Modelling and Analysis of Hub-and-Spoke Networks under Stochastic Demand and Congestion

Nader Azizi✉[1], Navneet Vidyarthi[2], Satyaveer S. Chauhan[2]

Abstract Motivated by the strategic importance of congestion management, in this paper we present a model to design hub-and-spoke networks under stochastic demand and congestion. The proposed model determines the location and capacity of the hub nodes and allocate non-hub nodes to these hubs while minimizing the sum of the fixed cost, transportation cost and the congestion cost. In our approach, hubs are modelled as spatially distributed M/G/1 queues and congestion is captured using the expected queue lengths at hub facilities. A simple transformation and a piecewise linear approximation technique are used to linearize the resulting nonlinear model. We present two solution approaches: an exact method that uses a cutting plane approach and a novel genetic algorithm based heuristic. The numerical experiments are conducted using CAB and TR datasets. Analysing the results obtained from a number of problem instances, we illustrate the impact of congestion cost on the network topology and show that substantial reduction in congestion can be achieved with a small increase in total cost if congestion at hub facilities is considered at the design stage. The computational results further confirm the stability and efficiency of both exact and heuristic approaches.

Keywords Hub-and-spoke, congestion, cutting plane approach, genetic algorithm

✉Corresponding author -Tel: +44-1227-827639- Email: n.azizi@kent.ac.uk

[1] Kent Business School, University of Kent, Canterbury, Kent, CT2 7PE, UK

[2] John Molson School of Business, Concordia University, Montreal, QC, H3G 1M8, Canada

1 Introduction

Instead of serving every origin-destination pair directly, a hub-and-spoke network provides service via a smaller set of links between origins-hub, pairs of hubs, and hub-destinations. The use of fewer links in the network concentrates the flow by reducing setup costs, centralizes commodity handling and sorting operations, and allows the economies of scale on transportation cost to be exploited.

Hub-and-spoke systems have various applications including air passenger and air freight transportation (e.g., Bryan and O'Kelly, 1999; Martin and Roman, 2004), less-than- truckload freight transportation (e.g., Cunha and Silva, 2007; Cheung and Muralidharan (1999)), rail freight transportation (e.g., Jeong *et al.*, 2007), urban public transportation/rapid transit (e.g., Nickel *et al.*, 2001), postal delivery (Ernst and Krishnamoorthy, 1996, 1999; Cetiner *et al.*, 2010), express package and cargo delivery (e.g.,Yaman *et al.*, 2007), and telecommunications and computer networks (e.g., Carello *et al.*, 2004) and physical distribution in supply chains (e.g., Lapierre *et al.*, 2004). Since the seminal work of O'Kelly (1986 a) several variants and extensions of the hub location problem such as *p*-hub median, uncapacitated hub location, *p*-hub centre and hub covering problem have been proposed and studied in the literature. Campbell and O'Kelly (2012) provide a detailed account of this research area.

Hub location problems are categorised into two distinctive groups namely single and multiple allocation problems. In a single allocation version of the problem, all incoming and outgoing traffic from and to every node is routed via a single hub whereas in a multiple allocation, each demand node can receive and send flow through more than one hub. Earlier studies on hub-and-spoke systems focuses on providing a tight mathematical formulation for the problem, more recent studies however, aim to develop efficient solution methods for large scale instances of the problem.

Over the years a number of approximation and exact methods have been developed to tackle various hub location problems. Examples of such methods include greedy randomized adaptive search procedure (e.g., Klincewicz, 1992), tabu search (e.g., Klincewicz, 1992), simulated annealing (e.g., Abdinnour-Helm, 2001), genetic algorithm (e.g., Abdinnour-Hel and Venkataramanan, 1998; Kratica *et al.*, 2007; Azizi et al. 2016), evolutionary algorithms (e.g., Koksalan and Soylu, 2010), neural networks (e.g., Smith *et al.*, 1996), Particle Swarm Optimisation (e.g., Azizi, 2017) general variable neighbourhood search (e.g., Ilic *et al.*, 2010), Lagrangean relaxation (e.g., Elhedhli and Wu, 2010), Benders decomposition (e.g., Camargo *et al.*, 2009b; Contreras *et al.*, 2012), branch and bound (e.g., Ernst and Krishnamoorthy, 1996,

1998b), branch and price (e.g., Contreras *et al.*, 2011c), and branch and cut (e.g., Yaman and Carello, 2005) among others. Further information about the hub location problem and its various solution techniques could be found in review articles such as Klincewicz (1998), Bryan and O'Kelly (1999), Alumur and Kara (2008), and Campbell and O'Kelly (2012).

Adopting the hub-and-spoke topology provides enterprises with the opportunity of exploiting the economies-of-scale through flow concentration and consolidation on the inter-hub links. However, studies have shown that these networks may suffer from the increasing flow at hubs which result in congestion in these facilities. Uncertainty in demand and variability in service times at hubs are the other potential causes of congestion. In urban traffic, to deal with congestion one way is to use the pricing. Pricing is a mechanism to charge the users for the negative externalities generated by the peak demand in excess of available supply. In airline transportation, empirical studies have shown that hubbing is the primary contributor to air traffic delays and congestion (Mayer and Sinai, 2003). Increasing capacity by, for instance, building new runaways to allow more take offs and landings is one way to ease the congestion and delays at major airports. For example, in 2008, O'Hare International Airport in Chicago, a hub for both United and American Airlines, opened a new runway to ease congestion and improve on-time performance. However, such strategies (e.g., building new runways) are often very expensive.

Furthermore, research has shown that uncapacitated hub location models that do not consider fixed cost associated with opening hubs and/or accounts for hub capacities produce solutions in which some hubs are subjected to heavy traffic while others rarely used (Camargo *et al.*, 2011). In short, congestion is an important strategic issue in hub-and-spoke systems that needs to be considered seriously when deciding the location of the hub facilities and allocating demand points to these hubs.

In this study, we present a model that captures the effect of congestion at hub facilities in the context of hub-and-spoke network. More specifically, our model simultaneously determines the location and capacity of the hubs and allocates demand to these facilities such that the sum of the congestion cost, the fixed cost of opening hubs and the transportation cost is minimal. The proposed model captures the trade-off between the transportation cost savings induced by the economies of scale and the cost associated with the flow congestion at hub facilities. We setup the problem as a network of spatially distributed queues (at hubs) with *Poisson* arrivals and general service time distributions (i.e., M/G/1 queues). The congestion effects are captured using the average number of users in the system. The problem is modelled as a nonlinear integer program.

To linearize the model, we use a piecewise linear approximation technique. The resulting model is then solved for small and medium size problem instances using a *cutting plane* approach, a well-known exact method. To solve larger instances, we further present a Genetic Algorithm (GA) based heuristic. We demonstrate the efficiency of the proposed heuristic by comparing its performance against the optimal solutions provided by our exact algorithm for a class of benchmark problems. Explicit consideration of the congestion cost in deciding hub locations, their capacity levels, and the flow routing decisions distinguishes this work from other hub location models.

The work of Grove and O'Kelly (1986) is one of the earliest studies to investigate the effect of congestion in hub-and-spoke networks. By simulating a single allocation hub network with fixed hub locations, Grove and O'Kelly demonstrated how schedule delays of airline systems are influenced by the amount of flow at hubs.

At least three different approaches have been proposed in the literature to model congestion at hub facilities. The first approach attempts to address the congestion by restricting the amount of flow passing through hubs using capacity constraints. The main drawback of this approach is that capacity constraints with deterministic demand do not imitate the exponential nature of the congestion effects. As a remedy to this shortcoming, Elhedhli and Hu (2005) proposed the use of a power law function to represent the congestion cost in the objective function. The value of the power-law function proposed by Elhedhli and Hu (2005) increases exponentially as more flow arrives at hubs. The function is expressed by $f(x) = ax^b$, where x is the flow at a hub and a and b are positive constants. Nevertheless the work of Elhedhli and Hu (2005) do not account for variability in demand and stochastic processing times at hubs. Along the same line, Camargo *et al.* (2009a) proposed a generalized convex cost function to model congestion in an uncapacitated multiple-allocation hub location problem under deterministic demand. Camargo *et al.* (2011) extended their model to deal with uncapacitated single-allocation hub location problem under congestion using a power-law function as well as average queuing delay function (M/M/1 queue). They present an outer approximation technique combined with Benders Decomposition to solve the model.

The second approach to capture congestion effects models a hub as a queue and uses performance measures such as average waiting time or the probability distribution of the queue length to measure congestion (Guldmann and Shen, 1997; Marianov and Serra, 2003; Elhedhli and Wu, 2010). Guldmann and Shen (1997) present a nonlinear model for hub-and-spoke network design that selects hubs and links, determines hub capacities, and assigns flows over paths, while minimizing the sum of the fixed cost, capacity cost, and the operating/congestion

cost on the links and at hubs. In the work of Guldmann and Shen (1997) hubs are modelled as M/M/1 queues and congestion is computed using the mean waiting time at hubs. Marianov and Serra (2003) present a model to find the optimal location of the hubs in airline networks. In Marianov and Serra's study hubs are modelled as M/D/c queues and congestion is captured using a probabilistic capacity constraint that limits the queue length at hub facilities. To solve the model, they proposed a Tabu search based heuristic. More recently, Elhedhli and Wu (2010) present a model where hubs are modelled as M/M/1 queues and congestion at hubs is computed as the ratio of the total flow to the surplus capacity. They present a Lagrangean heuristic to solve the non-linear mixed integer programming formulation of the problem. Similar to Elhedhli and Wu (2010) approach, we calculate the congestion as the ratio of the flow to the surplus capacity but in our study hubs are modelled as M/G/1 queues and congestion is computed using the number of users at these facilities.

In the literature, another stream of research addresses network design with stochastic demand and capacity selection but without considering the congestion effects. Examples of such studies include Correia *et al.* (2010) and Alumur *et al.* (2012). Unlike other studies in this area that often assume demand is deterministic and hub capacity is exogenous, in this paper variability in demand and service times at hubs is modelled explicitly and hub capacity decisions are considered endogenous.

Another related body of the literature is the facility location problems with immobile servers, stochastic demand, and congestion. Application of such problems ranges from location of emergency medical clinics, fire stations, automated teller machines, and internet mirror site location to design of telecommunication network and distribution networks in supply chains to name a few (Boffey *et al.*,2007; Vidyarthi *et al.*, 2009). To ensure the problem is tractable, researchers in this area often make strong assumptions such as fixing the number of facilities and/or their capacities, considering identical facilities and having exponential demand and service processes (Boffey *et al.*, 2007). To the best of our knowledge, all the references to date in this area have addressed the general discrete facility location problem without assuming any special network structure. This paper is an attempt to model the effect of stochastic demand and congestion cost on the location of the hub facilities in networks with hub-and-spoke topologies.

The remainder of this paper is organized as follows. In Section 2 we present the problem description and mathematical formulation. Linearization and the cutting plane approach will be discussed in section 3. Section 4 describes the proposed genetic algorithm based heuristic. Computational results, sensitivity analysis, and observations are presented in section 5. In

section 6 we summarize our findings and present the concluding remarks with some future research directions.

2 Model Formulation

The single allocation p-hub median problem has been studied by O'Kelly (1987), Campbell (1994b), Skorin-Kapov and Skorin-Kapov (1994), O'Kelly *et al.* (1995), Ernst and Krishnamoorthy (1996), Smith *et al.* (1996), Ebery (2001), Elhedhli and Hu (2005), and many others. To develop a model that accounts for congestion, we use the classical uncapacitated single allocation p-hub median problem due to Skorin-Kapov *et al.* (1996). The Skorin-Kapov *et al.* (1996) formulation provides the tightest linear programming bound. The model has four underlying assumptions: (1) hub arcs have no setup cost (2) distances between nodes satisfy the triangle inequality (3) flows are consolidated by hubs (direct connections between non-hub nodes are not permitted) and (4) economies of scale exist in the form of a constant discount factor and only applies to flow cost between hub nodes. Assumptions (1) and (2) imply that hub nodes are fully interconnected and the last three assumptions result in origin-destination paths that include at least one and at most two hub nodes.

