
\sum_{q \in H} \sum_{s \in H} d_{ijqs} X_{ijqs} + (M - d_{ijkm})(z_k + z_m - 1) \leq M \forall i,j \in N; \forall k,m \in H \ (C_A3)
$$

where,

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

In the above equation by fixing $z_k$ and $z_m$ to 1, the allocations $X_{ijqs}$ will be on paths shorter than $d_{ijkm}$. $C_A3$ is an adaptation of a closest assignment constraint from Berman et al. (2009) for hub location problems.

The single level Hub Interdiction median problem with the addition of closest assignment constraint takes the following form:

$$
\max_{y_k, X_{ijkm}} \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \sum_{m \in H} W_{ij} d_{ijkm} X_{ijkm}
$$

s.t. (2), (3), (5) - (7) $C_A1$ or $C_A2$ or $C_A3$

The single level hub interdiction problem obtained with the use of closest assignment constraints is very smaller in size when compared to the single level hub interdiction problem obtained using KKT conditions. The former problem has the same number of variables as the original bilevel problem while the latter one has a very large model size due to extra binary variables required for linearizing the problem. For a 25-node 10-hub problem, the reduced single level problem using the closest assignment constraint contains 62510 variables and 131876 constraints compared to 326260 variables and 333126 constraints for the single level problem using KKT conditions. This enormous size makes the KKT approach computationally inefficient when compared with the Closest assignment constraints. Hence going forward, we focus on reduction using closest assignment constraints for solving the Hub Interdiction median problem.
3.3 Dominance relationship between constraints

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.

3.3.1 CAC2 is equivalent to CAC1

CAC2 can be written as:

\[
1 - \sum_{(qs)|D_{ijqs} \leq Di_{ijkm}} X_{ijqs} \leq 2 - z_k - z_m \quad \forall i, j \in N; \forall k, m \in H
\]

SEPARATING \(X_{ijkm}\) TERM WE GET:

\[
\sum_{qs \in C_{ijkm}} X_{ijqs} + X_{ijkm} \geq z_k + z_m - 1 \quad \forall i, j \in N; \forall k, m \in H
\]

Hence, \(CAC2 \implies CAC1\)

Similarly \(CAC1 \implies CAC2\) can be proved.

Therefore, \(CAC1 \equiv CAC2\) □.

3.3.2 CAC2 dominates CAC3

CAC2 can be written as:

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

Multiplying by \(\sum_{qs \in E_{ijkm}} d_{ijqs}\) on both sides we get,

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

Adding the following to both sides of the above inequality,

\[
\sum_{qs \in C_{ijkm}} d_{ijqs} X_{ijqs} + (M - d_{ijkm} - \sum_{qs \in E_{ijkm}} d_{ijqs})(z_k + z_m - 1)
\]
\[
\sum_{qs} d_{ijqs} X_{ijqs} + (M - d_{ijkm})(z_k + z_m - 1) \leq \\
\sum_{(qs) \in E_{ijkm}} d_{ijqs} + \sum_{(qs) \in C_{ijkm}} d_{ijqs} X_{ijqs} + (M - d_{ijkm})(z_k + z_m - 1) - \sum_{(qs) \in E_{ijkm}} d_{ijqs} (z_k + z_m - 1)
\]

RHS of the inequality is bounded by M since \(z_k \leq 1\) further, 
\[
\sum_{(qs) \in C_{ijkm}} d_{ijqs} X_{ijqs} \leq d_{ijkm}
\]
Therefore:
\[
\sum_{q \in H} \sum_{s \in H} d_{ijqs} X_{ijqs} + (M - d_{ijkm})(z_k + z_m - 1) \leq M
\]

\(CAC2 \implies CAC3\)

The results show that \(CAC1\) is equivalent to \(CAC2\), and \(CAC2\) dominates \(CAC3\). In the following section we suggest refinements of \(CAC1\) and \(CAC2\) and present two additional closest assignment constraint sets that have lesser number of constraints in their constraint set than their parent constraints.

### 3.4 Reduced formulation

We propose now a reduced formulation for \(CAC1\) and \(CAC2\) based on the observation that, for a given source, \(i\) and destination, \(j\) and a pair of hubs \(k\) and \(m\) and \(k \neq m\), the distances \(d_{ijkm}\) is either shorter than, greater than, or equal to \(d_{ijmk}\). Based on this observation we make the following proposition:

**Proposition 1.** For a given source \(i\), and destination, \(j\) and hubs \(k\) and \(m\) \((k \neq m)\) between them, \(CAC_{1ijkm}\) dominates the path \(CAC_{1ijmk}\) when \(d_{ijkm} < d_{ijmk}\)

**Proof.** Given \(CAC_{1ijkm}\), \(CAC_{1ijmk}\) and \(d_{ijkm} < d_{ijmk}\).

Comparing \(CAC_{1ijkm}\) and \(CAC_{1ijmk}\), we see RHS of both the constraints are the same and LHS of \(CAC_{1ijmk}\) contains the terms in the LHS of \(CAC_{1ijkm}\) (since \(d_{ijkm} < d_{ijmk}\)) and additional \(X_{ijqs}\) variables.

