ARTICLE IN PRESS

Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

European Journal of Operational Research xxx (2005) xxx–xxx

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

www.elsevier.com/locate/ejor

Discrete Optimization

The *p*-hub center allocation problem

Ann Melissa Campbell [a,*], Timothy J. Lowe [b], Li Zhang [c]

[a] Department of Management Sciences, University of Iowa, Iowa City, IA 52242-1994, United States
[b] Department of Supply Chain and Information Systems, Pennsylvania State University, University Park, PA 16802, United States
[c] Department of Mathematics and Computer Science, The Citadel, Charleston, SC 29409, United States

Received 13 August 2004; accepted 7 September 2005

Abstract

The *p*-hub center problem is to locate *p* hubs and to allocate non-hub nodes to hub nodes such that the maximum travel time (or distance) between any origin–destination pair is minimized. We address the *p*-hub center allocation problem, a subproblem of the location problem, where hub locations are given. We present complexity results and IP formulations for several versions of the problem. We establish that some special cases are polynomially solvable.

Keywords: Hub location; Networks; Complexity theory

1. Introduction

Hubs are commonly used in transportation, logistics, and telecommunication systems, such as air passenger travel, postal services, express shipments and distributed computer networks. Hubs serve as consolidation, switching and sorting centers, and allow for the replacement of direct connections between all nodes with fewer, indirect connections. In practice, the use of hubs can result in lower network costs, but it can be challenging to determine where hubs should be located or how customers should be allocated to them.

The formal study of hub location was introduced by O'Kelly [20,21] who provides a quadratic integer programming formulation for the *p*-hub median problem. The *p*-hub median problem is to locate the *p* hubs in a network and allocate non-hub nodes to hub nodes such that the sum of the costs of transporting flow between all origin–destination (o–d) pairs in the network is minimized. Hub location has attracted the

* Corresponding author. Tel.: +1 319 335 0918; fax: +1 319 335 0297.
E-mail addresses: ann-campbell@uiowa.edu (A.M. Campbell), tjl12@psu.edu (T.J. Lowe), li.zhang@citadel.edu (L. Zhang).

0377-2217/$ - see front matter
doi:10.1016/j.ejor.2005.09.024

attention of researchers from a wide variety of fields, with much of the literature focusing on the p-hub median problem. Many of the efforts focus on developing linearizations for O'Kelly's model and creating heuristics. Some of the literature focuses on uncapacitated variations, such as in [1,9,18,23], while others consider capacity restrictions on hubs, such as in [2,6,10]. See [4] or [5] for a nice overview of research on hub location problems.

A related problem is the p-hub center problem. The p-hub center problem is to locate p hubs in a network and to allocate non-hub nodes to hub nodes such that the maximum travel time (or distance) between any o–d pair is minimized (for example, see [3,7,8,14–16,22]). The p-hub center problem is important for time-sensitive or guaranteed time distribution systems, such as express mail services and emergency services. In these systems, such as at Federal Express, this maximum travel time represents the best time guarantee that can be offered to all customers. To be competitive, it is important that this value is as low as possible. There are many different variants of the p-hub center problem and the p-hub median problem which reflect operational constraints. For example, a non-hub node may be either allocated to a single hub or to multiple hubs, and the hubs (or hub links) may have a capacity restriction or be uncapacitated.

In this paper, we address the p-hub center allocation problem, which is a subproblem of the p-hub center problem. To avoid confusion, we will refer to the full p-hub center problem as the p-hub center *location* problem in the rest of this paper. The p-hub center allocation problem assumes the locations of the p hubs are known, but not the allocation of non-hub nodes to the hub nodes. Analysis of the allocation phase is extremely useful. In many real world situations, especially airline and cargo delivery systems, hub location is a long term strategic decision, and it takes considerable time and money to make changes to a preexisting hub system. However, reallocating non-hub nodes to hub nodes because of demographic changes is less expensive and more practical. Since the p-hub center location problem is NP-complete [15], many algorithms for the p-hub center location problem iteratively select hubs, then solve the resulting allocation problem. Thus, good methods for solving the allocation problem can be useful as part of solving the p-hub center location problem. Also, the study of the allocation problem can help us understand where the p-hub center location problem becomes hard.

In this paper, we present several complexity results for the p-hub center allocation problem. Specifically, Section 2 looks at the p-hub center single allocation problem where non-hub nodes are assigned to exactly one hub. We provide IP formulations for both uncapacitated and capacitated cases. We establish that some special uncapacitated cases are polynomially solvable, in particular, when $\alpha = 0$ for a hub-complete graph (where α is a discount factor that reflects economies of scale between hubs), when $p = 2$ for a hub-complete graph, and when the graph is a tree or path. Section 3 focuses on the p-hub center multiple allocation problem, where non-hub nodes can be assigned to one or more hubs. We provide IP formulations for the uncapacitated case and two variants of the capacitated case. We prove that one variant of the capacitated case is NP-complete, where the other is polynomially solvable. We also prove that the uncapacitated case is polynomially solvable. To the best of our knowledge, the capacitated p-hub center allocation problem (as well as the capacitated p-hub center location problem) has not been studied in the literature.

2. Single allocation

2.1. Notation and problem definition

Unless otherwise indicated, we will consider each problem on a *hub-complete* graph $G = (N, E)$ with node set $N = \{1, \ldots, n\}$, hub set $H = \{1, \ldots, p\}$, $H \subseteq N$, and undirected arc set E. A graph G is *hub-complete* (H-C) if there are arcs between each non-hub node and all hub nodes, and there is an arc between each pair of hubs. The travel time from node i to node j on arc (i,j) is t_{ij}, where these values obey the triangular inequality. We assume the values are symmetric, i.e., $t_{ij} = t_{ji}$. Also, we assume $t_{ii} = 0$ and $t_{ij} > 0$ if $i \neq j$.

Flow between every pair of hubs uses a rapid mode of transit as a result of economies of scale. The rapid transit mode is modeled by a discount factor α (between 0 and 1) which is a multiplier that reduces the travel time between hubs. The flow from node i to node j will be represented by w_{ij}. To simplify the formulations in this paper, unless otherwise stated, we assume $w_{ij} > 0$, $\forall i,j \in N$. Note that each hub is also a node that will be an origin and destination for flow. Unless otherwise indicated, we will assume that this flow will always be allocated to the associated hub. When we consider various types of hub capacity, we will denote the capacity of hub k by C_k.

There are three separate components for each flow: *collection* (origin node to hub node), *transfer* (hub node to hub node), and *distribution* (hub node to destination node). We denote the resulting travel time on a path between i and j after the allocation by T_{ij}. If i is assigned to hub k and j is assigned to hub h, then $T_{ij} = t_{ik} + \alpha t_{kh} + t_{jh}$ on the path $i \to k \to h \to j$. The objective is to minimize the maximum T_{ij} value.

The p-hub center single allocation problem is to allocate each of the non-hub nodes to exactly one hub such that the maximum travel time between any o–d pair is minimized. Note that for the single allocation problem, there is exactly one path between each o–d pair after the allocation.

2.2. IP formulations for the uncapacitated case

We define X_{ik} to be a binary variable that equals 1 if node i is allocated to hub k and 0 otherwise. If there is a hub at node k, then $X_{kk} = 1$.