The basic components of the p-hub median model is described as follows. Let $N = 1, 2, ..., n$ be the set of nodes that exchange traffic and the potential hub locations. We use k and m as indices for potential hub locations and i and j as indices for the origin and destination nodes. Therefore, paths between origin-destination (O-D) pairs are of the form of $i - j - k - m$; i and j represent the origin and destination and k and m the hubs to which i and j are respectively allocated. C_{ijkm} is the total cost of routing flow (i, j) through path (i, j, k, m) and it is given by $C_{ijkm} = \lambda_{ij}(\chi c_{ik} + \alpha c_{km} + \delta c_{mj})$ where λ_{ij} is the flow from origin i to destination j that will be routed through one or two hubs; c_{ij} is the unit transportation cost between origin i and destination j; χ is the coefficient of collection cost (per unit flow) from any origin to any hub node; δ is the coefficient of distribution cost (per unit flow) from any hub node to any destination; and α is the inter-hub discount factor.

In Skorin-Kapov *et al.* (1996) p-hub median model z_{ik} and x_{ijkm} are the two decision variables. The decision variable z_{ik} is equal 1 if node i is allocated to hub k and 0 otherwise; in particular, $z_{kk} = 1$ implies that node k is selected as a hub. The decision variable x_{ijkm} is the routing variable and equals 1 if the flow from node i to node j routed via hubs located at nodes k and m and 0 otherwise. With these notations, the formulation of the uncapacitated single-

allocation p-hub median problem (USApHMP) due to Skorin-Kapov *et al.* (1996) is presented as follows:

$$[USApHMP]: \min \quad \sum_i \sum_j \sum_k \sum_m C_{ijkm}\, x_{ijkm} \tag{1}$$

$$s.t. \quad \sum_k z_{ik} = 1 \qquad \forall i \tag{2}$$

$$z_{ik} \leq z_{kk} \qquad \forall i,k \tag{3}$$

$$\sum_k z_{kk} = p \tag{4}$$

$$\sum_m x_{ijkm} = z_{ik} \qquad \forall i,j,k \tag{5}$$

$$\sum_k x_{ijkm} = z_{jm} \qquad \forall i,j,m \tag{6}$$

$$x_{ijkm}\, , z_{ik} \in \{0,1\} \quad \forall i,j,k,m \tag{7}$$

Constraint set (2) ensures that every node is assigned to exactly one hub. Constraint (3) guarantees that a node will be assigned to an open hub. Constraint (4) ensures that exactly *p* hubs are opened in the network. Constraint (5) and (6) ensure that all the traffic between an origin-destination pair has been routed via a hub sub-network.

2.1 Modelling Congestion

In order to model congestion at hub facilities, we use the queuing delay function. Queuing based congestion captures the stochastic nature of the demand, variation in service times at hub facilities, the capacity of hubs, and represent the exponential nature of the delay as incoming flow reaches the capacity. For example, in airline networks though most flights follow a schedule, they are subject to delays both at the origin airports and during the flight which makes their arrival non-deterministic (Marianov and Serra, 2003). Upon arrival at an airport, airplanes go through three stages of service: landing at runways, service at gates, and departure through take-off runways. The service times at hubs are also highly variable and depend on several factors including types of planes and the prevailing weather conditions. Under these situations, it is reasonable to model airport hubs as queuing stations, where the queue is formed by airplanes waiting for landing and subsequent unloading/loading at the gates. In this case, congestion refers to the number of airplanes that are in the system (queuing +service) and the congestion cost is the cost per unit time incurred by the airline companies for the duration of the use of airport hubs.

A distribution network in supply chain in which trucks arrive at cross docks (or warehouses) for unloading, sorting, and loading of consignments is another example of the systems with potential congestion effects. Service times at cross docks depend on several factors including the availability of loading/unloading, sorting time of consignments and availability of personnel. Under these situations, it is reasonable to model cross docks (i.e., hubs) as queuing stations where the queue is formed by trucks waiting for unloading/loading at docks. In such cases, congestion refers to the number of trucks in the system (queuing +service).

In hub-and-spoke systems where to be concerned about the capacity and/congestion depends primarily on the type of resources and operations involved. For instance, as noted by Correia et al. 2010 in traffic logistics, the crucial capacity to consider is the inbound flow and the outbound is not important as people go in different directions depending on their destination. Similarly in other applications such as postal service where hub facilities are used for sorting operations the hub capacities also refer to the incoming flow from non-hub nodes. In such cases the incoming flow from other hubs as well as the outgoing flow can be ignored as they do not need to be processed (Ernst and Krishnamoorthy, 1999; Contreras et al 2009).

To model variability in demand, we assume the flow rate from origin *i* to destination *j* is an independent random variable that follows a Poisson process with mean λ_{ij}. Due to the superposition property of Poisson processes, the aggregate flow rate of traffic entering hub *k* via collection is also a random variable that follows a Poisson process with mean $\lambda_k = \sum_i \sum_j \sum_m \lambda_{ij}\, x_{ijkm}$. Although we model only the volume of traffic entering a hub via collection, the model can be extended to consider the traffic entering the hub via transfer as well.

We model the service times at hubs as a random variable that follows a general distribution. The service rate reflects hub capacity or the amount of flow that a hub is able to process in a given time period. In the literature, the following two approaches have been frequently used to model flexible capacity of a queuing system. The first approach is to model a single-server with flexible server capacity level (e.g., μ). In this case, the decision variable is μ which can be either continuous or discrete and the resulting model is M/G/1 queue. The second approach is to assume multiple parallel servers each with a given capacity level (μ). In this approach, the decision variable is the number of servers (e.g., *s*) to be installed at a particular location and the resulting model is M/G/s queue. The capacity can be adjusted in discrete steps by varying the number of servers. Under reasonably high service utilization, a system with *s* parallel servers (each with a capacity μ) will perform similarly to single server with capacity $s\mu$. Therefore, we choose to capture congestion effects at hubs using M/G/1 queue.

For each hub node *k*, the model is allowed to select one of the discrete capacity levels, μ_{k1}, μ_{k2},..., μ_{kL} with fixed costs of F_{k1} , F_{k2} , ..., F_{kL} respectively. These fixed costs refer to the amortized cost of acquiring capacity level at each hub facility. Let y_{kl} be a binary variable that equals 1 if hub *k* is equipped with capacity level *l*, and 0 otherwise. Each hub then can be modelled as an M/G/1 queue where mean service rate of hub *k* (with capacity level *l*) is given by

$$\mu_k = \sum_{l=1}^{L} \mu_{kl}\, y_{kl}$$

and the variance in service times is

$$\sigma_k^2 = \sum_l \ \sigma_{kl}^2\, y_{kl}$$

Let τ_k represent the mean service time at hub *k* ($\tau_k = 1/\mu_k$), ρ_k be the utilization of hub *k* ($\rho_k = \lambda_k / \mu_k$), and c_k^2 be the squared coefficient of variation of service times($c_k^2 = \sigma_k^2/\tau_k^2$). Under steady state condition ($\lambda_k < \mu_k$) and first-come-first-serve queuing discipline, the *average* waiting time (including the service time) of a unit flow at hub *k* is given by the Pollaczek-Khintchine (PK) formula:

$$\mathbb{E}[W_k] = \left(\frac{1+c_k^2}{2}\right)\frac{\tau_k \rho_k}{1-\rho_k} + \tau_k = \left(\frac{1+c_k^2}{2}\right)\frac{\lambda_k}{\mu_k(\mu_k-\lambda_k)} + \frac{1}{\mu_k} \qquad \forall k$$

The expected *total* number of users at hub *k* is obtained by multiplying the unit waiting time at hub *k* by the expected demand:

$$\mathbb{E}[L_k] = \left(\frac{1+c_k^2}{2}\right)\frac{\lambda_k^2}{\mu_k(\mu_k-\lambda_k)} + \frac{\lambda_k}{\mu_k}$$

This expression is equivalent to

$$\mathbb{E}[L_k(x,y)] = \frac{(1+\sum_l c_{kl}^2 y_{kl})\left(\sum_i \sum_j \sum_m \lambda_{ij}\, x_{ijkm}\right)^2}{2\sum_l \mu_{kl}\, y_{kl}\left(\sum_l \mu_{kl}\, y_{kl} - \sum_i \sum_j \sum_m \lambda_{ij}\, x_{ijkm}\right)} + \frac{\sum_i \sum_j \sum_m \lambda_{ij}\, x_{ijkm}}{\sum_l \mu_{kl}\, y_{kl}} \qquad (8)$$

The expression for $\mathbb{E}[L_k]$ is non-linear with respect to decision variables *x* and *y*.

2.2 Single-allocation p-hub location problem with stochastic demand and congestion

The resulting nonlinear integer programming formulation for the single-allocation p-hub location problem with stochastic demand, congestion and capacity selection is presented as follows:

$$[P]: \ \min \quad \sum_i \sum_j \sum_k \sum_m C_{ijkm}\, x_{ijkm} + \sum_k \sum_l F_{kl}\, y_{kl} + \theta \sum_k \mathbb{E}[L_k(x,y)] \qquad (9)$$

$$s.t \quad (2)-(6)$$

$$\sum_i \sum_j \sum_m \lambda_{ij}\, x_{ijkm} \leq \sum_l \mu_{kl}\, y_{kl} \qquad \forall k \tag{10}$$

$$\sum_l y_{kl} = z_{kk} \qquad \forall k \tag{11}$$

$$x_{ijkm}, y_{kl}, z_{ik} \in \{0,1\} \qquad \forall i,j,k,m \tag{12}$$

The objective function (9) minimizes the total network cost including the regular transportation cost, the fixed cost and the congestion cost. The first term in the objective function calculates the total transportation cost of the flow between all origin-destination node pairs. The second term accounts for the fixed cost (amortized over the planning period) of locating hubs with adequate service capacity level. The third term computes the total expected congestion cost at hubs and is expressed as the product of congestion cost factor per unit user θ and the expected total number of users in the system, $\mathbb{E}[L]$. Constraint set (10) is the capacity constraints at hubs. The capacity constraints can also be interpreted as the stability (steady state) condition of a queue ($\lambda_k \leq \mu_k$). Constraint set (11) ensures that a capacity level is assigned to hub *k* if node *k* is selected as a hub.

3 Model Linearization and Exact Solution Approach

The nonlinear term in the objective function [P] described above is linearized using simple transformation and a piecewise linear function. The resulting linear model has exponential number of constraints, but it is tractable using a *Cutting Plane Algorithm* (*CPA*) based exact solution approach.

3.1 Linearization

In order to linearize the objective function (9), the multiple terms in the expression for $\mathbb{E}[L_k(x,y)]$ can be rearranged and written as follows

$$\mathbb{E}[L_k(x,y)] = \frac{1}{2}\left\{(1+c_k^2)\frac{\lambda_k}{(\mu_k-\lambda_k)} + (1-c_k^2)\frac{\lambda_k}{\mu_k}\right\}$$

This is equivalent to

$$\frac{1}{2}\left\{\frac{(1+\sum_l c_{kl}^2\, y_{kl})\sum_i\sum_j\sum_m \lambda_{ij}\, x_{ijkm}}{\sum_l \mu_{kl} y_{kl} - \sum_i\sum_j\sum_m \lambda_{ij}\, x_{ijkm}} + \frac{(1-\sum_l c_{kl}^2\, y_{kl})\sum_i\sum_j\sum_m \lambda_{ij}\, x_{ijkm}}{\sum_l \mu_{kl} y_{kl}}\right\} \tag{13}$$

we define nonnegative auxiliary variables ρ_k and R_k such that

$$\rho_k = \frac{\lambda_k}{\mu_k} = \frac{\sum_i\sum_j\sum_m \lambda_{ij}\, x_{ijkm}}{\sum_l \mu_{kl} y_{kl}}$$

and

$$R_k = \frac{\lambda_k}{\mu_k - \lambda_k} = \frac{\sum_i \sum_j \sum_m \lambda_{ij}\, x_{ijkm}}{\sum_l \mu_{kl} y_{kl} - \sum_i \sum_j \sum_m \lambda_{ij}\, x_{ijkm}} \qquad \forall k$$

This implies that

$$\sum_i \sum_j \sum_m \lambda_{ij}\, x_{ijkm} = \frac{R_k}{1 + R_k} \sum_l \mu_{kl} y_{kl} = \rho_k \sum_l \mu_{kl} y_{kl} = \sum_l \mu_{kl} w_{kl}$$

where $w_{kl} = \rho_k$ if $y_{kl} = 1$ and 0 otherwise.