\(CAC_{1ijkm}\) and \(CAC_{1ijmk}\) constraint is binding when \(z_k\) and \(z_m = 1\). The additional \(X_{ijqs}\) variables in \(CAC_{1ijmk}\) are set to zero because LHS of \(CAC_{1ijkm}\) is equal to 1, making \(CAC_{1ijmk}\) redundant. Therefore \(CAC_{1ijkm} \implies CAC_{1ijmk}\)

Based on proposition 1, we propose a new formulation for \(CAC1\) which is given below:

We define a set 
\[
H'_{ijkm} = \{(k,m)|d_{ik} + \alpha \ast d_{km} + d_{pm} < d_{im} + \alpha \ast d_{mk} + d_{kj}|(k = m); \ (m,k)|d_{im} + \alpha \ast d_{mk} + d_{kj} < d_{ik} + \alpha \ast d_{km} + d_{mj} \ \text{and} \ m < k; \forall i,j \in N,k,m \in H\}
\]

Now we define the set \(C'_{ijkm}\) which eliminates the closest assignment constraint corresponding to the longest path as follows:

\[
C'_{ijkm} = \{(q,s)| \ d_{iq} + \alpha \ast d_{qs} + d_{kj} < d_{ik} + \alpha \ast d_{km} + d_{mj} \ \text{or} \ d_{iq} + \alpha \ast d_{qs} = d_{aj} < d_{ik} + \alpha \ast d_{km} + d_{mj} \ \text{and} \ (q < k \ \text{or}(q = k \ \text{and} \ s < m)); \forall i,j \in N,k,m \in H'_{ijkm}\}
\]
The new closest assignment constraint is:

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

**Proposition 2.** For a given source, $i$ and destination, $j$ and hubs $k$ and $m$ ($k \neq m$) between them, the closest assignment constraint, $CAC2_{ijkm}$ 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 $CAC2_{ijkm} \Rightarrow CAC2_{ijmk}$ \[ \square \]

Based on proposition 2, we propose a new formulation for $CAC2$ which is given below:

$$H''_{ijkm} = \{(k, m) | d_{ik} + \alpha * d_{km} + d_{mj} \leq d_{im} + \alpha * d_{mk} + d_{kj}| (m, k) | d_{ik} + \alpha * d_{km} + d_{mj} > d_{im} + \alpha * d_{mk} + d_{kj} \forall i, j \in N, k, m \geq k \in H\}$$

Now we define the set $E''_{ijkm}$ which eliminates the closest assignment constraint corresponding to the longest path as follows:

$$E''_{ijkm} = \{(q, s) | d_{iq} + \alpha * d_{qs} + d_{sj} > d_{ik} + \alpha * d_{km} + d_{mj}; \forall i, j \in N, k, m \in H''_{ijkm}\}$$

The new closest assignment constraint is:

$$\sum_{qs \in E''_{ijkm}} X_{ijqs} \leq 2 - z_k - z_m \quad \forall i, j \in N; \forall k, m \geq k \in H \quad (rCAC2)$$

4 Advantages of $CAC2$ over $CAC1$

Although $CAC1$ and $CAC2$ constraints are equivalent, $CAC2$ has certain inherent advantages in its formulation which helps in solving the single level problem faster than the $CAC1$ constraint. In this section, we outline the advantages of $CAC2$ over $CAC1$ in different stages of the solution process. These advantages are also applicable to $rCAC2$ since it is a tighter version of $CAC2$.

4.1 Advantage at Presolve step

A presolve algorithm reduces the size of a given optimization problem, by removing redundant variables and constraints [Savelsbergh, 1994]. In this subsection, we show that $CAC2$ and $rCAC2$ eliminate a lot of variables by presolving in conjunction with constraint (6).

**Proposition 3.** $X_{ijkm}$ variables that appear common in constraint (6) and $CAC2_{ijkm}$ for a given $k$ can be fixed to zero.

**Proof.** Given $CAC2_{ijkm}$,

Let the set $A = \{X_{ijkm} | X_{ijkm} \in CAC2; X_{ijkm} \in H\}$

Case 1: $z_k = 0$

Variables in set $A$ are reduced to zero by constraint (6).
Case 2: $z_k = 1$

Variables in set A are reduced to zero by $CAC2_{ijkm}$.