The original integer programming formulation for the uncapacitated p-hub center location problem was given by Campbell [3]. If the set of hubs H is given, we can modify Campbell's quadratic programming formulation for the allocation problem as follows:

$$\min \max_{i,j \in N; k,m \in H} X_{ik} X_{jm} (t_{ik} + \alpha t_{km} + t_{jm}), \tag{1}$$

s.t.:

$$\sum_{k \in H} X_{ik} = 1, \quad \forall i \in N, \tag{2}$$

$$X_{kk} = 1, \quad \forall k \in H, \tag{3}$$

$$X_{ik} \in \{0,1\}, \quad \forall i \in N, \ k \in H. \tag{4}$$

The objective of the formulation is quadratic. Only if $X_{ik} = 1$ and $X_{jm} = 1$ (i.e., i is assigned to hub k and j is assigned to hub m), the travel time on the path $i \to k \to m \to j$ equals $t_{ik} + \alpha t_{km} + t_{jm}$ and equals 0 otherwise. The objective minimizes the maximum travel time between any o–d pair. Constraints (2) and (4) ensure that every node is assigned to exactly one hub.

We provide a second formulation for the uncapacitated p-hub center single allocation problem that is linear

USAP: $\min Z, \tag{5}$

s.t.:

$$Z \geqslant X_{ik} t_{ik} + \alpha t_{km} + X_{jm} t_{jm}, \quad \forall i,j \in N; \ k,m \in H, \tag{6}$$

$$\sum_{k \in H} X_{ik} = 1, \quad \forall i \in N, \tag{7}$$

$$X_{kk} = 1, \quad \forall k \in H, \tag{8}$$

$$X_{ik} \in \{0,1\}, \quad \forall i \in N; \ k \in H. \tag{9}$$

Since every node that is also a hub is allocated to that hub, there will always be flow between every pair of hubs. This makes constraint (6) valid.

Remark. Ernst et al. [7,8] provide two different formulations that are linear for the uncapacitated p-hub center single allocation problem, and one of the two formulations is similar to ours.

2.3. Complexity results for the uncapacitated case

Kara and Tansel [15] prove that the uncapacitated single allocation p-hub center location problem is NP-complete by a reduction from the dominating set problem. Ernst et al. [7] show that the uncapacitated p-hub center single allocation problem is NP-complete by reduction from the independent transversal problem [13].

In the following subsections, we identify several polynomially solvable variations of the uncapacitated p-hub center single allocation problem.

2.3.1. $\alpha = 0$ on an H-C graph

If the discount factor $\alpha = 0$ on an H-C graph, the uncapacitated p-hub center single allocation problem is polynomially solvable. The optimal solution is to assign each node to its nearest hub.

Theorem 1. *The uncapacitated p-hub center single allocation problem can be solved optimally in* $O(np)$ *time when* $\alpha = 0$.

Proof. If $\alpha = 0$, the right-hand side of (6) becomes $X_{ik}t_{ik} + X_{jm}t_{jm}$. Assigning each node to its nearest hub minimizes the right-hand side of (6) for every $i,j \in N$, thereby minimizing Z. Since we can assign each non-hub node to its nearest hub in $O(p)$ time (since we have p hubs and we only need to evaluate the allocation for the $n - p$ non-hub nodes), the uncapacitated p-hub center single allocation problem can be solved optimally in $O((n-p)p) = O(np)$ time when $\alpha = 0$. □

2.3.2. $p = 2$ on an H-C graph

In this section, we will demonstrate that the single allocation problem when $p = 2$ is polynomially solvable. We let h_1 and h_2 denote the two hubs. We use T to denote a potential final objective function value, or best time guarantee. Thus, for any feasible time guarantee T, there must exist at least one assignment of the non-hub nodes to the hub nodes such that $T_{ij} \leqslant T$, $\forall i,j \in N$. The problem of minimizing the maximum T_{ij} value is equivalent to the problem of minimizing T such that $T_{ij} \leqslant T$, $\forall i,j \in N$.

To establish the complexity of the case when $p = 2$, we first need to establish the result in Lemma 1. For any point x on arc (i,j), we denote the travel time from node i (node j) to x by $t_{ix}(t_{jx})$. If arc (i,j) is discounted, then the travel time is $\alpha t_{ix}(\alpha t_{jx})$.

Lemma 1. *For a given value T, T is a feasible time guarantee if and only if there is an assignment* $\mathscr{A}$ *of non-hub nodes to the two hubs such that there is a point C on the arc between the two hubs* h_1 *and* h_2 *(including* h_1 *and* h_2*) such that* $t_{ih_i} + \alpha t_{h_iC} \leqslant \frac{T}{2}$, $\forall i \in N$ *where i is assigned to* h_i *(*h_i *is* h_1 *or* h_2*).*

Proof. ($\Rightarrow$) If T is a feasible time guarantee, then there exists an assignment $\mathscr{A}$ of the non-hub nodes to the two hubs such that $T_{ij} = t_{ih_i} + \alpha t_{h_ih_j} + t_{jh_j} \leqslant T$, where i is assigned to h_i and j is assigned to h_j, $\forall i,j \in N$. Based on this assignment $\mathscr{A}$, suppose the longest path is between two nodes i^* and j^*, where i^* and j^* are not necessarily distinct (i.e., the longest path may be from i^* to its assigned hub and back to i^*). Suppose i^* is assigned to h_{i^*} and j^* is assigned to h_{j^*}, where $h_{i^*}, h_{j^*} = h_1$ or h_2. Then, $T_{i^*j^*} = t_{i^*h_{i^*}} + \alpha t_{h_{i^*}h_{j^*}} + t_{j^*h_{j^*}} \leqslant T$ and $T_{ij} \leqslant T_{i^*j^*}$, $\forall i,j \in N$. We can see that $t_{i^*h_{i^*}} \leqslant \frac{T}{2}$, otherwise $T_{i^*i^*} = 2t_{i^*h_{i^*}} > T$ which contradicts the assumption. By the same argument, $t_{j^*h_{j^*}} \leqslant \frac{T}{2}$ as well. If $t_{i^*h_{i^*}} + \alpha t_{h_{i^*}h_{j^*}} \leqslant \frac{T}{2}$, then locate point C at h_{j^*} (which implies $t_{i^*h_{i^*}} + \alpha t_{h_{i^*}C} = t_{i^*h_{i^*}} + \alpha t_{h_{i^*}h_{j^*}} \leqslant \frac{T}{2}$and $t_{j^*h_{j^*}} + \alpha t_{h_{j^*}C} = t_{j^*h_{j^*}} \leqslant \frac{T}{2}$). If $t_{i^*h_{i^*}} + \alpha t_{h_{i^*}h_{j^*}} > \frac{T}{2}$, it must be the case that $h_{i^*} \neq h_{j^*}$. Since $t_{i^*h_{i^*}} \leqslant \frac{T}{2}$, $t_{j^*h_{j^*}} \leqslant \frac{T}{2}$, and $T_{i^*j^*} = t_{i^*h_{i^*}} + \alpha t_{h_{i^*}x} + \alpha t_{xh_{j^*}} + t_{j^*h_{j^*}} \leqslant T$ for any point x on the arc between h_{i^*} and h_{j^*}, there exists some point C on the arc between h_{i^*} and h_{j^*} such that $t_{i^* h_{i^*}} + \alpha t_{h_{i^*} C} \leqslant \frac{T}{2}$ and $t_{j^* h_{j^*}} + \alpha t_{h_{j^*} C} \leqslant \frac{T}{2}$.

For any $i \in N$, if i is assigned to h_{i^*}, then $t_{ih_{i^*}} \leqslant t_{i^* h_{i^*}}$. Otherwise $T_{ij^*} > T_{i^* j^*}$ which contradicts the assumption. Thus, $t_{ih_{i^*}} + \alpha t_{h_{i^*} C} \leqslant t_{i^* h_{i^*}} + \alpha t_{h_{i^*} C} \leqslant \frac{T}{2}$. If i is assigned to h_{j^*}, interchange i^* and j^* above, and the proof will be similar.