As there is at most one capacity level *l'* with $y_{kl'} = 1$ while $y_{kl} = 0$ for all other $l \neq l'$, the expression $w_{kl} = \rho_k y_{kl}$ can be ensured by adding the following set of constraints:

$$w_{kl} \leq y_{kl} \qquad \forall k, l$$

$$\sum_l w_{kl} = \rho_k \qquad \forall k$$

The hub utilization can be expressed as $\rho_k = \frac{R_k}{1+R_k}$ .The function $\rho_k = \frac{R_k}{1+R_k}$ is concave *w.r.t.* R_k, and it can be approximated by an infinite set of piecewise linear functions that are tangent to the function at a given set of points R_k^h i.e. $\rho_k = min_{h \in H} \left\{ \frac{1}{\left(1+R_k^h\right)^2} R_k + \frac{(R_k^h)^2}{\left(1+R_k^h\right)^2} \right\}$.

This can be written as

$$\rho_k \leq \frac{1}{\left(1 + R_k^h\right)^2} R_k + \frac{\left(R_k^h\right)^2}{\left(1 + R_k^h\right)^2} \qquad \forall k, h \in H$$

As a result, the nonlinear term of the objective function reduces to:

$$\mathbb{E}[L_k] = \frac{1}{2}\left\{\left(1 + \sum_l c_{kl}^2\, y_{kl}\right) R_k + \left(1 - \sum_l c_{kl}^2\, y_{kl}\right) \rho_k\right\}$$

$$= \frac{1}{2}\left\{R_k + \rho_k + \sum_l c_{kl}^2\, (v_{kl} - w_{kl})\right\}$$

where $v_{kl} = R_k$; if $y_{kl} = 1$ and 0 otherwise.

Because there exists at most one *l'* with $y_{kl'} = 1$ while $y_{kl} = 0$ for all other $l \neq l'$, the expression $v_{kl} = R_k y_{kl}$ can be ensured by adding the following set of constraints:

$$v_{kl} \leq M y_{kl} \qquad \forall k, l$$

$$\sum_l v_{kl} = R_k \qquad \forall k$$

The resulting linear Mixed Integer Programing (MIP) formulation is presented as follows:

$$[P_{L(H)}]\text{: min } \sum_i \sum_j \sum_k \sum_m C_{ijkm}\, x_{ijkm} + \sum_k \sum_l F_{kl}\, y_{kl} + \frac{\theta}{2} \sum_k \left\{ R_k + \rho_k + \sum_l c_{kl}^2\, (v_{kl} - w_{kl}) \right\} \quad (14)$$

$$s.t \quad (2)-(6)\,;(10)-(11)$$

$$\sum_i \sum_j \sum_m \lambda_{ij}\, x_{ijkm} \; - \sum_l \mu_{kl} w_{kl} = 0 \qquad \forall k \quad (15)$$

$$\rho_k \leq \frac{1}{\left(1 + R_k^h\right)^2} R_k + \frac{\left(R_k^h\right)^2}{\left(1 + R_k^h\right)^2} \qquad \forall k, h \in H \quad (16)$$

$$w_{kl} - y_{kl} \leq 0 \qquad \forall k, l \quad (17)$$

$$\rho_k - \textstyle\sum_l w_{kl} = 0 \qquad \forall k \quad (18)$$

$$\textstyle\sum_l y_{kl} \leq 1 \qquad \forall k \quad (19)$$

$$v_{kl} - M y_{kl} \leq 0 \qquad \forall k, l \quad (20)$$

$$R_k - \textstyle\sum_l v_{kl} = 0 \qquad \forall k \quad (21)$$

$$x_{ijkm}\,, y_{kl}, z_{ik} \; \in \{0,1\} \qquad \forall i, j, k, m, l \quad (22)$$

$$0 \leq \rho_k \leq 1; \quad 0 \leq w_{kl} \leq 1 \quad \forall k, l \quad (23)$$

$$R_k, v_{kl} \; \geq 0 \qquad \forall k, l \quad (24)$$

Stability (steady state) requirements of queuing system ($\lambda_k < \mu_k$) translate into capacity constraints, and are enforced by the constraints (15) and (17).

For coefficient of variance of service times, c = 0 (M/D/1 case) and c = 1 (M/M/1 case), the expression reduces to $\mathbb{E}[L_k]_{M/D/1} = \frac{1}{2}\{R_k + \rho_k\}$ and $\mathbb{E}[L_k]_{M/M/1} = R_k$ respectively.

This will further simplify the model as:

$$\left[P_{L(H)_{M/D/1}}\right]\text{: min } \sum_i \sum_j \sum_k \sum_m C_{ijkm}\, x_{ijkm} + \sum_k \sum_l F_{kl}\, y_{kl} + \frac{\theta}{2} \sum_k (R_k + \rho_k)$$

$$s.t \quad (2)-(6)\,;(10)-(11);(15)-(19)$$

$$x_{ijkm}\,, y_{kl}, z_{ik} \; \in \{0,1\} \qquad \forall i, j, k, l, m$$

$$R_k \geq 0; 0 \leq \rho_k \leq 1; \quad 0 \leq w_{kl} \leq 1 \quad \forall k, l$$

and

$$\left[P_{L(H)_{M/M/1}}\right]\text{: min } \sum_i \sum_j \sum_k \sum_m C_{ijkm}\, x_{ijkm} + \sum_k \sum_l F_{kl}\, y_{kl} \; + \theta \sum_k R_k$$

$s.t \quad (2)-(6)\,;(10)-(11);(15)-(19)$

$x_{ijkm}\,, y_{kl}, z_{ik} \;\in \{0,1\} \quad \forall i,j,k,l,m$

$R_k, \geq 0; 0 \leq \rho_k \leq 1; \quad 0 \leq w_{kl} \leq 1 \quad \forall k,l$

To avoid establishing hubs with long queues in the above models the value of ρ_k could be set to less than 1 e.g., ≤ 0.95.

3.2 Exact solution approach

The objective of $[P_{L(H)}]$ is a minimization, therefore, at least one of the constraints in (16) will be binding. This implies that

$$\rho_k = min_{h\in H}\left(\frac{1}{\left(1+R_k^h\right)^2}R_k + \frac{\left(R_k^h\right)^2}{\left(1+R_k^h\right)^2}\right) \qquad \forall k \; when \; y_{kl}=1$$

The nonlinearity of [P] was eliminated at the expense of an infinite number of constraints in the linear MIP model$[P_{L(H)}]$. To solve $[P_{L(H)}]$with an infinite number of constraints, we present the following cutting plane algorithm. For an initial and finite set of points $\left(R_k^h\right)_{\bar{H}\subset H}$, $[P_{L(\bar{H})}]$ is a relaxation of the full problem $[P_{L(H)}]$, hence a lower bound to$[P_{L(H)}]$ or [P] is provided by the optimal objective function value of $v(P_{L(\bar{H})})$, which is given by

$$LB = v(P_{L(\bar{H})}) = \sum_i\sum_j\sum_k\sum_m C_{ijkm}\,\bar{x}_{ijkm} + \sum_k\sum_l F_{kl}\,\bar{y}_{kl} + \frac{\theta}{2}\sum_k\left\{\bar{R}_k + \bar{\rho}_k + \sum_l c_{kl}^2\,(\bar{v}_{kl}-\bar{w}_{kl})\right\}$$

where $(\bar{x},\bar{y},\bar{z},\bar{\rho},\bar{w},\bar{R},\bar{v})$ is the solution of $[P_{L(\bar{H})}]$. Furthermore, the solution $(\bar{x},\bar{y})$of $[P_{L(\bar{H})}]$ is a feasible solution to [P] and so the upper bound is obtained as:

$$\text{UB} = \sum_i\sum_j\sum_k\sum_m C_{ijkm}\,\bar{x}_{ijkm} + \sum_k\sum_l F_{kl}\,\bar{y}_{kl} + \frac{\theta}{2}\sum_k\left\{\frac{\left(1+\sum_l c_{kl}^2\,\bar{y}_{kl}\right)\sum_i\sum_j\sum_m \lambda_{ij}\,\bar{x}_{ijkm}}{\sum_l \mu_{kl}\bar{y}_{kl} - \sum_i\sum_j\sum_m \lambda_{ij}\,\bar{x}_{ijkm}} + \frac{\left(1-\sum_l c_{kl}^2\,\bar{y}_{kl}\right)\sum_i\sum_j\sum_m \lambda_{ij}\,\bar{x}_{ijkm}}{\sum_l \mu_{kl}\bar{y}_{kl}}\right\}$$

If the best known upper bound coincides with the lower bound at a given iteration then the optimal solution is obtained and the algorithm is terminated. Otherwise, a new set of points $R_k^{h_{new}}$ are generated using the current solution $(\bar{x},\bar{y})$ as follows

$$R_k^{h_{new}} = \frac{\sum_i \sum_j \sum_m \lambda_{ij}\, \bar{x}_{ijkm}}{\sum_l \mu_{kl} \bar{y}_{kl} - \sum_i \sum_j \sum_m \lambda_{ij}\, \bar{x}_{ijkm}}$$

This new set of points is appended to $\left(R_k^h\right)_{\bar{H} \subset H}$ and is used to generate a set of cuts

$$\rho_k \leq \frac{1}{\left(1 + R_k^{h_{new}}\right)^2} R_k + \frac{\left(R_k^{h_{new}}\right)^2}{\left(1 + R_k^{h_{new}}\right)^2} \qquad \forall k, h \in H$$

The algorithmic steps of the cutting plane approach is outlined in Figure 1.

Initialization:
$UB \leftarrow \bowtie; LB \leftarrow -\bowtie; q \leftarrow 0$
Choose an initial set of points R^h
While $UB \neq LB$ do
Solve $\left[P_{L(H^q)}\right]$ to obtain $\left(\bar{x}^q, \bar{y}^q, \bar{z}^q, \bar{\rho}^q, \bar{w}^q, \bar{R}^q, \bar{v}^q\right)$
Update the lower bound: $LB^q \leftarrow v\left(P_{L(\bar{H}^q)}\right)$
Update the upper bound: $UB^q \leftarrow min\{UB^{q-1}, Z(\bar{x}^q, \bar{y}^q, \bar{z}^q)\}$
Get new points: $R_k^{h_{new}} = \frac{\sum_i \sum_j \sum_m \lambda_{ij} \bar{x}_{ijkm}}{\sum_l \mu_{kl} \bar{y}_{kl} - \sum_i \sum_j \sum_m \lambda_{ij} \bar{x}_{ijkm}} \quad \forall k$
Generate new cuts: $\rho_k \leq \frac{1}{\left(1+R_k^{h_{new}}\right)^2} R_k + \frac{\left(R_k^{h_{new}}\right)^2}{\left(1+R_k^{h_{new}}\right)^2} \qquad \forall k, h \in H$
Append new cuts: $H^{q+1} \leftarrow H^q \cup \{h_{new}\}$
$q \leftarrow q + 1$
End while

Fig.1 The cutting plane algorithm

As will be shown in our computational results, the above models formulation could be used to solve small to medium size problem instances to optimality. Due to the limitation in using exact methods such as cutting plan approach in solving large problem instances of the proposed model, one way forward is to design an efficient metaheuristic. In this study, we present a metaheuristic based on a well-known evolutionary algorithm of Genetic Algorithm. These algorithm is discussed in the following section.