Since the variables in set A are reduced to zero eitherwise, they can be eliminated from the model.

Thus, $CAC2$ and $rCAC2$ formulations eliminates a lot of variables by Presolve algorithm. In the following subsection, we present the advantage provided by them in a branch and cut procedure.

4.2 Advantage at Branch and Cut step

In a branch and cut 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. These advantages in CAC2 and rCAC2 formulations result in improved performance in solving the model.

5 Computational Results

Here we compare the performances of different closest assignment constraints $CAC1$, $CAC2$, $CAC3$, $rCAC1$, $rCAC2$ by solving the single level Hub interdiction problem. All computational experiments were performed on a workstation with a 2.60GHz Intel Xeon - e5 processor and 24GB memory. We use Civil Aeronautics Board (CAB) dataset containing 25 nodes with 7 and 10 hubs and Australian post (AP) dataset for comparison purposes. The hubs selected in CAB dataset and AP dataset are the optimal hubs in a hub location problem.

The effectiveness of $CAC2$ over $CAC1$ and $CAC3$ is presented in table 1 for CAB dataset with 7 hubs and in table 2 for CAB dataset with 10 hubs. We also present the number of remaining rows and columns after presolving the problem in those tables. We see that $CAC2$, $rCAC1$ and $rCAC2$ constraints are very effective than $CAC1$ constraints in reducing the size of the problem and results in better computational time. We also see that, $rCAC2$ is marginally better than $CAC2$ constraint in computational times, but it is on an average four times faster than $CAC1$ constraint. The dominated constraint $CAC3$ is the slowest due to its poor LP relaxation.

In table 3, we present the comparison of $CAC2$ and $rCAC2$ constraints by solving the single level interdiction problem using AP dataset. The computational results show that for large datasets, $rCAC2$ is marginally better than $CAC2$. The dominant constraint and variables in $CAC2$ which are removed to form $rCAC2$ are the variables and constraints which are removed in the presolving stage when solving the MIP with the $CAC2$. Therefore, the advantage $rCAC2$ has over $CAC2$ is the reduction in presolving time. After presolve, the resultant $rCAC2$ constraint has the same number of variables as $CAC2$ but a slightly lesser number of constraints. In order to fully utilise the benefits of $rCAC2$, we propose a Benders decomposition procedure for solving the problem. Since the number of constraints in $rCAC2$ become the number of dual variables in the sub problem of the Benders decomposition, we get advantage in terms of the reduced dimensions of the sub problem.
Table 1: Comparison between CAC1, CAC2, CAC3, rCAC1 and rCAC2 constraints (No. of constraints: 35626, No of variables: 30633).