($\Leftarrow$) If there is an assignment $\mathscr{A}$ of non-hub nodes to the two hubs and there is a point C on the arc between the two hubs h_1 and h_2 (including the two hubs) such that $t_{ih_i} + \alpha t_{h_i C} \leqslant \frac{T}{2}$ for any $i \in N$ and i is assigned to h_i, then $T_{ij} = t_{ih_i} + \alpha t_{h_i h_j} + t_{jh_j} \leqslant t_{ih_i} + \alpha t_{h_i C} + \alpha t_{Ch_j} + t_{jh_j} \leqslant \frac{T}{2} + \frac{T}{2} = T$, $\forall i,j \in N$. Thus, T is a feasible time guarantee. □

We will use the ideas from Lemma 1 to establish an algorithm which finds the optimal solution to the 2-hub center single allocation problem.

We know that for every pair of nodes i and j, there are at most four possible values for the travel time between i and j since we have only 2 hubs. If we explicitly compute the possible travel time values between every pair of nodes, then we will have $O(n^2)$ potential T values since we have $O(n^2)$ pairs. Among these $O(n^2)$ values, there must exist at least one value which is a feasible time guarantee (for example, the largest value of these $O(n^2)$ values). Since a binary search is faster than a linear search, we perform a binary search on these $O(n^2)$ values to find the minimum time guarantee.

For each value T', we retain the two nodes, i^* and j^*, and the hub allocation that created it such that $T' = T_{i^* j^*} = t_{i^* h_{i^*}} + \alpha t_{h_{i^*} h_{j^*}} + t_{j^* h_{j^*}}$ where i^* is assigned to h_{i^*} and j^* is assigned to h_{j^*}. With this information, we can find the location of the point C corresponding to this value T'. If $t_{i^* h_{i^*}} + \alpha t_{h_{i^*} h_{j^*}} < \frac{T'}{2}$, then locate point C at h_{j^*}. Otherwise, locate C at a point on the arc between h_{i^*} and h_{j^*} such that $t_{i^* h_{i^*}} + \alpha t_{h_{i^*} C} = \frac{T'}{2}$. Furthermore, Lemma 1 implies that if we can find an allocation such that $t_{ih_i} + \alpha t_{h_i C} \leqslant \frac{T'}{2}$, $\forall i \in N$, then T' is a feasible time guarantee, and the associated assignment is optimal for this time guarantee. Otherwise, T' is not a feasible time guarantee.

Based on these ideas, we propose Algorithm 1 which finds the optimal solution for the 2-hub single allocation problem.

Algorithm 1. 2-Hub Center Single Allocation Problem

1: Compute all potential T values by explicitly computing the possible travel time values between every pair of nodes.
2: Sort the values found in Step 1 in ascending order.
3: Starting from the largest value, use binary search to select the next candidate value T' until the binary search is exhausted.
4: For a given T', find the location of C;
5: **if** every non-hub node i can be assigned to some hub h_i such that $t_{ih_i} + \alpha t_{h_i C} \leqslant \frac{T'}{2}$ **then**
6: Go to Step 3 for a smaller value;
7: **else**
8: Go to Step 3 for a larger value;
9: **end if**

We established that Step 1 takes $O(n^2)$ time and Step 2 takes $O(n^2 \log n)$ time using a sort algorithm such as Merge Sort. Step 3 requires $O(\log(n^2)) = O(\log n)$ time for the binary search. For a given T, it takes $O(1)$ time to find the location of C in Step 4 and $O(n)$ time to find the assignment of every non-hub node in Step 5. Thus, Steps 3–9 are $O(n \log n)$. Overall, the algorithm is dominated by the sort and is $O(n^2 \log n)$. We have the following theorem:

Theorem 2. *The* 2*-hub center single allocation problem can be solved optimally in* $O(n^2 \log n)$ *time.*

Due to Theorem 2, we can also solve the 2-hub center location problem in polynomial time.

Theorem 3. *The single allocation* 2*-hub center location problem can be solved optimally in* $O(n^4 \log n)$ *time.*

Proof. There are $O(n^2)$ possible locations for the two hubs, and for every pair of hubs, the optimal allocation of the non-hub nodes can be determined in $O(n^2 \log n)$ time by Theorem 2. Thus, all of the possible solutions can be evaluated in $O(n^4 \log n)$ time. □

Theorem 3 also shows the importance of studying the allocation problem as a means to build insight into the *p*-hub center location problem.

2.3.3. Tree graph

A tree graph is a connected graph with n nodes and $n-1$ arcs. Telecommunication and distribution networks are often sparse due to the high cost of establishing connections between nodes and often resemble trees. For the uncapacitated problem on a tree, we are able to establish an exact solution approach with polynomial worst case complexity. This is useful for applications with graph structures resembling a tree, but is also useful in developing heuristics for the general problem. The study of location problems on a tree graph is quite common (see, for example, [26]) because of the structure these graphs provide.

On a tree graph, each node is no longer directly connected to each hub and there is no longer an arc between each pair of hubs. To allow for service between all nodes, though, we will assume flow can still occur from all nodes to all hubs and between all hubs. On a tree graph, there is a unique path between each pair of nodes. If nodes i, j are connected by a path of length l, we can simply set $t_{ij} = l$. Note that because of the assumption of triangle inequality on the travel times, there is never a path between nodes that is shorter than a direct connection if one exists.

If the arcs linking nodes form a tree, the results in Iyer and Ratliff [14] combined with Tamir's result [25] demonstrate that the uncapacitated single allocation *p*-hub center location problem can be solved exactly in $O(n^4 \log n)$ time. We will summarize their results and then show how their results can be modified for the allocation problem. In the centralized sorting system problem (CSS) defined by Iyer and Ratliff [14], every non-hub node is assigned to a hub node, and the location of a global center C is determined. C is an accumulation point for the hubs, i.e., C must be visited when travelling between every pair of hubs. For any $i,j \in N$, if i is assigned to a hub h_i and j is assigned to a hub h_j and the travel time between i and j is denoted by θ_{ij}, then $\theta_{ij} = t_{ih_i} + \alpha(t_{h_iC} + t_{Ch_j}) + t_{jh_j}$. Iyer and Ratliff [14] show that there exists a corresponding global center C for the uncapacitated single allocation *p*-hub center location problem on a tree graph such that the allocation problem can be solved by solving the corresponding CSS problem. They also demonstrate that this global center C must be on the path between some pair of hubs. Iyer and Ratliff [14] provide a pseudo-polynomial algorithm for solving the corresponding CSS problem on a tree graph, where the run time of their algorithm is dependent on the travel time on the longest path of the tree graph. Later Tamir [25] suggested that if we explicitly pre-compute the possible travel times between every pair of nodes (as we did in Section 2.3.2), then we can solve both problems exactly in $O(n^4 \log n)$ time.

We modify their methods to solve the allocation problem as follows. Assume G is a tree graph, and the locations of the p hubs are given. Since the desired global center C must be on the path between some pair of hubs, we can find the location of C and the corresponding time guarantee T using the method described for the $p = 2$ case in the previous section if the locations of these two hubs are known. The allocation of the non-hub nodes to the hub nodes can be modified as follows. For any $i \in N$, assign i to a hub h_i if $t_{ih_i} + \alpha t_{h_iC} \leqslant \frac{T}{2}$. This requires $O(pn)$ time. Thus, if the two hubs are known, we can solve the resulting allocation problem in $O(n^2 \log n)$ time. Since the location of the two hubs that define C are not known, we can explicitly compute all possible $O(p^2)$ pairs of hubs and then find the optimal solution among these $O(p^2)$ candidate solutions. Thus, we can solve the allocation problem in $O(p^2n^2 \log n)$ time.