4 Genetic algorithm

Genetic algorithm (GA) is an efficient metaheuristic based on the evolutionary idea of natural selection and genetics. Various types of the algorithm have been successfully applied to a wide range of combinatorial optimization problems (Salhi, 2017). The works of Kratica *et al.* (2007) and Koksalan and Soylu (2010) are examples of GAs application in hub location problems. In

the following, we briefly describe the GA based heuristic used in this study to solve our model. The proposed GA begins to search the solution space by randomly generating a population of solutions. Then two parent chromosomes from the current population are selected one at a time to generate offspring chromosomes. The newly generated chromosomes are constructed via crossover and mutation operators. Upon completion of the (offspring) population, members of the current as well as those in the newly generated population are ranked in descending and ascending order respectively. Elements of the two populations are compared one to another and those inferior members of the current population are replaced by chromosomes with higher quality in the offspring population. The algorithmic steps of the proposed GA is outlined in Figure 2. In the following subsections, we elaborate on the solution representation, initial population generation, crossover and mutation operators.

4.1 Solution representation

In our GA, a solution is represented by an array (string) with the length of $1 \times N$ where N corresponds to the number of nodes in the network. For instance, a solution to a problem with 10 nodes and 3 hubs could be represented as [1 3 3 3 5 3 1 1 5 5]. Decoding the string from left to right, the first location corresponds to node number 1, the second location corresponds to node number 2, and so on. Each location on the string (i.e., a gene) contains a number which may or may not be the same as the "location number". Each of these numbers refers to a hub in the network. Each hub node is allocated to itself. For example, nodes 1, 3, and 5 are assumed to be hubs and therefore, they are allocated to themselves and other nodes in the network have been assigned to one of these hubs.

4.2 Initial population

Solutions of the initial population are generated randomly. The procedure to generate a member of the population (i.e., chromosome) is presented as follows. First, an empty one-dimensional array of length N is constructed. The location of hubs is then determined by generating p (unidentical) random integers between 1 and N. Each of these p integers is assigned to its corresponding position in the chromosome. For example, if the first random number is "3" then it occupies the third position (from left) in the chromosome. To complete the chromosome, the rest of the (non-hub) nodes are randomly allocated to the p hubs in such way that at least one node from the remaining $N - p$ nodes is assigned to each hubs. The proposed solution representation scheme and initial population generation procedure ensures the feasibility of the solutions.

```
Initialization:
  Set the GA parameters: crossover probability p_c ; mutation probability p_m ; population
     size pop.size; and  the computational time
     t ← 0
    Generate an initial population: P (t)
    Evaluate the initial population: P (t)
  Do while (the termination condition is not met)
    t←t+1
    Select two parents randomly from P (t − 1)
    Generate a random number, Random1 ∈ {0, 1}
    If Random_1 ≤ P_c then
      Perform crossover
      Perform mutation
      Evaluate offsprings
      If Offspring's fitness function is improved upon mutation then
        Add the mutated offspring to the new population
      Else
        Add the crossovered offspring to the new population
      End If
    Else
      Select one of the two parents randomly
      Generate a random number
      If Random_2 ≤ P_m then
        Perform mutation
        If chromosome's or parents fitness function is improved upon mutation then
          Add the mutated offspring to the new population
        Else
          Add the parent to the population
        End If
      Else
        Add the selected parent to the new population
      End If
    End If
    Rank the parents in population P (t) in descending order.
    Rank the offspring population O(t) in ascending order.
    Insert the superior members of P (t) and O(t) into P (t + 1)
    Evaluate P (t + 1)
  Loop
```

Fig.2 The pseudo code of the proposed genetic algorithm

4.3 Crossover operation

The classical GA's crossover operators (e.g., two-point crossover) that combine parents' chromosomes to construct new offspring often generate infeasible solutions which slows down the search process. This phenomena is commonly blamed for poor performance of the genetic algorithm based heuristics in solving some combinatorial optimization problems. In this study, we tailored a special type of crossover operator to produce such offspring chromosomes that are safely decoded into feasible solutions. Details of the crossover operation are briefly

described as follows. To generate an offspring, first a template chromosome (i.e., an empty array) with the length of the number of nodes in the problem in hand is constructed and then two parents are selected randomly from the current population. The genetic structure of the offspring chromosome is assembled by taking one of the two parents and transferring the first gene from the parent into the offspring template chromosome. This gene (i.e., a hub) is placed in the offspring chromosome array where the location corresponds to its value. For instance, if the value of the selected gene is 3 then it is placed in the third location of the offspring array. Once the gene is transferred, the parent chromosome is scanned and all other genes with the same value (e.g., 3) are similarly moved to their corresponding locations in the offspring chromosome. The other parent is then selected and the above steps are repeated. This process continues by consecutively selecting the remaining genes in parent chromosomes and embedding them into the offspring chromosome. The crossover operation stops when the offspring chromosome is completely constructed.

4.4 Mutation operations

To mutate a chromosome, we randomly select two unidentical genes that represent non-hub nodes in the network and swap their positions. For example, if the selected chromosome for mutation is [1 3 3 3 5 3 1 1 5 5], then we select two unidentical genes from the non-hub nodes i.e., 2, 4, 6, 7, 8, 9, and 10 randomly. If the selected non-hub nodes are 7 and 9 with genes 1 and 5, then swapping their position yields the mutated offspring [1 3 3 3 5 3 5 1 1 5]. Following the mutation operation, the fitness values of the original offspring and the mutated chromosomes are compared. The chromosome with the lower cost is inserted into the new population. This approach is different from traditional mutation operators that are usually applied with low probability on any chromosome in the population. In our case, the mutation operator will either improve a chromosome, or leave it unchanged.

5 Computational results

270 test problems are derived from U.S. Civil Aeronautics Board (CAB) (O'Kelly, 1987) and Turkish (TR) datasets (Yaman *et al.*, 2007). The algorithms were coded in C and run on a Dell Intel Core PC with 2.40 GHz processor with 2 GB of RAM. The MIP problems were solved using the callable library of CPLEX 11.2. The MIP problems are solved to optimality (with a gap of 10^{-6}) using the exact approach. For the GA, we report the best solution obtained after 20 replications of the algorithm for every instance.

Test problems

Using the CAB dataset, we generate 216 problem instances by setting the number of nodes (N) to 10, 15, 20, and 25, the number of hubs (p) to 3 and 4, the inter-hub discount factor (α) to 0.2, 0.4 and 0.8, the congestion cost factor (θ) to 1, 20, and 50, and the coefficient of variance of service times (c) to 0 (M/D/1 case), 1 (M/M/1 case), and 2 (M/G/1 case). The average flow rate/demand λ_{ij} and the unit transportation cost c_{ij} between each pair of nodes (i, j) are obtained from the dataset. The collection and distribution cost coefficients are set to $\chi = \delta = 1$ per unit. For every potential hub, we generate three capacity levels: small (S), medium (M) and large (L); the associated fixed costs are set to 150 (S), 200 (M) and 250 (L) and the capacity levels are decided using $\frac{\sum_i \sum_{j \neq k} \lambda_{ij}}{p} + \beta A_l \sum_i \sum_{j \neq k} \lambda_{ij}$, where k is the hub in a one-hub network with n nodes that receives the least total flow. The coefficient β is set to 0.21, 0.22, 0.23, 0.24 for 10, 15, 20, and 25 nodes respectively. A_l is a constant that takes the value of -1, 0, and 1 for l = 1(S), 2(M), and 3(L) respectively.

The TR dataset consists of flow and distance between 81 cities in Turkey. We generate 54 instances of the problem by setting N to 25, 55, and 81, p to 3 and 4, α to 0.2, 0.5 and 0.8, θ to 1, 20, and 50, c to 0, 1 and 2, and $\chi = \delta = 1$ per unit. Similar to that in CAB dataset, we generate three capacity levels: small, medium and large for every potential hub in the network; the corresponding fixed costs to each capacity level are 50 (S), 100 (M) and 150 (L). The capacity levels are decided according to $\frac{\sum_i \sum_{j \neq k} \lambda_{ij}}{p} + \beta A_l \sum_i \sum_{j \neq k} \lambda_{ij}$. Similar to that in CAB dataset, k is the hub in a one-hub network with n nodes that receives the least total flow. The coefficient β is set to 0.20, 0.25, and 0.27 for problem with 25, 55, and 81 nodes respectively.

5.1 An illustrative example

One of the objectives of this research is to compare the network configurations and their associated costs (e.g., regular and total transportation costs) of a single allocation p-hub median problem with and without congestion effects consideration. For this purpose we solve a problem from the CAB dataset with N = 15 nodes, p = 3 hubs, and inter- hub discount factor (i.e., α) of 0.4 to optimality (with a gap of 10^{-6}) using the exact method. Table 1 summarizes the results for various unit of congestion cost θ (i.e., the θ is set to 0, 1, 10, 20, 30, 50, 100, and 200) under three scenarios: M/D/1 (c = 0), M/M/1 (c = 1), and M/G/1 (c = 2). The computational results in Table 1 include the total objective function value (*OBJ*), the transportation cost (*TC*), the fixed cost (*FC*), the congestion cost (*CC*), the total number of users in the system ($\mathbb{E}[L]$), the hub locations and their capacities, the aggregate flow arrival rate

at hubs (λ_k), the service capacity at hubs (μ_k), the average hub utilization (ρ_k), the average queue length (L_k), the number of iterations of the cutting plane algorithm (#ITR), and the CPU time in seconds (CPU(s)).

Figure 3 illustrates the effect of changing congestion cost factor θ on the total expected number of users in the system $\mathbb{E}[L]$. Figure 4 shows the trade-off between the total expected number of users $\mathbb{E}[L]$and the sum of fixed costs and expected transportation costs. The insights are summarized as follows:

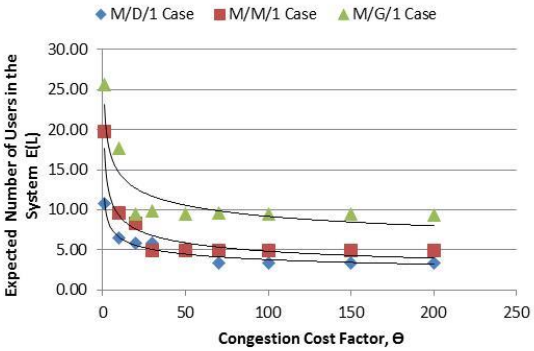

Fig.3 The effect of changing congestion cost factor on the total expected number of users in the system

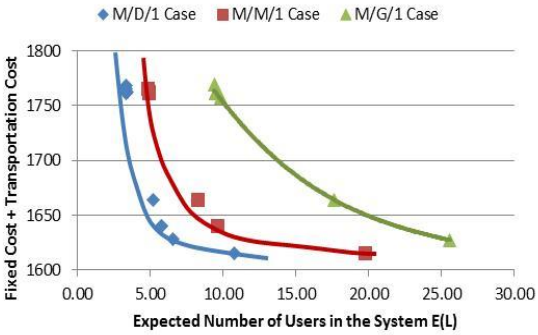

Fig.4 The trade-off between the total expected number of users in the system and the sum of transportation cost and fixed cost

Observation 1: *The hub-and-spoke network configuration (location, capacity, and allocation of nodes to hubs) that considers congestion effects differs from the traditional configurations that ignores congestion and/hub capacity.*

The optimal network for the classical single allocation *p*-hub median problem with 15 nodes, 3 hubs and inter-hub discount factor of 0.4 (i.e., *α)* recommends Chicago, Dallas-FW and Los Angeles as the optimal locations for the three hub facilities. The network configuration also show that while 10 out of the 15 cities are allocated to Chicago hub, the Los Angeles hub does not serve any of the demand nodes (cities); the two other cities are assigned to the remaining hub, Dallas-FW. This is understandable as the objective of the classical version of the problem is just to minimize the transportation cost. Table 1 presents the configuration of the hub-and-spoke networks for different values of the congestion cost factor (i.e., θ). The optimal network without congestion (θ = 0, c = 1) i.e., the capacitated version suggests Chicago (Large), Los Angeles (Medium), and Memphis (Medium) as the location of the hub facilities, whereas the model with congestion (θ = 30, c = 1) recommends to open hubs at Chicago (Large), Cleveland (Large), and Dallas-FW (Large). From this observation, it can be concluded that the topologies of these three networks (i.e., classical, capacitated, and capacitated with congestion) differ both

in terms of recommended hub locations and the allocation of demand to these facilities.