<table>
<thead>
<tr>
<th>CAB dataset</th>
<th>CAC1</th>
<th>CAC2</th>
<th>CAC3</th>
<th>rCAC1</th>
<th>rCAC2</th>
</tr>
</thead>
<tbody>
<tr>
<td>25 7 1 1.0.9,1 9911 8619 4</td>
<td>1329 845 2</td>
<td>34986 30058 30</td>
<td>6611 8619 3</td>
<td>1329 845 2</td>
<td></td>
</tr>
<tr>
<td>25 7 2 1.0.9,1 27593 23775 15</td>
<td>10383 4655 5</td>
<td>35094 30058 47</td>
<td>18377 23479 8</td>
<td>10365 4655 4</td>
<td></td>
</tr>
<tr>
<td>25 7 3 1.0.9,1 27593 23775 17</td>
<td>10383 4655 6</td>
<td>35094 30058 53</td>
<td>18377 23479 12</td>
<td>10365 4655 5</td>
<td></td>
</tr>
<tr>
<td>25 7 4 1.0.9,1 27593 23775 22</td>
<td>10383 4655 5</td>
<td>35094 30058 65</td>
<td>18377 23479 16</td>
<td>10365 4655 5</td>
<td></td>
</tr>
<tr>
<td>25 7 5 1.0.9,1 27593 23775 13</td>
<td>10383 4655 3</td>
<td>35094 30058 155</td>
<td>18377 23479 10</td>
<td>10365 4655 2</td>
<td></td>
</tr>
<tr>
<td>25 7 6 1.0.9,1 0 0 3</td>
<td>0 0 3</td>
<td>0 0 4</td>
<td>0 0 2</td>
<td>0 0 2</td>
<td></td>
</tr>
<tr>
<td>25 7 1 1.0.5,1 19519 17015 7</td>
<td>3713 2405 3</td>
<td>34964 30058 45</td>
<td>12815 17015 5</td>
<td>3713 2405 3</td>
<td></td>
</tr>
<tr>
<td>25 7 2 1.0.5,1 29809 25835 18</td>
<td>11220 6567 7</td>
<td>35094 30058 48</td>
<td>19645 25529 12</td>
<td>10904 6567 6</td>
<td></td>
</tr>
<tr>
<td>25 7 3 1.0.5,1 29809 25835 30</td>
<td>11220 6567 7</td>
<td>35094 30058 51</td>
<td>19645 25529 19</td>
<td>10904 6567 8</td>
<td></td>
</tr>
<tr>
<td>25 7 4 1.0.5,1 29809 25835 28</td>
<td>11220 6567 11</td>
<td>35094 30058 80</td>
<td>19645 25529 19</td>
<td>10904 6567 8</td>
<td></td>
</tr>
<tr>
<td>25 7 5 1.0.5,1 29809 25835 9</td>
<td>11220 6567 3</td>
<td>35094 30058 88</td>
<td>19645 25529 6</td>
<td>7817 4922 3</td>
<td></td>
</tr>
<tr>
<td>25 7 6 1.0.5,1 0 0 3</td>
<td>0 0 2</td>
<td>0 0 4</td>
<td>0 0 2</td>
<td>0 0 2</td>
<td></td>
</tr>
<tr>
<td>25 7 1 1.0.1,1 26581 23197 11</td>
<td>6303 4143 3</td>
<td>34996 30058 47</td>
<td>17061 23197 7</td>
<td>6303 4143 3</td>
<td></td>
</tr>
<tr>
<td>25 7 2 1.0.1,1 31579 27481 28</td>
<td>13176 8669 12</td>
<td>35094 30058 52</td>
<td>20280 27167 16</td>
<td>12956 8669 9</td>
<td></td>
</tr>
<tr>
<td>25 7 3 1.0.1,1 31579 27481 43</td>
<td>13176 8669 10</td>
<td>35094 30058 70</td>
<td>20280 27167 25</td>
<td>12956 8669 10</td>
<td></td>
</tr>
<tr>
<td>25 7 4 1.0.1,1 31579 27481 36</td>
<td>13176 8669 10</td>
<td>35094 30058 73</td>
<td>20280 27167 20</td>
<td>12956 8669 11</td>
<td></td>
</tr>
<tr>
<td>25 7 5 1.0.1,1 31579 27481 8</td>
<td>13176 8669 3</td>
<td>35094 30058 38</td>
<td>20280 27167 5</td>
<td>12956 8669 3</td>
<td></td>
</tr>
<tr>
<td>25 7 6 1.0.1,1 0 0 3</td>
<td>0 0 2</td>
<td>0 0 4</td>
<td>0 0 2</td>
<td>0 0 2</td>
<td></td>
</tr>
</tbody>
</table>
Table 2: Comparison between CAC1, CAC2, CAC3, rCAC1 and rCAC2 (No. of constraints: 69376, No of variables: 62501).