Theorem 4. *The uncapacitated p-hub center single allocation problem can be solved optimally in* $O(p^2n^2 \log n)$ *time on a tree graph.*

2.3.4. Path graph

A path graph is a connected graph with n nodes and $n - 1$ arcs, and the degree of each node is either 1 or 2. In this section, we consider the problem where there are p hubs, and the arcs linking nodes form a path. Distribution networks may resemble a path, such as in shipping products along the Eastern seaboard of the US, across Canada using the transcontinental railroad, or up and down the Mississippi river on ships. Path graphs are well studied since they are helpful in providing insights into solving the general problem and their structure allows for low order polynomial algorithms. (For another example, see [19].)

Since a path graph is also a tree graph, we can apply the algorithm for the tree graph (see Section 2.3.3) and solve this problem optimally. Path graphs have a special structure, though, that allows us to solve these problems more efficiently. We will describe this structure in detail, show how it can be used to solve the variant when $p = 2$ in $O(n)$ time, and briefly describe how the algorithm can be extended when $p > 2$.

We will define some notation first. Imagine that the path graph is oriented horizontally. If there are non-hub nodes on the left side of the leftmost hub, we call them *left outer nodes*, and if there are non-hub nodes on the right side of the rightmost hub, we call them *right outer nodes*. Also, we call all the non-hub nodes between the leftmost hub and the rightmost hub *inner nodes*. When we say two hubs H_L and H_R are *h-adjacent*, we assume there is no other hub node lying on the path between H_L and H_R.

It is not hard to see that among all of the optimal solutions, there is an optimal solution such that if a non-hub node B is between two h-adjacent hubs L and R, then it is assigned to either L or R. If B is a left outer node it is assigned to the leftmost hub, and if B is a right outer node it is assigned to the rightmost hub. See Zhang [27] for the formal proof.

We will require two concepts with respect to the inner nodes between any two h-adjacent hubs: *partition* and *confliction*. Suppose s is the set of inner nodes which are between two h-adjacent hubs L and R where L is to the left of R. A *partition* of s means there is some dividing point such that all of the nodes of s on the left side of the point are assigned to L and all of the nodes of s on the right side of the point are assigned to R. If there is a partition of s on the path, then we say that there is no confliction in s. A *confliction* of s means that there is a node B of s allocated to R and some other node(s) of s to the right of B is allocated to L. We call node B a *conflicting node* in s.

Lemma 2. *If there is a confliction in the assignment of the inner nodes between two h-adjacent hubs, we can always reassign the conflicting node(s) such that the travel time on the longest path (i.e., the maximum travel time between any o–d pair) will not increase after the reassignment.*

Proof. Without loss of generality, let B be a conflicting node where all the nodes between L and B are assigned to L and B is assigned to R (i.e., B is the first node strictly to the right of L that is assigned to R). Let D be a node to the right of B which is assigned to L (a confliction). Let $t_{LR} = l$ so that $T_{LR} = \alpha l$, and assume $t_{LB} \leqslant \frac{l}{2}$. (Otherwise, if $t_{LB} > \frac{l}{2}$, then $t_{DR} \leqslant \frac{l}{2}$ and a similar constructive proof can be developed.) For a graphic example of this, see Fig. 1.

Let A be any node between L and B (if one exists). Also, let C be any node to the right of B (if one exists) where C is assigned to R. Note that A is assigned to L by the assumption above. Suppose $t_{LA} = u$, $t_{AB} = v$, $t_{BD} = w$, $t_{DR} = x$, $t_{BC} = y$, and $t_{CR} = z$.

Let I be any node to the left of L and J be any node to the right of R. Suppose I is assigned to some hub H_I and J is assigned to some hub H_J. Since H_I will be either I's nearest right or I's nearest left hub, H_I is also

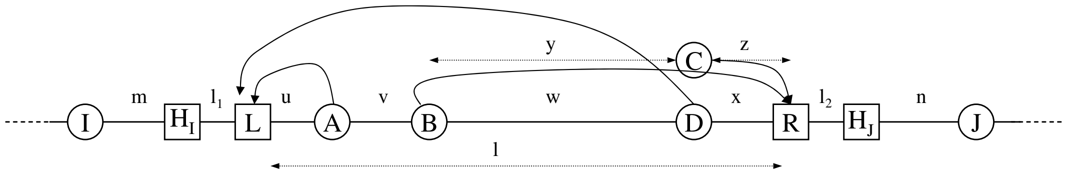

Fig. 1. Before we reassign node B.

to the left of L (or equal to L) because I is to the left of L. Similarly, H_J is to the right of R (or equal to R). Assume $t_{IH_I} = m$, $t_{JH_J} = n$, $t_{H_IL} = l_1$, $t_{RH_J} = l_2$.

We will show that by reassigning B to L, the travel time on the longest path will not increase after the reassignment. We can see that we only need to consider the travel time between B and all other nodes, because the travel time between any other pair not containing B will not change after we reassign B. If we reassign B to L and the assignment of the other nodes are unchanged, it is straightforward to establish the following inequalities: $T_{BD} \geqslant T_{BA}$, $T_{BB} \geqslant T_{BC}$, $\widehat{T_{BD}} \geqslant \widehat{T_{BA}}$, $\widehat{T_{BD}} \geqslant \widehat{T_{BB}}$, $\widehat{T_{CD}} \geqslant \widehat{T_{BC}}$, $\widehat{T_{DI}} \geqslant \widehat{T_{BI}}$, $\widehat{T_{DJ}} \geqslant \widehat{T_{BJ}}$, $T_{BB} \geqslant \widehat{T_{BB}}$, $T_{BD} \geqslant \widehat{T_{BD}}$, and $T_{BI} \geqslant \widehat{T_{BI}}$, where $\widehat{T}$ represents the distance after reassignment. Letting $\max\{list\}$ be the largest of the recorded travel times prior to the reassignment of B and $\max\{\widehat{list}\}$ be the largest after the reassignment of B, it follows that $\max\{list\} \geqslant \max\{\widehat{list}\}$.

Thus, the travel time on the longest path will not increase after we reassign B, and our result holds. □

The following lemma can be directly derived from Lemma 2.

Lemma 3. *There is an optimal solution to the p-hub center single allocation problem on a path graph where the assignment is a partition of the inner nodes between every h-adjacent pair of hubs.*

Lemma 3 implies that if $p = 2$ on a path graph, we can find the partition of the assignment to the inner nodes by a linear search. For a given partition, let A (B) be the node adjacent and to the left (right) of the partition point. Also, let I (J) be the leftmost (rightmost) outer node. The maximum travel time for the given partition will be the largest value in $\{T_{kl} : k, l \in \{I, J A, B\}\}$. Due to Lemma 3 and this observation, we have the following lemma:

Lemma 4. *The p-hub center single allocation problem on a path graph when $p = 2$ can be solved optimally in* O(n) *time.*

Proof. Since there are O(n) nodes between L and R and it takes O(1) time to find the maximum travel time in each iteration, the p-hub center single allocation problem on a path graph when $p = 2$ can be solved optimally in O(n) time. □

We have extended the idea of partitions to develop an algorithm for solving the p-hub center single allocation problem on a path graph when $p > 2$. In the algorithm detailed in [27], we evaluate each pair of h-adjacent hubs from left to right. For each hub pair, we can reduce the set of potential partition points using the distances between the nodes located between the two hubs as well as information collected for the preceding hubs.