Further examination of Table 1 confirms that the network configurations (i.e. hub location, their capacity levels, and allocation of nodes to hubs) changes as the values of the congestion cost factor varies. The results also show that as the congestion cost factor increases, the model tries to reallocate non-hub nodes in order to balance the amount of flow passing through hubs and ultimately reduce the overall congestion in the network. For example, at $\theta = 0$ and $c = 1$, the total flow passing through hubs are: Chicago (Large; 1,204,290), Los Angeles (Medium; 399,236), and Memphis (Medium; 768,188), whereas when congestion effect is considered ($\theta = 20$, $c = 1$) the selected hubs and their flow are: Chicago (Large; 797,140), Cleveland (Large; 828,076), and Dallas-FW (Large; 739,725). The results in Table 1 further show that for very high values of congestion cost factors the configurations are not significantly different for M/D/1, M/M/1, and M/G/1 cases.

Although establishing hubs with large capacity is expensive especially at the beginning, the decision provides the firm with the competitive advantage of routing the flow in a timely and responsive manner. In short, capacity selection, and allocation/routing of flow are interrelated decisions and should be made in conjunction rather than isolation.

Observation 2: *Substantial reduction in congestion can be achieved with a small increase in total costs (fixed cost + transportation cost) by incorporating congestion cost in the model.*

Examining Figure 3 show that by incorporating the congestion cost factor into the model (θ = 0 to 10 to 20 to 30), the average queue length at hubs $\mathbb{E}[L]$decreases substantially at the beginning which results in relatively low level of congestion in the network. Further examination of Figure 3 reveals that large reduction in the congestion can be achieved without large increase in the fixed cost and transportation cost (see also the steepness of the left part of the curve in Figure 4). This is also evident from Table 1 where for M/G/1 case, the total expected queue length $\mathbb{E}[L]$decreases from 4988.52 to 25.56 with very small value of $\theta = 1$. The rationale behind this significant reduction in overall congestion is that with increase in congestion cost (a) hubs with higher capacity levels are utilized (b) flow is distributed more evenly across the existing hubs and (c) the average hub utilization is increased.

Observation 3: *For a fixed value of coefficient of variance of service times c, an increase in congestion cost factor, θ, results in (i) a decrease and then an increase in the transportation costs (TC); (ii) an increase in congestion costs (CC); (iii) a decrease in total expected queue*

length $\mathbb{E}[L]$*; (iv) a decrease in average hub utilization (ρ); (v) a decrease in queue length at hubs (*L_k*); (vi) a reduction in hub congestion and (vii) an increase in computing time of the algorithms.*

For a fixed value of coefficient of variance of service times *c*, as the congestion cost factor θ increases, the queue length at hubs and consequently the total expected queue length in the system $\mathbb{E}[L]$decreases. Increase in the congestion cost factor also causes (naturally) the total congestion cost to grow. For instance, in M/M/1 case, as θ increases from 1 to 10 to 20, the expected queue length,$\mathbb{E}[L]$, decreases from 19.81 to 9.62 to 8.30, and the congestion cost increases from 19.81 to 96.21 to 166.1. The model tradeoffs the congestion cost against the transportation cost and the fixed cost through (1) reallocation of nodes to hubs in an attempt to balance the flows at hubs (2) hubs capacities improvement and/or (3) change in the potential hub locations. Reallocating the flow initially reduces and then increases the transportation cost (e.g. as θ increases from 0 to 1 to 10, the transportation cost (*TC*) decreases from 938.2 to 914.9 and then increases to 940.2). As θ increases from 0 to 1, the total fixed cost of establishing a hub also increases from 650 to 700 because of the change in hubs capacity levels. As a result of the above changes, the average utilization is more even across the various hub locations. We also observed that the length of the computational times in various problem instances is affected by the congestion cost factor, the quality of the solution of LP relaxation and the number of iterations of the branch and bound.

Observation 4: *For a fixed value of the congestion cost factor θ, an increase in coefficient of variance of service times (c) results in (i) an increase in transportation cost (TC); (ii) an increase in congestion cost (CC); (iii) an increase in total expected queue length* $\mathbb{E}[L]$*; (iv) a decrease in average hub utilization (ρ); (v) a decrease in queue length at hubs (*L_k *); (vi) an increase in hub congestion; and (vii) an increase in computation time of algorithms.*

As the variability in service times increases, the total expected queue length increases which cause the congestion cost to increase. In response to an increase in service times, the proposed model reallocates and/or reroutes the flow in order to reduce the congestion at hubs. For example, as shown in Table 1, with θ = 10, as the variability in service time increases from *c* = 0 to 1 to 2, the total expected queue length increases from $\mathbb{E}[L]$= 6.54 to 9.62 to 17.67, which cause the congestion cost to rise from 65.52 to 96.21 to 176.7 unit. In this case, the model reallocates the flow by changing the assignment of the non-hub nodes in the network to minimize congestion. This can be verified by examining the flow that passes through the

Table 1 Comparison of the Hub Location and Network Configuration for M/D/1, M/M/1, and M/G/1Cases: An Illustrative Example - 15 Node, 3 Hubs, α = 0:4, CAB Dataset

θ	c	OBJ	TC	FC	CC	E(L)	Hub opened (capacity level)	λ_k	μ_k	ρ_k	L_k	(#ITR)	CPU(s)
0	0	1588.2	938.2	650	0	999.68	Chicago(L)	1,204,290	1,275,193	0.94	8.88	0	92.9
							Los Angeles (M)	399,236	768,188	0.52	0.80		
							Memphis (M)	768,188	768,188	1.00	990		
	1	1588.2	938.2	650	0	1996.89	Chicago (L)	1,204,290	1,275,193	0.94	16.81	0	92.7
							Los Angeles (M)	399,236	768,188	0.52	1.08		
							Memphis (M)	768,188	768,188	1.00	1979		
	2	1588.2	938.2	650	0	4988.53	Chicago (L)	1,204,290	1,275,193	0.94	40.61	0	92.7
							Los Angeles (M)	399,236	768,188	0.52	1.92		
							Memphis (M)	768,188	768,188	1.00	4946		
1	0	1625	914.9	700	10.9	10.81	Chicago(L)	1,204,290	1,275,193	0.94	8.88	0	71
							Los Angeles (M)	276,108	768,188	0.36	0.46		
							Memphis (L)	844,544	1,275,193	0.69	1.48		
	1	1634.7	914.9	700	19.81	19.81	Chicago (L)	1,204,290	1,275,193	0.94	16.98	9	1170
							Los Angeles (M)	276,108	768,188	0.36	0.56		
							Memphis (L)	844,544	1,275,193	0.69	2.26		
	2	1653	927.4	700	25.56	25.56	Chicago (L)	1,130,983	1,275,193	0.89	18.28	1	141
							Los Angeles (M)	276,108	768,188	0.36	0.86		
							Memphis (L)	957,851	1,275,193	0.75	6.94		
10	0	1694.3	928.7	700	65.52	6.54	Chicago(L)	1,124,035	1,275,193	0.88	4.15	4	75.1
							Los Angeles (M)	276,108	768,188	0.36	0.46		
							Memphis (L)	964,800	1,275,193	0.76	1.98		
	1	1736.4	940.2	700	96.21	9.62	Chicago (L)	1,050,728	1,275,193	0.82	4.68	5	1179
							Los Angeles (M)	276,108	768,188	0.36	0.56		
							Memphis (L)	1,038,107	1,275,193	0.81	4.38		
	2	1840.7	964.0	700	176.7	17.67	Chicago (L)	1,050,728	1,275,193	0.82	10.47	6	1534
							Los Angeles (M)	399,236	768,188	0.52	1.93		
							Memphis (L)	914,978	1,275,193	0.72	5.27		
20	0	1756.4	940.2	700	116.2	5.81	Chicago(L)	1,050,728	1,275,193	0.82	2.75	5	1196
							Los Angeles (M)	276,108	768,188	0.36	0.46		
							Memphis (L)	1,038,107	1,275,193	0.81	2.6		
	1	1830.1	964.0	700	166.1	8.30	Chicago (L)	1,050,728	1,275,193	0.82	4.68	1	378
							Los Angeles (M)	399,236	768,188	0.52	1.08		
							Memphis (L)	914,978	1,275,193	0.72	2.54		
	2	1950.4	1011.0	750	189.4	9.47	Chicago (L)	797,140	1,275,193	0.63	3.23	1	374
							Cleveland(L)	828,076	1,275,193	0.65	3.66		
							Dallas-FW (L)	739,725	1,275,193	0.58	2.58		
30	0	1814.5	940.2	700	174.3	5.81	Chicago(L)	1,050,728	1,275,193	0.82	2.75	5	1317
							Los Angeles (M)	276,108	768,188	0.36	0.46		
							Memphis (L)	1,038,107	1,275,193	0.81	2.60		
	1	1908.1	1011.0	750	147.0	4.90	Chicago (L)	797,140	1,275,193	0.63	1.67	1	366
							Cleveland(L)	828,076	1,275,193	0.65	1.85		
							Dallas-FW (L)	739,725	1,275,193	0.58	1.38		
	2	2052.1	1007.2	750	294.9	9.83	Chicago (L)	857,469	1,275,193	0.67	4.12	2	289
							Cincinnati(L)	828,124	1,275,193	0.65	3.66		
							Dallas-FW (L)	679,349	1,275,193	0.53	2.05		
50	0	1923.1	964.0	700	259.1	5.18	Chicago(L)	1,050,728	1,275,193	0.82	2.75	1	942
							Los Angeles (M)	662,732	1,275,193	0.52	0.80		
							Memphis (L)	914,978	1,275,193	0.72	1.63		
	1	2006.1	1011.0	750	245.0	4.90	Chicago (L)	797,140	1,275,193	0.63	1.67	1	254
							Cleveland(L)	828,076	1,275,193	0.65	1.85		
							Dallas-FW (L)	739,725	1,275,193	0.58	1.38		
	2	2234.6	1011.0	750	473.5	9.47	Chicago (L)	797,140	1,275,193	0.63	3.23	1	210
							Cleveland(L)	828,076	1,275,193	0.65	3.66		
							Dallas-FW (L)	739,725	1,275,193	0.58	2.58		
100	0	2102.3	1012.2	750	340.1	3.40	Chicago(L)	857,469	1,275,193	0.67	1.36	0	136
							Cincinnati(L)	767,747	1,275,193	0.60	1.06		
							Dallas-FW (L)	739,725	1,275,193	0.58	0.98		
	1	2256.9	1012.2	750	494.7	4.95	Chicago (L)	857,469	1,275,193	0.67	2.05	1	550
							Cincinnati(L)	767,747	1,275,193	0.60	1.51		
							Dallas-FW (L)	739,725	1,275,193	0.58	1.38		
	2	2708.1	1011.0	750	947.1	9.47	Chicago (L)	797,140	1,275,193	0.63	3.23	1	253
							Cleveland(L)	828,076	1,275,193	0.65	3.66		
							Dallas-FW (L)	739,725	1,275,193	0.58	2.58		
200	0	2439.3	1014.9	750	674.3	3.37	Chicago(L)	754,054	1,275,193	0.59	1.02	1	207
							Cincinnati(L)	828,124	1,275,193	0.65	1.25		
							Dallas-FW (L)	782,764	1,275,193	0.61	1.10		
	1	2742.7	1014.9	750	977.8	4.89	Chicago (L)	754,054	1,275,193	0.59	1.45	1	214
							Cincinnati(L)	828,124	1,275,193	0.65	1.85		
							Dallas-FW (L)	782,764	1,275,193	0.61	1.59		
	2	3648.0	1019.3	750	1879	9.39	Chicago (L)	814,430	1,275,193	0.64	3.46	1	325
							Cincinnati(L)	767,747	1,275,193	0.60	2.88		
							Dallas-FW (L)	782,764	1,275,193	0.61	3.05		

following hubs: the flow passes through Chicago changes from 1,124,035 to 1,050,728; through Los Angeles changes from 276,108 to 399,236; and through Memphis changes from 964,800 to 1,038,107 to 914,978. The location and the capacity of the hubs remain unchanged: Chicago (L), Los Angeles (M), and Memphis (L). The average hub utilization first decreases from 0.67 to 0.66 and then increases to 0.69. The average queue length at hubs, L_k, increases from 4.15 to 4.68 to 10.47 at Chicago hub; from 0.46 to 0.56 to 1.93 at Los Angeles hub; and from 1.93 to 4.38 to 5.27 at Memphis hub.