<table>
<thead>
<tr>
<th>CAB dataset</th>
<th>CAC1</th>
<th>CAC2</th>
<th>CAC3</th>
<th>rCAC1</th>
<th>rCAC2</th>
</tr>
</thead>
<tbody>
<tr>
<td>nodes</td>
<td>p</td>
<td>r</td>
<td>α, δ, χ</td>
<td>Cons. Vars. time (secs)</td>
<td>Cons. Vars. time</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>1</td>
<td>1.0, 0.9, 1</td>
<td>28475 25786 14</td>
<td>1743 1102 7</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>2</td>
<td>1.0, 0.9, 1</td>
<td>58805 53083 48</td>
<td>14045 6741 12</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>3</td>
<td>1.0, 0.9, 1</td>
<td>58805 53083 61</td>
<td>14045 6741 13</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>4</td>
<td>1.0, 0.9, 1</td>
<td>58805 53083 74</td>
<td>14045 6741 16</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>5</td>
<td>1.0, 0.9, 1</td>
<td>58805 53083 67</td>
<td>14045 6741 15</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>6</td>
<td>1.0, 0.9, 1</td>
<td>58805 53083 82</td>
<td>14045 6741 12</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>7</td>
<td>1.0, 0.9, 1</td>
<td>58805 53083 96</td>
<td>14045 6741 14</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>8</td>
<td>1.0, 0.9, 1</td>
<td>58805 53083 49</td>
<td>14045 6741 8</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>9</td>
<td>1.0, 0.9, 1</td>
<td>0 0 9</td>
<td>0 0 8</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>1</td>
<td>1.0, 0.5, 1</td>
<td>45559 41340 22</td>
<td>5301 3430 8</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>2</td>
<td>1.0, 0.5, 1</td>
<td>62389 56487 64</td>
<td>17091 9855 18</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>3</td>
<td>1.0, 0.5, 1</td>
<td>62389 56487 84</td>
<td>17091 9855 22</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>4</td>
<td>1.0, 0.5, 1</td>
<td>62389 56487 115</td>
<td>17091 9855 25</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>5</td>
<td>1.0, 0.5, 1</td>
<td>62389 56487 104</td>
<td>17091 9855 21</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>6</td>
<td>1.0, 0.5, 1</td>
<td>62389 56487 109</td>
<td>17091 9855 19</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>7</td>
<td>1.0, 0.5, 1</td>
<td>62389 56487 115</td>
<td>17091 9855 18</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>8</td>
<td>1.0, 0.5, 1</td>
<td>62389 56487 47</td>
<td>17091 9855 9</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>9</td>
<td>1.0, 0.5, 1</td>
<td>0 0 9</td>
<td>0 0 8</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>1</td>
<td>1.0, 1, 1</td>
<td>56371 51118 36</td>
<td>9639 6468 9</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>2</td>
<td>1.0, 1, 1</td>
<td>64741 58721 62</td>
<td>2143 14681 25</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>3</td>
<td>1.0, 1, 1</td>
<td>64741 58721 83</td>
<td>2143 14681 26</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>4</td>
<td>1.0, 1, 1</td>
<td>64741 58721 178</td>
<td>2143 14681 37</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>5</td>
<td>1.0, 1, 1</td>
<td>64741 58721 152</td>
<td>2143 14681 45</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>6</td>
<td>1.0, 1, 1</td>
<td>64741 58721 127</td>
<td>2143 14681 39</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>7</td>
<td>1.0, 1, 1</td>
<td>64741 58721 99</td>
<td>2143 14681 31</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>8</td>
<td>1.0, 1, 1</td>
<td>64741 58721 31</td>
<td>2143 14681 11</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>9</td>
<td>1.0, 1, 1</td>
<td>0 0 9</td>
<td>0 0 8</td>
</tr>
</tbody>
</table>
Table 3: Comparison between CAC2 & rCAC2

<table>
<thead>
<tr>
<th>nodes</th>
<th>p</th>
<th>r</th>
<th>α, χ, δ</th>
<th>Gap</th>
<th>CPU time</th>
<th>Gap</th>
<th>CPU time</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>10</td>
<td>4</td>
<td>3,0,75,2</td>
<td>0</td>
<td>10524</td>
<td>0</td>
<td>9004</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
<td>5</td>
<td>3,0,75,2</td>
<td>0</td>
<td>8881</td>
<td>0</td>
<td>8315</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
<td>6</td>
<td>3,0,75,2</td>
<td>0</td>
<td>6039</td>
<td>0</td>
<td>5070</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
<td>7</td>
<td>3,0,75,2</td>
<td>0</td>
<td>4299</td>
<td>0</td>
<td>3996</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
<td>8</td>
<td>3,0,75,2</td>
<td>0</td>
<td>3395</td>
<td>0</td>
<td>3255</td>
</tr>
</tbody>
</table>

6 Benders Decomposition

The single level hub interdiction problem with closest assignment constraint is a mixed integer program which makes a great case for solving it using benders decomposition algorithm. In a benders decomposition method, the given MIP is partitioned into a master problem which contains integer variables from the original problem, and a sub problem which contains continuous variables from the original problem. Since the sub problem is a LP the solution to the dual of the sub problem gives different feasible solutions of the original problem, which is used to generate cuts for the master problem. These cuts commonly known as the Benders cuts helps attain master problem better solutions in each iteration. The algorithm terminates when the best of the solutions of the sub problem lies within a minimal gap to the solution of the master problem. In this section we present the benders decomposition algorithm for both CAC2 and rCAC2 constraints and present the computational results for the same.

We decompose the hub interdiction problem into a master problem containing $z_k$ variables and a sub problem containing $X_{ijkm}$ variables.

The Master problem can be written as:

\[
\begin{align*}
\max & \quad \theta \\
\text{s.t} & \quad \sum_{k \in H} z_k = p - r \\
& \quad z_k \in \{0,1\}
\end{align*}
\]

Solving the master problem gives the variable $\bar{z}$ and an upper bound to the original problem.