2.4. IP formulations for the capacitated case

Next, we will address the issue of capacity restrictions on hubs. Again, for the general problem, we will assume G is an H-C graph. In the literature, there are two ways that capacity is considered in hub location problem formulations. Ebery et al. [6] consider capacity restrictions only on the volume of traffic entering a hub via collection. For example, in postal delivery, once mail has been sorted after collection, it does not need to be sorted again for distribution. Campbell [3] restricts the sum of the traffic entering any hub via collection and transfer in and out of the hub to be less than or equal to a specific capacity. This view can also be valid, as in airline applications where the capacity is determined by the number of runways and terminal sizes. Campbell [5] points out that the differences between these two types of constraints have very little influence on the solution techniques.

If we consider capacity restriction only on the volume of traffic entering a hub via collection, the basic formulation is the same as **USAP**, but we must add Constraint (10). We denote $\sum_{j\in N} w_{ij} = O_i$, $\forall i \in N$. In constraint (10), $\sum_{i\in N} O_i X_{ik}$ represents the total flow into hub k via collection

$$\sum_{i\in N} O_i X_{ik} \leqslant C_k, \quad \forall k \in H. \tag{10}$$

If we restrict the sum of the traffic entering any hub via collection and transfer to be less than or equal to the capacity of that hub, we replace constraint (10) with constraint (11). The left-hand side represents the total flow entering hub k via collection and transfer. Note that $X_{ik} + X_{jk} - X_{ik}X_{jk}$ ensures its value to be less than or equal to one. That is, if $X_{ik} = 1$ and $X_{jk} = 1$ (i.e., flow w_{ij} goes through path $i \to k \to j$), $X_{ik} + X_{jk} - X_{ik}X_{jk}$ ensures that w_{ij} (instead of $2w_{ij}$) is added to the traffic entering hub k.

$$\sum_{i\in N}\sum_{j\in N} w_{ij}(X_{ik} + X_{jk} - X_{ik}X_{jk}) \leqslant C_k, \quad \forall k \in H. \tag{11}$$

Since constraint (11) is quadratic, we propose an alternative formulation that is linear. We introduce a binary variable X_{ijk} which equals 1 if and only if $X_{ik} = 1$ and $X_{jk} = 1$. The basic formulation is still the same as **USAP** but instead of adding constraint (11), we add constraints (12)–(14) and replace constraint (9) with constraint (15). Constraints (12) and (13) ensure that $X_{ijk} = 1$ if and only if $X_{ik} = 1$ and $X_{jk} = 1$. Constraint (14) is the capacity constraint on the hub nodes where the left-hand side represents the total flow entering hub k due to collection and transfer

$$X_{ik} + X_{jk} - X_{ijk} \leqslant 1, \quad \forall i,j \in N; \;\; k \in H, \tag{12}$$

$$2X_{ijk} \leqslant X_{ik} + X_{jk}, \quad \forall i,j \in N; \;\; k \in H, \tag{13}$$

$$\sum_{i\in N}\sum_{j\in N} w_{ij}(X_{ik} + X_{jk} - X_{ijk}) \leqslant C_k, \quad \forall k \in H, \tag{14}$$

$$X_{ik}, X_{ijk} \in \{0,1\}, \quad \forall i,j \in N; \;\; k \in H. \tag{15}$$

2.5. Complexity results for the capacitated case

Since the uncapacitated p-hub center single allocation problem is NP-complete, it follows that the capacitated p-hub center single allocation problem must also be NP-complete. The capacity constraints make even the problems on many simpler graph structures, including path graphs, NP-complete. For details, see Zhang [27].

3. Multiple allocation

3.1. Problem definition

The p-hub center multiple allocation problem is to allocate each non-hub node to one or more hubs such that the maximum travel time between any o–d pair is minimized.

It is clear that the multiple allocation problem will have an objective function value no larger than that of the single allocation problem since the unique allocation constraints are relaxed in the multiple allocation problem. For example, in the airline application, if non-hub cities offer flights to several hubs rather than just one, customers may be able to travel between certain locations faster. Also, it is easier to find an optimal solution for a multiple allocation problem than it is for a single allocation problem. For example, we will demonstrate in Section 3.3 that the uncapacitated p-hub center multiple allocation problem is polynomially solvable, while the uncapacitated p-hub center single allocation problem is NP-complete [7]. Thus, the solution to a multiple allocation problem can be used as a lower bound for solving a single allocation problem.

In [3], Campbell introduces several variants of the multiple allocation p-hub center location problem. In Campbell's formulations, the decision variable X_{ijkh} is a binary variable. If $X_{ijkh} = 1$, the full flow of w_{ij} goes through path $i \to k \to h \to j$, 0 otherwise. Note that in the literature for multiple p-hub median location problems, X_{ijkh} is sometimes defined to be continuous between 0 and 1 (that is, X_{ijkh} represents the fraction of w_{ij} that travels the path $i \to k \to h \to j$). We can see that for the case where we restrict X variables to be binary, a non-hub node can be assigned to several hubs, but the flow between every o–d pair travels through exactly one path. For the case where we relax the integrality restriction on the X variables, the flow between each o–d pair may be split among several paths.

Note that for both uncapacitated and capacitated versions of the single allocation problem, the flow between o–d pairs will not be split. Since each non-hub node is assigned to exactly one hub in the single allocation problem, the amount of flow into and out of each hub is precisely determined by the allocation, and there is no advantage to splitting.

For the uncapacitated p-hub center multiple allocation problem, even if the integrality restriction on the X variables is relaxed, X values will still be binary in the optimal solution. The flow between every o–d pair will travel through the shortest path between them if there is no capacity restriction. For the capacitated p-hub center multiple allocation problem, though, we will have a lower cost solution if the integrality restriction on the X variables is relaxed. In applications such as airline networks and postal systems, it is more practical and flexible if the flow between every o–d pair is allowed to split among several paths. Also, if the integrality restriction on the X variables is relaxed, the problem becomes much easier to solve as we will see in Section 3.6. However, allowing flow to be split into several paths implies that more arcs between non-hub nodes and hub nodes may be used in a network. If it is expensive to establish or operate arcs between non-hub nodes and hub nodes, keeping the flow between each pair of nodes together may be preferred to simplify customer service and administrative functions. Thus, it is important for us to consider both types of restrictions on flow when we study the capacitated p-hub center multiple allocation problem.

We will restrict X variables to be binary in Sections 3.2–3.5 and will consider the case where X variables are continuous in Section 3.6.

3.2. IP formulations for the uncapacitated case

We can modify Campbell's [3] formulation for the multiple allocation p-hub center location problem for the allocation problem. The notation is the same as for the p-hub center single allocation problem.

UMAP: $$\min Z, \tag{16}$$

s.t.:

$$Z \geqslant X_{ijkh}(t_{ik} + \alpha t_{kh} + t_{jh}), \quad \forall i,j \in N;\ \ k,h \in H, \tag{17}$$

$$\sum_{k\in H}\sum_{h\in H} X_{ijkh} = 1, \quad \forall i,j \in N, \tag{18}$$

$$X_{ijkh} \in \{0,1\}, \quad \forall i,j \in N;\ \ k,h \in H. \tag{19}$$

If $X_{ijkh} = 1$, the travel time on the path $i \to k \to h \to j$ equals $t_{ik} + \alpha t_{kh} + t_{jh}$ and equals 0 otherwise. Constraints (18) and (19) ensure that the path from origin i to destination j must go through exactly one hub pair.

3.3. Complexity result for the uncapacitated case

The uncapacitated p-hub center multiple allocation problem is polynomially solvable. Since each node can be assigned to several hubs and there is no capacity restriction on the nodes or links, the flow between every o–d pair travels through the shortest path between them.