We also observed that as the nonlinear component of the objective function dominates, the cutting plane algorithm requires more iterations to converge and therefore, the CPU time increases from 75 to 1179 to 1534 seconds. In some cases, the proposed model prescribes increasing the service capacity of hubs and/or changing the locations of the hubs and routings of flows while trading off the congestion cost against the fixed cost and the transportation cost. For example, for μ = 20, as the variability in service times increases from c = 1 to 2, the hub locations and their capacities changes from Chicago (L), Los Angeles (M), and Memphis (L) to Chicago (L), Cleveland (L), and Dallas-FW (L).

5.2 The effect of Adding a Priori Set of Cuts on the Performance of Exact Solution Approach

Our second set of experiments compares the performance of the cutting plane algorithm with (CPA-ap) and without a priori set of cuts (CPA-∅) in terms of the number of iterations (#ITR) and the computational times (CPU(s)). We generate a priori set of cuts to approximate the function $\frac{R}{1+R}$ at 32 points as R^h = [0, 0.0326554, 0.102376, 0.179404, 0.264797, 0.359813, 0.465954, 0.585027, 0.719222, 0.871213, 1.04429, 1.24255, 1.47111, 1.7365, 2.04706, 2.41367, 2.85069, 3.37736, 4.02001,4.8154, 5.81609, 7.09939, 8.78276, 11.0518, 14.2139, 18.8083, 25.854, 37.4721, 58.7112, 104.244, 233.952, 988.484]. This provides an initial approximation within 0.001 to the function $\frac{R}{1+R}$ (See Elhedhli (2005) for further information).

Table 2 demonstrates the effect of adding a priori set of cuts at the start of the cutting plane algorithm on the computational times and the number of iterations. The results show that for the CPA-∅ the CPU times (per second) are on average 998 (M/D/1 case), 1069 (M/M/1 case), and 1077 (M/G/1 case) and the average number of iterations is 12 in all three cases. With the addition of a priori cuts, the average CPU times reduce significantly to 134, 131, 161 second and the average number of iterations reduces to 0.5, 0.7, and 0.9 for M/D/1, M/M/1, and M/G/1 cases respectively. Furthermore, our results show that the effect of adding a priori set of cuts on the computational time of CPA is more significant as the congestion cost factor and the

coefficient of service time increases. This is expected as larger values of c and θ inflate the approximation error and requires additional cuts.

It is worthwhile to mention that in some instances of M/G/1 the CPU times are lower than their corresponding M/M/1 and/or M/D/1 cases due to the quality of LP relaxation bound obtained at root node of the branch and bound algorithm. Overall, the proposed algorithm along with a set of a priori cuts (CPA-ap) proved to be an efficient method in solving the model. Therefore, we use the algorithm with a set of a priori cuts (CPA-ap) in all the other set of experiments reported in this paper.

5.3 Performance of the Exact Approach and the Genetic Algorithm

Table 3, 4, and 5 report the computational performance of the two solution approaches, the CTA and the GA, on CAB dataset under different values of coefficient of variance of service times: c = 0, 1, and 2. We report the computational performance of the GA on larger instances from TR dataset in Table 6. It is worth noting that results on the performance of the exact approach (i.e., CTA) on TR dataset was not available as computational times exceeded the time limit of 25,000 seconds (6.94 hours). These tables show the total cost (OBJ), the transportation cost (TC), the fixed cost (FC), the congestion cost (CC), the number of iterations (#ITR), and the CPU times in seconds (CPU). The results of the GA are reported as the upper bound (UB) and percentage gap are calculated as $\%Gap = \frac{UB-OBJ}{OBJ} \times 100$.

For all instances derived from the CAB dataset, the exact approach provides optimal solutions (with optimality gap of 10^{-6}) within an average CPU time of 1176, 1351, and 2075 seconds for M/D/1, M/M/1, and M/G/1 cases. The maximum CPU times for M/D/1, M/M/1, and M/G/1 cases are 9449, 10569, and 20367 second while the maximum number of iterations are 2, 2 and 4 respectively. The number of iterations of the exact method implies that only a fraction of constraints (16) of $P_L(H)$ is used which confirms stability and efficiency of the algorithm in finding optimal solutions. As expected, with increase in the number of hubs to be opened, the problem requires more computational effort. The CPU time for the exact approach also increases as the inter-hub discount factor takes larger values. Finally, our results confirm that with increase in the value of the congestion cost factor θ, the congestion cost function dominates and consequently the exact method requires excessive computational time to solve a problem to optimality.

For the CAB dataset, GA provides quality solutions in very short computing times (<10 second). The average percentage gaps of the solutions provided by the algorithm are 3.8,

3.6, and 3.4 % in M/D/1, M/M/1, and M/G/1 cases respectively. The genetic based heuristic approach finds optimal solutions for 14 (M/D/1 case), 11 (M/M/1 case), and 13 (M/G/1 case) instances. For the TR dataset, GA provides feasible solutions to the problems with up to 81 nodes within an average computing time of 40 second. Note that unlike the exact approach, increasing inter-hub discount factor α and/or the congestion cost factor θ do no significantly impact the computational performance of the GA. our computational result confirms the stability and the efficiency of the GA in finding near-optimal solutions to the problem within reasonable optimality gap.

Table 2 The effect of adding a priori set of cuts in cutting plane algorithm on computation time

				M/D/1(c=0)					M/M/1(c=1)					M/G/1(c=2)				
					CPA-∅		CPA-ap			CPA-∅		CPA-ap			CPA-∅		CPA-ap	
n	p	α	θ	OBJ	#ITR	CPU(s)	#ITR	CPU(s)	OBJ	#ITR	CPU(s)	#ITR	CPU(s)	OBJ	#ITR	CPU(s)	#ITR	CPU(s)
10	3	0.2	1	1159.5	9	23	0	3	1170.6	9	26	0	3	1204.1	11	41	1	10
			20	1317.3	8	55	0	4	1375.5	7	53	1	10	1505.3	10	60	1	10
			50	1455.5	8	72	1	10	1563.7	10	70	1	11	1860.7	13	106	1	14
		0.4	1	1236.0	11	40	0	6	1247.2	11	37	0	3	1280.6	11	46	1	11
			20	1392.5	7	51	0	4	1450.8	7	58	1	10	1580.6	9	60	1	11
			50	1530.8	8	79	1	7	1639.0	10	71	1	11	1942.1	12	92	1	18
		0.8	1	1389.1	14	106	0	9	1400.3	10	80	0	8	1433.8	12	98	0	13
			20	1534.9	10	88	0	6	1596.8	9	73	1	12	1726.6	7	65	1	12
			50	1676.8	8	78	1	11	1785.0	8	79	1	10	2104.0	15	207	1	22
	4	0.2	1	1257.9	12	36	0	6	1263.5	11	29	0	4	1280.3	9	26	1	6
			20	1407.6	11	37	1	6	1505.4	15	78	1	17	1635.3	11	62	1	10
			50	1606.7	13	72	1	14	1704.2	11	56	1	7	1938.8	9	30	1	7
		0.4	1	1333.2	15	45	0	8	1338.8	12	31	0	3	1355.5	10	25	1	6
			20	1482.9	10	31	1	6	1590.2	13	80	1	18	1731.4	10	52	1	16
			50	1699.0	15	105	2	37	1800.3	10	59	0	7	2034.9	9	40	1	7
		0.8	1	1472.0	19	72	1	14	1484.8	16	62	0	13	1501.5	12	50	0	10
			20	1628.9	10	56	1	6	1740.5	10	65	1	26	1905.6	12	133	1	22
			50	1860.8	12	103	1	31	1974.5	10	104	1	25	2208.6	8	55	1	12
15	3	0.2	1	1503.3	13	627	0	100	1522.7	14	968	1	173	1539.3	10	818	0	80
			20	1642.2	13	1676	0	67	1702.0	15	1627	0	88	1867.6	17	2667	1	673
			50	1795.0	15	1604	1	227	1923.3	16	2304	1	462	2151.8	19	3331	1	192
		0.4	1	1625.8	7	528	0	72	1634.7	7	505	1	170	1653.0	7	822	1	144
			20	1756.4	13	1953	1	198	1830.1	16	2635	1	381	1950.4	15	3181	1	377
			50	1923.1	17	2883	1	947	2006.1	15	2728	1	256	2241.5	18	2307	1	212
		0.8	1	1749.3	12	400	0	62	1761.9	12	612	1	112	1799.8	12	1410	1	160
			20	1939.5	16	4660	0	556	1985.4	13	4130	0	201	2076.5	12	2497	0	107
			50	2056.3	13	3214	0	148	2132.1	11	3251	0	907	2359.7	14	1587	1	211
	4	0.2	1	1475.5	7	283	0	30	1488.7	10	401	1	67	1528.3	13	809	1	143
			20	1639.5	14	1675	0	109	1696.9	16	1892	1	325	1799.1	14	2204	2	546
			50	1780.1	15	2257	2	561	1860.7	15	2272	1	280	2025.8	16	1866	2	441
		0.4	1	1621.9	12	827	0	62	1634.7	10	550	1	120	1653.0	10	1199	1	119
			20	1756.4	12	2218	1	164	1826.4	19	3703	0	198	1928.1	13	2798	2	730
			50	1909.6	15	2356	0	406	1993.1	16	3908	1	389	2160.0	15	1725	1	753
		0.8	1	1749.3	12	464	0	60	1761.9	12	439	1	114	1799.8	12	1438	1	157
			20	1939.5	16	4054	0	564	1985.4	12	3491	1	903	2076.5	11	3830	0	86
			50	2056.3	13	3091	1	289	2132.1	10	1635	1	171	2359.7	16	2847	0	452
	Min				7	23	0	3		7	26	0	33		7	25	0	6
	Ave				12	998	0.5	134		12	1069	0.7	131		12	1077	0.9	161
	Max				19	4660	2	947		19	4130	1	903		19	3830	2	753

Table 3 Performance of the Exact Approach and Genetic Algorithm on CAB Dataset: M/D/1 Case (c = 0)