For the variables $\bar{z}$ fixed by the master problem, the sub problem for the hub interdiction problem with rCAC2 can be written as:

\[
\begin{align*}
\max & \quad \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \sum_{m \in H} W_{ij}d_{ijkm}X_{ijkm}s.t \\
& \quad \sum_{i \in N} \sum_{j \in N} X_{ijkm} = 1 \quad \forall k, m \in H \\
& \quad \sum_{m \in H} X_{ijkm} + \sum_{m \in H/k} X_{ijmk} \leq \bar{z}_k \quad \forall i, j \in N, k \in H \\
& \quad \sum_{qs \in E_{ijkm}} X_{ijqs} \leq 2 - \bar{z}_k - \bar{z}_m \quad \forall i, j \in N, k, m \in H
\end{align*}
\]
\[ X_{ijkm} \geq 0 \forall i, j \in N, k, m \in H \] (30)

Deriving the dual for the primal sub problem (26) - (30) by fixing dual variables \( \phi_{ij}, \lambda_{ijk} \) and \( \beta_{ijkm} \) for constraints (27), (28) and (29) we get:

\[
\begin{align*}
\min & \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \lambda_{ijk} z_k + \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \sum_{m \in H} \beta_{ijkm} (2 - z_k - z_m) \\
\text{s.t} & \sum_{q | d_{ijqs} < d_{ijkm}} \beta_{ijqs} + \lambda_{ijk} + \phi_{ij} = W_{ij} D_{ijkm} \quad \forall i, j \in N, k, m \in H, k = m \\
& \sum_{q | d_{ijqs} < d_{ijkm}} \beta_{ijqs} + \lambda_{ijk} - \lambda_{ijm} + \phi_{ij} = W_{ij} D_{ijkm} \quad \forall i, j \in N, k, m \in H, m \geq k \\
& \beta_{ijkm}, \phi_{ij}, \lambda_{ijk} \geq 0 \forall i, j \in N, k, m \in H. 
\end{align*}
\] (31)

where,

\[ B_{ijkm} = \{(k, m) | D_{ijkm} < D_{ijmk} || (m, k) | D_{ijmk} < D_{ijkm}, \forall i, j \in N, k, m \geq k, i, H\} \]

The dual for the primal sub problem with CAC2 added instead of rCAC2 can be written as:

\[
\begin{align*}
\min & \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \lambda_{ijk} z_k + \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \sum_{m \in H} \beta_{ijkm} (2 - z_k - z_m) \\
\text{s.t} & \sum_{q | d_{ijqs} < d_{ijkm}} \beta_{ijqs} + \lambda_{ijk} + \phi_{ij} = W_{ij} D_{ijkm} \quad \forall i, j \in N, k, m \in H, k = m \\
& \sum_{q | d_{ijqs} < d_{ijkm}} \beta_{ijqs} + \lambda_{ijk} - \lambda_{ijm} + \phi_{ij} = W_{ij} D_{ijkm} \quad \forall i, j \in N, k, m \in H \\
& \beta_{ijkm}, \phi_{ij}, \lambda_{ijk} \geq 0 \quad \forall i, j \in N, k, m \in H. 
\end{align*}
\] (35)

From the objective function (31) of the dual problem, we can formulate the Benders cut: rHIMP with rCAC2 constraint:

\[
\theta \leq \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \lambda_{ijk} z_k + \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \sum_{m \in H} \beta_{ijkm} (2 - z_k - z_m) 
\] (39)

rHIMP with CAC2 constraint:

\[
\theta \leq \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \lambda_{ijk} z_k + \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \sum_{m \in H} \beta_{ijkm} (2 - z_k - z_m) 
\] (40)

Solving the master problem with the addition of Benders cut gives the upper bound to the original problem while the sub problem gives the best lower bound to the original problem. The algorithm terminates when the difference between upper and lower bound is either zero or a negligible number \( \epsilon \).
6.1 Computational results

In this section we compare the \( rHIMP \) formulations with \( CAC2 \) and \( rCAC2 \) constraints in a Benders decomposition algorithm. We use 100 and 200 node AP dataset with 10 and 15 hubs for comparison purposes. We impose a time limit of 36000 seconds (10 hours) to solve the problem, and if the problem is not solved within the time we report the optimality gap. The computational results shows that \( rCAC2 \) is more efficient than \( CAC2 \) in solving the interdiction problem using benders decomposition algorithm.