Theorem 5. *The uncapacitated p-hub center multiple allocation problem can be solved optimally in* $O(pn^2)$ *time.*

Proof. The following makes use of Floyd's all-pairs shortest path algorithm [11]. Suppose we renumber the nodes and make the first p nodes the hub nodes. We begin with an n by n matrix, where $T^0_{ij} = \infty$ if $i, j \notin H$, $T^0_{ij} = t_{ij}$ if one of i or $j \in H$, or $T^0_{ij} = \alpha t_{ij}$ if both $i, j \in H$. The kth step of the algorithm computes T^k_{ij} from the matrix in the previous $(k-1)$th step, where $T^k_{ij} = \min\{T^{k-1}_{ik} + T^{k-1}_{kj}; T^{k-1}_{ij}\}$. T^k_{ij} is the travel time on the shortest path between i and j at the kth step, where the path is allowed to use any subset of the first k hubs as intermediate nodes. Thus, at the end of the pth step of the algorithm, the matrix contains the travel time values on the shortest paths between o–d pairs using at least one hub from the p hubs as an intermediate node. Due to the initial values, the shortest path between every o–d pair will never be a direct path between two non-hub nodes, and by the triangular inequality, the shortest path between every o–d pair will never use more than two hubs as intermediate nodes.

Creating the pn by n matrices is $O(pn^2)$, and the time to search for the longest path in the resulting matrix is $O(n^2)$, so the total run time is $O(pn^2)$. Thus, the uncapacitated p-hub center multiple allocation problem can be solved optimally in $O(pn^2)$ time. □

Ernst et al. [7] claim a similar result for the uncapacitated p-hub center multiple allocation problem with a higher run time ($O(n^3)$), but no detailed algorithm is provided. Also, we note that Sohn and Park [24] used Floyd's algorithm on a median version of the p-hub location problem.

3.4. IP formulations for the capacitated case

To the best of our knowledge, the capacitated p-hub center multiple allocation problem (as well as its corresponding location problem) has not been studied in the literature. In [6], Ebery et al. model and present solution approaches for the capacitated multiple allocation p-hub median location problem. An important detail is discussed that is important to consider in studying our problem. Similar to the uncapacitated p-hub center multiple allocation problem, the flow between every o–d pair travels on the shortest path between them in the uncapacitated p-hub median multiple allocation problem. In a solution, the flow from a hub to another hub will travel directly on the arc between them. For the capacitated multiple allocation case, this may not always be true. Because of the capacity restriction on the hub nodes, flow w_{kk} (where k is a hub) may be routed via another hub to k, and the flow from a hub to another hub may not travel directly on the arc between them. For example, suppose we have four hubs h_1, h_2, h_3 and h_4, and $C_{h_1} = 5$, $C_{h_2} = 2$, $C_{h_3} = 1$, $C_{h_4} = 2$. Also, suppose that $w_{h_1h_1} = 4$ and $w_{h_1h_3} = 2$, while the flow between the other o–d pairs equals 0. If the capacity restriction is on collection and the flow $w_{h_1h_1}$ is allocated to h_1, then the flow $w_{h_1h_3}$ of 2 units must be routed to hub h_2 or h_4 and then transferred to h_3. Carefully modeling how flow originates from and terminates at hubs is important for the capacitated multiple allocation case. That is, we must decide whether or not to include constraints which enforce that all flow originating from or terminating at a hub must be allocated to the associated hub.

We have elected not to add such constraints for two reasons. First, multiple allocation means that a node can be assigned to several hubs, thus we should be consistent with hubs. Second, allowing flow originating from or terminating at hubs to be routed via other hubs provides greater flexibility and increases the likelihood of finding a feasible solution. In a postal delivery application, the volume of mail during Christmas at a hub (sorting center) is likely to exceed its capacity, where the capacity of a sorting center is represented by the volume of mail that can be sorted. Some of the mail originating from that hub could be sent to other hubs to be sorted. With this added flexibility, the longest path may now be one of the following four types: (a) from a non-hub node to a non-hub node, (b) from a non-hub node to a hub node, (c) from a hub node to a non-hub node, (d) from a hub node to a hub node.

If the capacity of each hub is based on the sum of collection and transfer, the IP formulation for the capacitated p-hub center multiple allocation problem is the same as **UMAP** except constraint (17) is replaced by constraints (20)–(23) (where $N - H$ is the set of non-hub nodes) and constraint (24) is added. Constraints (20)–(23) ensure that the maximum travel time (Z) between any o–d pair considers the four cases (a) through (d) listed above. It is necessary to consider each of these cases separately due to the discount factor α which applies only to arcs between hubs. Constraint (24) models the capacity constraints on the hub nodes

$$Z \geqslant X_{ijkh}(t_{ik} + \alpha t_{kh} + t_{jh}), \quad \forall i,j \in N - H; \; k,h \in H, \tag{20}$$

$$Z \geqslant X_{ijkh}(t_{ik} + \alpha t_{kh} + \alpha t_{jh}), \quad \forall i \in N - H; \; k,h,j \in H, \tag{21}$$

$$Z \geqslant X_{ijkh}(\alpha t_{ik} + \alpha t_{kh} + t_{jh}), \quad \forall j \in N - H; \; i,k,h \in H, \tag{22}$$

$$Z \geqslant X_{ijkh}(\alpha t_{ik} + \alpha t_{kh} + \alpha t_{jh}), \quad \forall i,j,k,h \in H, \tag{23}$$

$$\sum_{i \in N} \sum_{j \in N} w_{ij} \left[\sum_{h \in H} (X_{ijkh} + X_{ijhk}) - X_{ijkk} \right] \leqslant C_k, \quad \forall k \in H. \tag{24}$$

If we consider capacity restriction only on the volume of traffic entering a hub via collection, we can replace constraint (24) with

$$\sum_{i \in N} \sum_{j \in N} \sum_{h \in H} w_{ij} X_{ijkh} \leqslant C_k, \quad \forall k \in H. \tag{25}$$

3.5. Complexity result for the capacitated case

We will demonstrate that the capacitated p-hub center multiple allocation problem (CMA) is NP-complete. First we will prove that a restricted version of CMA, denoted by RCMA, is NP-complete and build on this result. RCMA is the problem of allocating every pair of non-hub nodes to exactly one hub such that the maximum travel time between any o–d pair is minimized and such that the capacity constraints on the hub nodes are satisfied. Note that RCMA has no transfer flow. If there is no transfer flow, the two versions of capacity are the same, so the result holds for both capacity restrictions given earlier.

We prove that RCMA is NP-complete by a reduction from the Bin Packing Problem (BPP) which is known to be NP-complete [13]. BPP is defined as:

Instance: Finite set U of items, a size $s(u) \in Z^+$ for each $u \in U$, a positive integer bin capacity B, and a positive integer K.

Question: Is there a partition of U into disjoint sets $U_1, U_2, \ldots, U_K$ such that the sum of the sizes of the items in each U_i is B or less?

Theorem 6. *RCMA is NP-complete even if the travel time from any non-hub node to any hub node is the same.*

Proof. RCMA is in class NP because we can verify a solution and evaluate the objective value of the solution in polynomial time. The theorem is proved by a reduction from BPP.

For an instance of BPP, we reduce it to an instance of RCMA as follows:

Let $p = K$ and $n = \lceil\sqrt{|U|}\rceil + K$ (i.e., $\lceil\sqrt{|U|}\rceil$ equals the number of non-hub nodes). Thus, we will have $\lceil\sqrt{|U|}\rceil^2 \geqslant U$ o–d pairs of non-hub nodes. We will associate each $u \in U$ with one o–d pair of non-hub nodes. For each o–d pair of non-hub nodes i and j and its corresponding item $u \in U$, let $w_{ij} = s(u)$. If $\lceil\sqrt{|U|}\rceil^2 > U$, then there will be some extra o–d pairs of non-hub nodes. For each "extra" o–d pair of non-hub nodes i and j, let $w_{ij} = 0$. We will associate each bin U_k with a hub k and let $C_k = B$. Let the flow originating from or terminating at each hub equal 0. For any non-hub node i and hub k, let $t_{ik} = t$ where t is an arbitrary positive number. This implies that if the constructed RCMA has a feasible solution, then for every o–d pair of non-hub nodes i and j, $T_{ij} = 2t$.