				Cutting Plane Algorithm						Genetic Algorithm		%Gap
n	p	α	θ	TC	FC	CC	OBJ	#ITR	CPU(s)	UB	CPU(s)	
10	3	0.2	1	495.7	650	13.8	1159.5	0	2.7	1159.4	0.6	-
			20	500.0	700	117.2	1317.3	0	4.0	1317.3	0.4	-
			50	500.0	750	205.5	1455.5	1	9.7	1455.5	0.3	-
		0.4	1	572.2	650	13.8	1236.0	0	6.3	1236.0	0.2	-
			20	575.3	700	117.2	1392.5	0	4.3	1392.5	0.2	-
			50	575.3	750	205.5	1530.8	1	7.4	1530.8	0.5	-
		0.8	1	725.4	650	13.8	1389.1	0	9.2	1389.1	0.2	-
			20	717.0	700	120.2	1537.1	0	5.5	1537.1	0.2	-
			50	721.3	750	205.5	1676.8	1	10.8	1676.8	0.4	-
	4	0.2	1	500.0	750	7.9	1257.9	0	6.2	1319.7	0.3	4.9
			20	500.0	750	157.6	1407.6	1	6.2	1464.9	0.5	4.1
			50	400.6	950	256.1	1606.7	1	14.2	1618.9	0.4	0.8
		0.4	1	575.3	750	7.9	1333.2	0	7.6	1423.5	0.2	6.8
			20	575.3	750	157.6	1482.9	1	6.1	1569.2	1.1	5.8
			50	634.8	750	314.2	1699.0	2	36.8	1723.9	0.3	1.5
		0.8	1	743.7	700	28.3	1472.0	1	14.0	1579.9	0.2	7.3
			20	721.3	750	157.6	1628.9	1	6.5	1738.9	0.4	6.8
			50	799.9	750	310.9	1860.8	1	31.0	1903.0	0.7	2.3
15	3	0.2	1	826.3	650	26.9	1503.3	0	99.8	1543.4	1.3	2.7
			20	835.9	700	103.6	1639.5	0	67.3	1639.5	4.7	-
			50	835.9	700	259.1	1795.0	1	226.9	1795.0	0.6	-
		0.4	1	914.9	700	10.9	1625.8	0	71.7	1642.1	3.9	1.0
			20	940.2	700	116.2	1756.4	1	198.0	1792.7	1.7	2.1
			50	964.0	700	259.1	1923.1	1	946.8	1935.3	1.2	0.6
		0.8	1	1234.8	500	14.5	1749.3	0	61.8	1818.1	1.2	3.9
			20	1140.3	700	99.2	1939.5	0	556.0	1939.5	1.2	-
			50	1138.1	750	169.1	2057.2	0	148.2	2056.2	1.7	-
	4	0.2	1	659.3	800	16.2	1475.5	0	30.5	1603.3	7.5	8.7
			20	835.9	700	103.6	1639.5	0	108.7	1749.3	1.7	6.7
			50	684.8	900	195.3	1780.1	2	560.5	1903.0	3.4	6.9
		0.4	1	914.9	700	10.9	1625.8	0	61.7	1763.7	6.9	8.5
			20	940.2	700	116.2	1756.4	1	164.4	1884.0	2.6	7.3
			50	815.7	900	194.4	1910.1	0	406.0	2038.4	2.3	6.7
		0.8	1	1234.8	500	14.5	1749.3	0	60.3	1956.8	2.4	11.9
			20	1140.3	700	99.2	1939.5	0	564.1	2091.2	2.1	7.8
			50	1137.7	750	168.6	2056.3	1	289.3	2271.7	5.5	10.5
20	3	0.2	1	724.5	650	15.6	1390.1	0	524.5	1441.9	2.8	3.7
			20	724.5	700	110.7	1535.2	0	368.3	1535.2	8.9	-
			50	731.3	700	250.9	1682.1	0	314.7	1725.1	8.6	2.6
		0.4	1	847.8	650	15.6	1513.4	0	250.0	1570.1	6.6	3.7
			20	847.8	700	110.7	1658.4	0	442.0	1687.8	4.0	1.8
			50	865.0	700	250.9	1815.9	0	676.8	1836.8	3.6	1.2
		0.8	1	1173.9	500	10.8	1684.7	0	254.5	1770.4	6.4	5.1
			20	1182.5	500	199.7	1882.2	1	2481.5	1895.0	3.6	0.7
			50	1103.0	750	162.9	2015.8	0	1248.6	2027.7	6.4	0.6
	4	0.2	1	731.3	700	14.8	1446.1	1	826.1	1534.7	3.3	6.1
			20	587.8	900	125.8	1613.6	1	2438.1	1659.1	4.7	2.8
			50	589.8	750	221.7	1761.5	0	687.3	1785.2	6.2	1.3
		0.4	1	847.8	700	23.6	1571.4	1	503.5	1702.6	5.9	8.4
			20	733.2	900	125.0	1758.2	0	1097.8	1827.1	6.7	3.9
			50	748.8	950	214.1	1912.9	1	1639.7	1921.5	7.8	0.5
		0.8	1	1091.1	700	14.8	1805.8	0	2458.2	1911.7	5.1	5.9
			20	1103.0	750	100.1	1953.0	0	3184.4	2066.2	6.5	5.8
			50	1103.0	750	250.2	2103.1	1	2731.9	2236.4	4.7	6.3
25	3	0.2	1	785.1	650	12.2	1447.3	0	1232.3	1521.6	5.2	5.1
			20	767.3	700	85.5	1552.8	0	2131.8	1559.9	5.1	0.5
			50	770.6	700	209.3	1679.8	1	3800.2	1723.2	7.6	2.6
		0.4	1	915.26	650	17.2	1582.5	0	2253.8	1611.9	5.3	1.9
			20	903.5	700	85.5	1689.0	0	1042.6	1755.4	8.5	3.9
			50	903.91	700	217.6	1821.5	0	1122.6	1886.5	4.9	3.6
		0.8	1	1319.05	500	8.4	1827.5	0	1446.4	1920.0	3.6	5.1
			20	1168.66	700	85.5	1954.1	0	2172.6	2073.8	8.2	6.1
			50	1166.3	700	217.6	2083.9	0	5517.5	2122.1	7.7	1.8
	4	0.2	1	770.6	700	8.1	1478.6	0	495.1	1559.1	6.4	5.4
			20	770.6	700	161.0	1631.6	1	2848.3	1745.4	8.4	7.0
			50	815.5	750	280.7	1846.3	1	2256.5	1861.0	9.4	0.8
		0.4	1	901.7	700	9.3	1611.0	0	696.9	1794.3	7.8	11.4
			20	903.5	700	168.8	1772.3	1	3667.5	1880.9	8.9	6.1
			50	794.5	950	222.3	1966.8	1	9448.6	2031.8	8.1	3.3
		0.8	1	1165.9	700	9.3	1875.1	0	1016.8	2078.8	5.3	10.9
			20	1168.7	700	168.8	2037.4	1	8253.3	2218.2	5.8	8.9
			50	1250.3	750	213.9	2214.2	0	8325.0	2335.4	7.9	5.5
	Min					7.9	1159.5	0	2.7	1159.4	0.2	0
	Average					122.0	1688.1	0.4	1175.9	1753.8	4.0	3.8
	Max					314.2	2214.2	2.0	9448.6	2335.4	9.4	11.9

Table 4 Performance of the Exact Approach and Genetic Algorithm on CAB Dataset: M/M/1 Case (c = 1)

				Cutting Plane Algorithm						Genetic Algorithm		%Gap
n	p	α	θ	TC	FC	CC	OBJ	#ITR	CPU(s)	UB	CPU(s)	
10	3	0.2	1	495.7	650	24.9	1170.6	0	2.6	1170.6	0.3	-
			20	500.0	700	125.5	1375.5	1	10.4	1375.5	0.3	-
			50	500.0	750	313.7	1563.7	1	10.7	1563.7	2.9	-
		0.4	1	572.2	650	24.9	1247.2	0	3.2	1247.2	0.2	-
			20	575.3	750	125.5	1450.8	1	10.5	1450.8	0.1	--
			50	575.3	750	313.7	1639.0	1	10.5	1639.0	0.8	-
		0.8	1	725.4	650	24.9	1402.0	0	7.5	1402.0	0.2	-
			20	717.0	750	125.5	1596.8	1	12.2	1596.8	0.9	-
			50	721.3	750	313.7	1785.0	1	9.8	1785.0	0.2	-
	4	0.2	1	500.0	750	13.5	1263.5	0	3.6	1326.6	2.9	5.0
			20	400.6	950	154.7	1505.4	1	16.7	1518.5	0.3	0.9
			50	432.9	1000	271.3	1704.2	1	6.8	1711.0	0.3	0.4
		0.4	1	575.3	750	13.5	1338.8	0	3.2	1430.4	0.4	6.8
			20	575.3	750	205.4	1590.2	1	17.5	1623.1	0.2	2.1
			50	634.8	1000	271.3	1800.3	0	7.4	1814.8	0.8	0.8
		0.8	1	529.0	750	13.5	1484.8	0	13.3	1585.9	0.3	6.8
			20	721.3	750	269.2	1740.5	1	26.1	1809.7	0.5	4.0
			50	703.7	1000	270.9	1974.5	1	25.0	1993.9	0.8	1.0
15	3	0.2	1	802.9	700	19.8	1522.7	1	173.5	1527.8	3.5	0.3
			20	835.9	700	166.1	1702.0	0	87.7	1736.3	1.8	2.0
			50	928.2	750	245.0	1923.3	1	462.4	1929.5	1.5	0.3
		0.4	1	914.9	700	19.8	1634.7	1	170.2	1666.2	4.6	1.9
			20	964.0	700	166.1	1830.1	1	380.6	1848.7	1.6	1.0
			50	1011.0	750	245.0	2006.1	1	256.2	2009.4	2.4	0.2
		0.8	1	1234.8	500	27.1	1761.9	1	112.0	1813.8	2.3	2.9
			20	1138.1	750	98.2	1986.3	0	201.1	1985.4	3.4	-
			50	1138.1	750	245.6	2133.6	0	97.3	2133.6	2.7	-
	4	0.2	1	659.3	800	29.4	1488.7	1	67.5	1623.1	5.9	0.9
			20	686.2	900	112.1	1698.2	1	325.5	1802.8	5.5	6.2
			50	693.8	950	216.1	1859.9	1	279.6	1978.2	7.9	6.4
		0.4	1	914.9	700	19.8	1634.7	1	120.3	1764.4	2.8	7.9
			20	815.7	900	111.4	1827.1	0	198.1	1951.7	7.8	6.8
			50	831.9	950	211.2	1993.1	1	388.6	2130.6	5.0	6.9
		0.8	1	1243.7	500	26.6	1770.3	1	114.2	1970.7	3.3	11.3
			20	1137.7	750	97.8	1985.4	1	902.6	2166.4	2.9	9.1
			50	1137.7	750	244.4	2132.1	1	171.2	2321.9	3.1	8.9
20	3	0.2	1	724.5	650	28.6	1403.2	0	284.1	1413.1	5.7	0.7
			20	731.3	700	161.5	1592.8	0	525.2	1654.0	6.0	3.8
			50	802.0	750	232.8	1784.9	2	1375.9	1824.7	3.7	2.2
		0.4	1	847.8	650	28.6	1526.4	0	479.7	1592.2	6.1	4.3
			20	865.0	700	161.5	1726.6	1	1108.5	1744.4	5.3	1.0
			50	908.2	750	245.7	1903.9	0	688.1	1913.2	7.2	0.5
		0.8	1	1173.5	500	19.8	1693.3	0	295.7	1778.5	7.3	5.0
			20	1102.1	700	136.2	1938.2	0	1031.1	1944.3	5.6	0.3
			50	1103.0	750	235.7	2088.6	0	1146.0	2098.3	4.5	0.5
	4	0.2	1	731.3	700	27.2	1458.5	1	1413.6	1571.4	5.3	7.7
			20	589.8	950	130.2	1670.0	0	407.4	1677.6	2.6	0.5
			50	604.5	950	307.7	1862.3	1	2109.9	1874.9	6.5	0.7
		0.4	1	847.8	700	44.9	1592.7	1	626.3	1765.5	7.9	10.9
			20	727.1	950	145.3	1822.4	1	3242.1	1869.0	5.1	2.6
			50	751.9	950	307.7	2009.6	1	1328.7	2018.2	3.9	0.4
		0.8	1	1097.3	700	17.0	1814.3	1	5059.5	1955.0	6.7	7.8
			20	1103.0	750	157.9	2010.8	0	1132.9	2138.4	3.4	6.3
			50	1121.1	750	364.4	2235.1	1	7027.9	2333.7	5.7	4.4
25	3	0.2	1	785.9	650	21.8	1457.7	0	1875.0	1598.5	6.0	9.7
			20	770.6	700	127.7	1598.2	1	3281.4	1615.1	7.5	1.1
			50	770.6	700	319.1	1789.7	1	2331.7	1837.4	8.2	2.7
		0.4	1	922.1	650	21.8	1593.8	0	964.5	1655.4	6.1	3.9
			20	903.5	700	131.8	1735.3	0	1302.5	1792.0	7.6	3.3
			50	913.1	700	319.1	1932.3	1	4311.1	1985.6	8.9	2.8
		0.8	1	1319.1	500	15.1	1834.2	0	1163.6	1949.7	5.9	6.3
			20	1168.7	700	131.8	2000.5	0	2778.9	2096.9	5.8	4.8
			50	1171.4	700	329.5	2200.9	0	9113.2	2257.3	6.9	2.6
	4	0.2	1	770.6	700	13.7	1484.2	0	575.6	1625.5	8.5	9.5
			20	776.9	700	265.7	1742.6	1	2361.3	1816.7	8.7	4.3
			50	624.1	1000	250.2	1892.3	1	2325.2	1977.4	5.8	4.5
		0.4	1	901.7	700	16.2	1617.9	0	778.9	1772.5	9.5	9.6
			20	791.6	900	169.1	1860.7	1	6918.0	1934.8	6.6	4.0
			50	803.0	1000	250.2	2053.3	1	4740.0	2129.9	6.9	3.7
		0.8	1	1165.9	700	16.1	1882.0	0	1121.3	2083.2	9.8	10.7
			20	1251.6	750	130.1	2131.7	0	5801.4	2220.2	9.0	4.2
			50	1252.5	750	325.9	2328.4	1	10569.1	2450.0	8.8	5.2
	Min					13.5	1170.6	0	2.6	1170.6	0.1	0
	Average					148.4	1767.1	0.6	1351.5	1810.7	4.4	3.6
	Max					364.4	3333.0	2.0	10569.1	2450.0	9.8	11.3