Table 4: Comparison between CAC2 & rCAC2 in Benders decomposition

<table>
<thead>
<tr>
<th>nodes</th>
<th>p</th>
<th>r</th>
<th>( \alpha, \chi, \delta )</th>
<th>Gap</th>
<th>CPU time (secs)</th>
<th>Gap</th>
<th>CPU time (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>10</td>
<td>5</td>
<td>3.0, 7.5, 2</td>
<td>0</td>
<td>1212</td>
<td>0</td>
<td>1100</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
<td>6</td>
<td>3.0, 7.5, 2</td>
<td>0</td>
<td>1290</td>
<td>0</td>
<td>1034</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
<td>7</td>
<td>3.0, 7.5, 2</td>
<td>0</td>
<td>562</td>
<td>0</td>
<td>444</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
<td>8</td>
<td>3.0, 7.5, 2</td>
<td>0</td>
<td>310</td>
<td>0</td>
<td>233</td>
</tr>
<tr>
<td>100</td>
<td>15</td>
<td>6</td>
<td>3.0, 7.5, 2</td>
<td>86%</td>
<td>36000</td>
<td>0</td>
<td>24091</td>
</tr>
<tr>
<td>100</td>
<td>15</td>
<td>7</td>
<td>3.0, 7.5, 2</td>
<td>94%</td>
<td>36000</td>
<td>0</td>
<td>20606</td>
</tr>
<tr>
<td>100</td>
<td>15</td>
<td>8</td>
<td>3.0, 7.5, 2</td>
<td>85%</td>
<td>36000</td>
<td>0</td>
<td>14956</td>
</tr>
<tr>
<td>200</td>
<td>10</td>
<td>5</td>
<td>3.0, 7.5, 2</td>
<td>42%</td>
<td>36000</td>
<td>0</td>
<td>5779</td>
</tr>
<tr>
<td>200</td>
<td>10</td>
<td>6</td>
<td>3.0, 7.5, 2</td>
<td>26%</td>
<td>36000</td>
<td>0</td>
<td>3328</td>
</tr>
<tr>
<td>200</td>
<td>10</td>
<td>7</td>
<td>3.0, 7.5, 2</td>
<td>0</td>
<td>18572</td>
<td>0</td>
<td>2274</td>
</tr>
<tr>
<td>200</td>
<td>10</td>
<td>8</td>
<td>3.0, 7.5, 2</td>
<td>0</td>
<td>8109</td>
<td>0</td>
<td>1180</td>
</tr>
</tbody>
</table>

7 Extension to Protection problem

A natural extension of the hub interdiction median problem is the protection problem in which the defender has an option to fortify some of the hubs prior to the worst case attack by an attacker. Protection problem is a tri-level problem, where in the first level the defender chooses to protect hubs to counter the worst case attack of the attacker. The subsequent two levels of the protection problem are the interdiction problem with an additional constraint which states that when a hub is protected it cannot be interdicted. We have already noted that the conversion of bilevel interdiction problem to single level problem with closest assignment constraints brings significant improvement in computational time. Using this, we convert the trilevel protection problem to bilevel problem using the best closest assignment constraint, to solve this otherwise intractable problem. We define a new variable \( y_k \), a protection variable, which takes value 1 when a hub is protected and 0 otherwise.

\( r \)-Hub median protection problem - Multiple Allocation (\( r \)-HMPP-MA):

\[
[M]: \min Z \\
\text{s.t. } \sum_{k \in H} y_k = q \\
y_k \in \{0, 1\}
\]
\[ Z = \max \sum_{i \in N} \sum_{j \in N} \sum_{k \in H} \sum_{m \in H} W_{ij} d_{ijkm} X_{ijkm} \]  
\[ \text{s.t.} \sum_{k \in H} \sum_{m \in H} X_{ijkm} = 1 \quad \forall i, j \in N; \]  
\[ \sum_{k \in H} z_k = p - r \]  
\[ \sum_{m \in H} X_{ijkm} + \sum_{m \in H(k)} X_{ijmk} \leq z_k \quad \forall i, j \in N; \forall k \in H \]  
\[ \sum_{qs \in E_{ijkm}} X_{ijs} \leq 2 - z_k - z_m \quad \forall i, j \in N; \forall k, m \geq k \in H \]  
\[ y_k \leq z_k \quad \forall k \in H \]  
\[ z_k \in \{0, 1\}, \quad X_{ijkm} \geq 0 \quad \forall i, j \in N; \forall k, m \in H \]  

The constraint (42) ensures \( q \) hubs to be protected from the \( p \) located hubs. Constraint (48) is the rCAC2 constraint which is proven to be the most efficient among all the closest assignment constraints. Constraint (49) makes sure that when a hub is protected \( y_k = 1 \), the hub has to remain open \( z_k = 1 \).