In general, every o–d pair that is assigned to a hub k corresponds to an item assigned to a bin U_k. We can see that the instance of BPP will have a feasible solution if and only if the newly constructed RCMA problem has a feasible solution. Since BPP is NP-complete, RCMA is NP-complete even if the travel time from any non-hub node to any hub node is the same. □

Theorem 7. *CMA is NP-complete even if the travel time from any non-hub node to any hub node is the same and* $\alpha = 0$.

Proof. If $t_{ih} = t$, $\forall i \in N - H$, $h \in H$ and $\alpha = 0$, then CMA will have an objective value that equals $2t$ if it has a feasible solution, and RCMA will have an objective value that equals $2t$ if it has a feasible solution. To prove CMA is NP-complete, we only need to prove that if a solution is feasible to RCMA, then it is also feasible to CMA, and if a solution is feasible to CMA, then we can construct a feasible solution to RCMA.

Because RCMA is a restricted version of CMA, any feasible solution to RCMA is also a feasible solution to CMA.

Suppose we are given a feasible solution to CMA, we can construct a feasible solution to RCMA as follows. For each flow w_{ij} routed via only one hub in CMA, this flow is assigned to that same hub in the RCMA solution. For each flow w_{ij} routed via two different hubs (assume i is assigned to hub k and j is assigned to hub m where $k \neq m$) in CMA, we can let w_{ij} be routed via only hub k in the RCMA solution. If flow w_{ij} travels through hub k, w_{ij} units of hub k's capacity are consumed. Thus, reassigning how this flow travels to j will not increase the demand on hub k or increase the objective function.

Thus, CMA is NP-complete even if the travel time from any non-hub node to a hub node is the same and $\alpha = 0$. □

Note that if the capacity constraints are on the hub arcs, the capacitated p-hub center multiple allocation problem is also NP-complete. The proof is very similar to the proof above. For a detailed proof, see Zhang [27].

3.6. Capacitated case with flow splitting

3.6.1. Problem definition

In Sections 3.2–3.5, we discussed the p-hub center multiple allocation problem where we assumed that the flow from node i to node j travels on exactly one path. However, if the flow between any o–d pair i and j is allowed to be split among several paths, then the integrality restriction on the X variables is removed from the IP formulation. X_{ijkh} now represents the fraction of w_{ij} that is routed via hubs k and h. For the capacitated case, if the integrality restriction on the X variables is relaxed, the resulting optimal objective function value will be no larger. Also, it provides greater flexibility if the flow between every o–d pair is allowed to split among several paths.

3.6.2. IP formulation

The cost of X_{ijkh} flow in the capacitated p-hub center multiple allocation problem is equal to $t_{ik} + \alpha t_{kh} + t_{jh}$ if $X_{ijkh} > 0$ and 0 otherwise. We introduce a new variable Y_{ijkh} for this IP formulation, where $Y_{ijkh} = 1$ if $X_{ijkh} > 0$ and 0 otherwise.

$$\min Z, \tag{26}$$

s.t.:

$$Z \geqslant Y_{ijkh}(t_{ik} + \alpha t_{kh} + t_{jh}), \quad \forall i,j \in N - H;\; k,h \in H, \tag{27}$$

$$Z \geqslant Y_{ijkh}(t_{ik} + \alpha t_{kh} + \alpha t_{jh}), \quad \forall i \in N - H;\; k,h,j \in H, \tag{28}$$

$$Z \geqslant Y_{ijkh}(\alpha t_{ik} + \alpha t_{kh} + t_{jh}), \quad \forall j \in N - H;\; i,k,h \in H, \tag{29}$$

$$Z \geqslant Y_{ijkh}(\alpha t_{ik} + \alpha t_{kh} + \alpha t_{jh}), \quad \forall i,j,k,h \in H, \tag{30}$$

$$\sum_{k\in H}\sum_{h\in H} X_{ijkh} = 1, \quad \forall i,j \in N, \tag{31}$$

$$\sum_{i\in N}\sum_{j\in N} w_{ij}\left[\sum_{h\in H}(X_{ijkh} + X_{ijhk}) - X_{ijkk}\right] \leqslant C_k, \quad \forall k \in H, \tag{32}$$

$$X_{ijkh} \leqslant Y_{ijkh}, \quad \forall i,j \in N;\; k,h \in H, \tag{33}$$

$$X_{ijkh} \geqslant 0, \quad \forall i,j \in N;\; k,h \in H, \tag{34}$$

$$Y_{ijkh} \in \{0,1\}, \quad \forall i,j \in N;\; k,h \in H. \tag{35}$$

Constraints (27)–(30) find the maximum travel time between any o–d pair. Constraint (31) ensures that the sum of the fractions of flow from i to j equals 1. Constraint (32) models the capacity restriction on the hub nodes. Constraint (33) ensures that $Y_{ijkh} = 1$ if $X_{ijkh} > 0$.

If we consider capacity restriction only on the volume of traffic entering a hub via collection, constraint (32) can be replaced by

$$\sum_{i\in N}\sum_{j\in N}\sum_{h\in H} w_{ij}X_{ijkh} \leqslant C_k, \quad \forall k \in H. \tag{36}$$

3.6.3. Complexity result

In Section 2.3.2, we used T to denote a potential best time guarantee for an allocation problem. Since solving the problem of minimizing the maximum T_{ij} value is equivalent to finding the minimum feasible time guarantee, we explicitly computed the possible travel time values between every pair of nodes and searched for the minimum feasible time guarantee. In this section, we use the same idea to demonstrate that the capacitated p-hub center multiple allocation problem with fractional flow is polynomially solvable.

Theorem 8. *The capacitated p-hub center multiple allocation problem with fractional flow is polynomially solvable.*

Proof. We can explicitly compute the travel time on all possible o–d paths. There are at most $O(p^2n^2)$ values since we have p hubs and n nodes. We can sort these $O(p^2n^2)$ values in ascending order and form a set S consisting of these sorted values. If the problem has a feasible solution, the optimal objective value Z must be a value in S. Thus, to find the minimum feasible time guarantee, we can perform a binary search on the values in S (recall that a binary search is faster than a linear search). Note that if the largest value in S is not a feasible time guarantee, then the problem has no feasible solution. This implies that the binary search should start from the largest value in S. The minimum time guarantee of the problem can be found by solving the following formulation:

$$\min T, \tag{37}$$

s.t.:

$$T \in S, \tag{38}$$

$$(t_{ik} + \alpha t_{kh} + t_{jh} - T)X_{ijkh} \leqslant 0, \quad \forall i,j \in N - H;\ \ k,h \in H, \tag{39}$$

$$(t_{ik} + \alpha t_{kh} + \alpha t_{jh} - T)X_{ijkh} \leqslant 0, \quad \forall i \in N - H;\ \ k,h,j \in H, \tag{40}$$

$$(\alpha t_{ik} + \alpha t_{kh} + t_{jh} - T)X_{ijkh} \leqslant 0, \quad \forall j \in N - H;\ \ i,k,h \in H, \tag{41}$$

$$(\alpha t_{ik} + \alpha t_{kh} + \alpha t_{jh} - T)X_{ijkh} \leqslant 0, \quad \forall i,j,k,h \in H, \tag{42}$$

(18), (32), (34).

Objective function (37) combined with constraint (38) guarantee that we find the minimum feasible time guarantee among the values in S. To find out whether or not a given value T is a feasible time guarantee, constraints (39)–(42) ensure that no assignments are considered that would result in an o–d path having a travel time greater than T. For example, if $t_{ik} + \alpha t_{kh} + t_{jh} > T$, $\forall i,j \in N - H; k,h \in H$, then constraint (39) forces $X_{ijkh} = 0$.

For a given T, verifying if there is a feasible solution that satisfies constraints (18), (32), (34), (39)–(42) is equivalent to solving a linear program. Since the binary search on the values in S requires $O(\log n)$ time, and we know that a linear programming problem is polynomially solvable [12,17], the capacitated p-hub center multiple allocation problem with fractional flow is polynomially solvable. □

Note that if the capacity constraints are on the hub arcs, the capacitated p-hub center multiple allocation problem is also polynomially solvable using a similar proof as for Theorem 8.

We note that solving a linear program may be computationally expensive for a large problem. As an alternative, the feasibility check for a fixed value of T can be accomplished by solving a network flow problem. This results in an $O(pn^5 \log n)$ algorithm that solves the problem exactly. See [27] for details.

4. Conclusions

In this paper, we have addressed the p-hub center allocation problem on a network. This problem is a subproblem of the p-hub center location problem, but it is of considerable interest on its own for time-sensitive delivery and transportation systems. We have presented complexity results and IP formulations for

Table 1
Complexity results for the p-hub center allocation problem

	Uncapacitated					Capacitated	
						Capacity restrictions on nodes	Capacity restrictions on arcs
	General graph	Special case or graph				General graph	General graph
		$\alpha = 0$	Path graph when $p = 2$	Tree graph	General graph when $p = 2$		
Single	NP-complete	Polynomial $O(p(n-p))$	Polynomial $O(n)$	Polynomial $O(p^2 n^2 \log n)$	Polynomial $O(n^2 \log n)$	NP-complete	NP-complete
Multiple	Polynomial $O(pn^2)$					NP-complete if $X_{ijkm} \in \{0,1\}$ Polynomial if $0 \leqslant X_{ijkm} \leqslant 1$	NP-complete if $X_{ijkm} \in \{0,1\}$ Polynomial if $0 \leqslant X_{ijkm} \leqslant 1$

uncapacitated and capacitated versions of the problem. We have shown that many of the more general versions of the problem are NP-hard. In cases where the underlying network has special structure or where only two hubs are present, we have provided polynomial time algorithms. A summary of the complete complexity results can be found in Table 1.

Since the general problem is intractable, heuristics may be one avenue for finding reasonable solutions on general graphs. One possible approach for solving the uncapacitated single allocation problem begins with finding a spanning tree that connects the nodes in the graph. Since the allocation problem is polynomially solvable on a tree graph, we can solve the allocation problem on the resulting spanning tree. We propose to test and evaluate this and other heuristics in the near future.

Acknowledgements

This work was partially supported by the National Science Foundation, through Grant No. 0237726 (Campbell), and served as the basis of the dissertation for Li Zhang. Much of Professor Lowe's work on this paper was done while he was a faculty member at the University of Iowa. We wish to thank our anonymous referees for providing helpful comments on an earlier version of our paper. Subsequent to our paper being in press, Arie Tamir has shown how to improve the complexity of Algrithm 1 to $O(n \log n)$. We are indebted to Professor Tamir for his assistance.

References

[1] S. Abdinnour-Helm, A hybrid heuristic for the uncapacitated hub location problem, European Journal of Operational Research 106 (1998) 489–499.
[2] T. Aykin, Lagrangian relaxation based approaches to capacitated hub-and-spoke network design problem, European Journal of Operational Research 26 (1992) 212–222.
[3] J.F. Campbell, Integer programming formulations of discrete hub location problems, European Journal of Operational Research 72 (1994) 387–405.
[4] J.F. Campbell, A survey of network hub location, Studies in Location Analysis 6 (1994) 31–49.
[5] J.F. Campbell, A. Ernst, M. Krishnamoorthy, Facility Location: Applications and Theory, Springer-Verlag, 2002.
[6] J. Ebery, M. Krishnamoorthy, A. Ernst, N. Boland, The capacitated multiple allocation hub location problem: Formulations and algorithms, European Journal of Operational Research 120 (1998) 614–631.
[7] A. Ernst, H. Hamacher, H. Jiang, M. Krishnamoorthy, G. Woeginger, Heuristic algorithms for the uncapacitated p-hub center single allocation problem, European Journal of Operational Research, forth coming.
[8] A. Ernst, H. Hamacher, H. Jiang, M. Krishnamoorthy, G. Woeginger, Uncapacitated single and multiple allocation p-hub center problems, Operations Research, in press.
[9] A. Ernst, M. Krishnamoorthy, Efficient algorithms for the uncapacitated single allocation p-hub median problem, Location Science 4 (3) (1996) 139–154.
[10] A. Ernst, M. Krishnamoorthy, Solution algorithms for the capacitated single allocation hub location problem, Annals of Operations Research 86 (1999) 141–159.
[11] R.W. Floyd, Algorithm 97: Shortest path, Communications of the ACM 5 (6) (1962) 345–370.
[12] P. Gacs, L. Lovasz, Khachian's algorithm for linear programmingMathematical Programming Study, vol. 14, North-Holland, Amsterdam, The Netherlands, 1981.
[13] M.R. Garey, D.S. Johnson, Computer and Intractability. A Guide to the Theory of NP-Completeness, W.H. Freeman and Company, New York, 1979.
[14] A.V. Iyer, H.D. Ratliff, Accumulation point location on tree networks for guaranteed time distribution, Management Science 36 (8) (1990) 958–969.
[15] B.Y. Kara, B.C. Tansel, On the single-assignment p-hub center problem, European Journal of Operational Research 125 (3) (2000) 648–655.
[16] B.Y. Kara, B.C. Tansel, The latest arrival hub location problem, Management Science 47 (10) (2001) 1408–1420.
[17] L.G. Khachian, Polynomial algorithms in linear programming, USSR Computational Mathematics and Mathematical Physics 20 (1980) 53–72.

[18] J.G. Klincewicz, A dual algorithm for the uncapacitated hub location problem, Location Science 4 (1996) 173–184.
[19] B.J. Lunday, J.C. Smith, J.B. Goldberg, Algorithms for solving the conditional covering problem on paths, Naval Research Logistics 52 (4) (2005) 293–301.
[20] M.E. O'Kelly, The location of interacting hub facilities, Transportation Science 20 (2) (1986) 92–106.
[21] M.E. O'Kelly, A quadratic integer program for the location of interacting hub facilities, European Journal of Operational Research 32 (1987) 393–404.
[22] M.E. O'Kelly, H.J. Miller, Solution strategies for the single facility minimax hub location problem, Papers in Regional Science: The Journal of the RSAI 70 (1991) 367–380.
[23] D. Skorin-Kapov, J. Skorin-Kapov, M.E. O'Kelly, Tight linear programming relaxations of uncapacitated *p*-hub median problems, European Journal of Operational Research 94 (1996) 582–593.
[24] J. Sohn, S. Park, Efficient solution procedure and reduced size formulations for *p*-hub location problems, European Journal of Operational Research 108 (1) (1998) 118–126.
[25] A. Tamir, Personal communication.
[26] A. Tamir, T. Lowe, The generalized *p*-forest problem on a tree network, Networks 22 (1992) 217–230.
[27] L. Zhang, The *p*-hub center allocation problem and the *q*-discount arc problem, Ph.D. thesis, The University of Iowa, Iowa City, Iowa, 2004.