### 7.1 Solution method

We present a solution method for the protection problem inspired by the implicit enumeration algorithm used by Scaparra and Church (2008) to solve \( p \)-facility protection problem. The underlying idea in the algorithm is based on the proposition that at least one of the nodes interdicted in the pure interdiction problem will be in the solution of the protection problem since any other combination of protected nodes will not deter the worst case attack of the attacker. The procedure is presented below:
Algorithm 1 Implicit enumeration

1: procedure Implicit enumeration–Algorithm
2: \( g \leftarrow 0 \) and \( y \leftarrow 0 \)
3: solve HIM and obtain \( z \)
4: stack of model;
5: branch
6: end procedure
7: procedure branch
8: model.push(HIM);
9: for iterations from 0 to \( r - 1 \) do
10: \( \text{model.pop;} \)
11: \( y_{z_{\text{iteration}}} \leftarrow 1 \)
12: \( \text{model.add}(y_{z_{\text{iteration}}} \leftarrow 1) \)
13: \( \text{model.solve and get } z'g \leftarrow g + 1; \)
14: \( \text{model.push;} \)
15: if \( g < r \) then \( z \leftarrow z' \)
16: branch
17: else
18: exit();
19: end if
20: end for
21: end procedure

The Implicit enumeration procedure solves at most \( r^{(q+1)} - 1/r - 1 \) sub problems. At every iteration of the implicit enumeration algorithms we solve the CHIM using benders decomposition procedure by adding constraint (22) to the master problem.

7.1.1 Results for the protection problem

Table 5 presents the computational results comparing Implicit enumeration and complete enumeration in solving the rHMPP. Lower level problems in both the procedures are solved using Benders decomposition. We see that Implicit enumeration + benders decomposition procedure is very effective in solving the large scale problem within reasonable time.

8 Conclusion

In this paper we presented a new formulation for the bilevel hub interdiction median problem. For the purpose of solving it, we explored the conversion of bilevel problem to single level using KKT conditions and Closest assignment constraints. On comparing the merit of the two methods, we pursued the single level reduction using closest assignment constraints. We presented two new closest assignment constraints and studied the dominance relationships of the constraints with the existing closest assignment constraint in literature and presented the best set of constraints for the Interdiction problem. We explored the properties of the proposed closest assignment constraints and reduced it further to an efficient form than the original constraints. We also presented computational results by solving the single level Hub interdiction problem using the closest assignment constraints in a CPLEX
Table 5: Computational results of the protection problem, (time in seconds).

<table>
<thead>
<tr>
<th>AP dataset</th>
<th>without rCAC2</th>
<th>with rCAC2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Comp. enumeration</td>
<td>Comp. enumeration</td>
</tr>
<tr>
<td>nodes</td>
<td>Comp. enumeration</td>
<td>Comp. enumeration</td>
</tr>
<tr>
<td>50 10 5 1  3,2,0.75</td>
<td>26124</td>
<td>845</td>
</tr>
<tr>
<td>50 10 5 2  3,2,0.75</td>
<td>52819</td>
<td>2922</td>
</tr>
<tr>
<td>50 10 6 1  3,2,0.75</td>
<td>17468</td>
<td>656</td>
</tr>
<tr>
<td>50 10 6 2  3,2,0.75</td>
<td>25977</td>
<td>2039</td>
</tr>
<tr>
<td>50 10 7 1  3,2,0.75</td>
<td>7510</td>
<td>413</td>
</tr>
<tr>
<td>50 10 7 2  3,2,0.75</td>
<td>7521</td>
<td>1320</td>
</tr>
<tr>
<td>100 10 5 1  3,2,0.75</td>
<td>*</td>
<td>3820</td>
</tr>
<tr>
<td>100 10 5 2  3,2,0.75</td>
<td>*</td>
<td>13176</td>
</tr>
<tr>
<td>100 10 6 1  3,2,0.75</td>
<td>*</td>
<td>3103</td>
</tr>
<tr>
<td>100 10 6 2  3,2,0.75</td>
<td>*</td>
<td>8908</td>
</tr>
<tr>
<td>100 10 7 1  3,2,0.75</td>
<td>*</td>
<td>1956</td>
</tr>
<tr>
<td>100 10 7 2  3,2,0.75</td>
<td>*</td>
<td>5356</td>
</tr>
</tbody>
</table>

* indicates out of memory status

The results established the superiority of our proposed closest assignment constraints over the one studied in the literature. In order to fully utilise the benefits of our proposed closest assignment constraint, we presented a Benders decomposition based solution procedure, which is capable of solving large datasets in a reasonable time.

We also studied a Hub protection problem, where the defender has an option to fortify a set of hubs such that to mitigate the worst case attack by an attacker. This problem is modeled in three levels and it is intractable for larger problems. The trilevel problem is reduced to bilevel using the best closest assignment constraint and the problem is solved using an Implicit enumeration + Benders decomposition procedure.
References