Table 5 Performance of the Exact Approach and Genetic Algorithm on CAB Dataset: M/G/1 Case (c = 2)

				Cutting Plane Algorithm						Genetic Algorithm		%Gap
n	p	α	*θ*	TC	FC	CC	OBJ	#ITR	CPU(s)	UB	CPU(s)	
10	3	0.2	1	495.7	650	58.4	1204.1	1	9.5	1204.1	0.4	-
			20	500.0	750	255.3	1505.3	1	9.7	1505.3	0.7	-
			50	553.4	750	557.3	1860.7	1	14.1	1860.7	0.5	-
		0.4	1	572.2	650	58.4	1280.6	1	11.1	1280.6	0.3	-
			20	575.3	750	255.3	1580.6	1	10.6	1580.6	0.3	--
			50	634.8	750	557.3	1942.1	1	18.3	1942.1	0.9	-
		0.8	1	725.4	650	58.4	1433.8	0	12.7	1433.7	0.3	-
			20	721.3	750	255.3	1726.6	1	11.7	1726.6	1.6	-
			50	797.6	750	557.3	2104.9	11	21.8	2104.0	0.5	-
	4	0.2	1	500.0	750	30.2	1280.3	1	5.9	1347.4	0.6	5.2
			20	432.9	1000	202.4	1635.3	1	10.1	1642.1	0.5	0.4
			50	432.9	1000	505.9	1938.8	1	6.9	1971.5	0.3	1.7
		0.4	1	575.3	750	30.2	1355.5	1	6.2	1451.2	0.3	7.1
			20	529.0	1000	202.4	1731.4	1	15.9	1745.8	0.5	0.8
			50	529.0	1000	505.9	2034.9	1	6.8	2067.8	0.8	1.6
		0.8	1	721.3	750	30.2	1501.5	0	9.8	1602.3	0.2	6.7
			20	703.7	1000	202.0	1905.6	1	22.1	1925.0	0.7	1.0
			50	703.6	1000	505.0	2208.6	1	12.5	2246.1	2.6	1.7
15	3	0.2	1	815.7	700	24.8	1540.5	0	80.5	1578.6	1.3	2.5
			20	926.1	750	194.9	1871.0	1	673.4	1873.9	1.5	0.2
			50	928.2	750	473.5	2151.8	1	191.7	2182.2	4.4	1.4
		0.4	1	927.4	700	25.6	1653.0	1	144.5	1693.0	2.1	2.4
			20	1011.0	750	189.4	1950.4	1	376.8	1950.4	4.7	-
			50	1011.0	750	473.5	2234.6	1	212.3	2234.6	7.1	-
		0.8	1	1234.8	500	65.0	1799.8	1	159.7	1837.8	2.6	2.1
			20	1138.1	750	189.9	2078.0	0	107.0	2078.0	4.2	-
			50	1137.7	750	472.0	2359.7	1	211.3	2359.7	6.6	-
	4	0.2	1	659.3	800	69.0	1528.3	1	143.2	1638.1	2.7	7.2
			20	684.8	900	214.1	1798.9	2	546.2	1912.7	3.4	6.3
			50	711.1	1000	314.7	2025.8	2	440.7	2226.6	4.5	9.9
		0.4	1	927.4	700	25.6	1653.0	1	118.9	1774.4	1.9	7.3
			20	815.7	900	212.4	1928.1	2	729.5	2062.5	2.4	7.0
			50	831.9	950	378.1	2160.0	1	753.1	2397.6	5.7	11.0
		0.8	1	1234.8	500	65.0	1799.8	1	157.1	1982.9	5.8	10.2
			20	1138.1	750	189.9	2078.0	0	85.7	2269.4	6.4	9.2
			50	1137.7	750	472.0	2359.7	1	452.2	2586.8	7.7	9.6
20	3	0.2	1	759.0	650	29.9	1438.9	1	1143.4	1448.5	5.1	0.7
			20	802.0	750	178.8	1730.9	2	1383.7	1745.7	5.3	0.9
			50	812.0	750	428.4	1990.4	1	1502.7	2016.9	8.6	1.3
		0.4	1	877.3	650	29.9	1557.2	0	561.6	1610.6	3.5	3.4
			20	908.2	750	191.6	1849.9	0	872.3	1889.3	7.6	2.1
			50	939.7	750	429.2	2118.9	3	3410.2	2166.5	5.6	2.2
		0.8	1	1173.5	500	46.9	1720.3	1	630.6	1798.9	2.4	4.6
			20	1103.8	750	186.1	2040.0	0	1348.7	2050.6	3.4	0.5
			50	1125.3	750	429.3	2304.6	4	6862.1	2314.5	9.6	0.4
	4	0.2	1	731.3	700	64.4	1495.7	1	1850.4	1571.3	2.9	5.1
			20	589.8	950	254.8	1794.6	1	922.1	1811.8	7.7	1.0
			50	626.1	1000	498.8	2124.9	1	3400.3	2177.8	6.5	2.5
		0.4	1	865.0	700	64.4	1629.5	1	1457.3	1720.7	7.1	5.6
			20	751.9	950	238.7	1940.6	1	1561.9	2014.4	8.2	3.8
			50	843.2	1000	405.8	2249.0	1	2156.7	2295.1	8.6	2.1
		0.8	1	1097.3	700	38.7	1836.0	1	4218.9	1997.5	5.4	8.8
			20	1121.1	750	201.0	2171.7	1	4955.5	2228.8	5.3	2.6
			50	1044.7	1000	405.3	2449.9	1	2952.9	2501.9	4.4	2.1
25	3	0.2	1	769.4	700	13.9	1483.3	0	3116.3	1548.0	9.9	4.4
			20	770.6	700	259.5	1730.1	1	4750.5	1791.8	6.6	3.6
			50	841.9	750	468.3	2060.2	1	3265.2	2089.7	8.6	1.4
		0.4	1	901.7	700	13.9	1615.6	0	1913.7	1663.4	5.9	3.0
			20	913.1	700	259.5	1872.6	1	4782.5	1948.8	7.3	4.1
			50	1018.6	750	422.8	2191.3	1	7473.5	2271.3	8.7	3.7
		0.8	1	1324.2	500	33.2	1857.4	0	1399.5	1959.5	8.9	5.5
			20	1168.7	700	270.8	2139.4	0	4557.5	2186.8	5.3	2.2
			50	1250.3	750	397.6	2397.9	2	20367.1	2459.3	6.8	2.6
	4	0.2	1	770.6	700	30.6	1501.1	0	518.7	1644.8	5.2	9.6
			20	904.9	750	289.0	1943.9	1	2743.5	1943.4	6.6	-
			50	689.2	1000	398.5	2087.7	1	4227.9	2214.4	8.2	6.1
		0.4	1	903.5	700	32.7	1636.2	1	1694.1	1825.2	5.5	11.5
			20	803.7	1000	188.8	1999.5	1	7302.5	2089.1	7.9	4.5
			50	855.8	1000	397.4	2253.1	2	10917.5	2322.9	6.2	3.1
		0.8	1	1165.9	700	36.8	1902.7	1	2067.5	2110.8	8.6	10.9
			20	1250.3	750	263.6	2263.9	1	12496.9	2364.8	5.8	4.5
			50	1150.2	1000	378.1	2528.3	1	10988.4	2570.2	7.5	1.7
	Min					13.9	1204.1	0	5.9	1204.1	0.2	0
	Average					237.4	1873.9	1	2075.3	1939.5	4.5	3.4
	Max					557.3	2528.3	4	20367.1	2586.8	9.9	11.5

Table 6 Performance of Genetic Algorithm on TR Dataset

				M/D/1		M/M/1		M/G/1	
n	*p*	α	*θ*	UB	CPU(s)	UB	CPU(s)	UB	CPU(s)
25	3	0.2	1	1675.97	7.9	1686.17	4.3	1695.25	24.3
			20	1782.89	24.1	1841.73	4.1	1946.26	11.3
			50	1904.20	8.8	2001.56	15.8	2191.48	17.8
		0.4	1	1774.62	23.1	1774.79	6.8	1824.71	5.7
			20	1888.52	26.5	1947.00	13.8	2049.33	23.4
			50	2006.61	10.8	2104.52	22.5	2312.79	21.2
		0.8	1	1860.59	7.0	1877.04	16.2	1896.54	20.5
			20	1986.80	11.4	2030.85	13.9	2148.14	9.5
			50	2110.29	23.2	2202.19	15.9	2393.24	17.4
	4	0.2	1	2050.03	18.2	2090.44	16.0	2052.56	23.8
			20	2193.09	22.2	2245.83	18.6	2385.37	25.5
			50	2349.06	21.4	2421.38	13.4	2615.61	21.8
		0.4	1	2164.98	15.6	2187.45	17.9	2209.42	26.8
			20	2338.56	22.2	2394.53	27.9	2510.95	27.9
			50	2466.68	12.3	2578.56	15.8	2758.00	11.8
		0.8	1	2269.45	26.3	2291.94	21.5	2315.29	21.9
			20	2453.28	14.6	2526.16	26.1	2659.28	11.3
			50	2627.29	11.8	2725.36	8.6	2950.63	6.6
55	3	0.2	1	1820.49	46.4	1829.48	38.1	1845.29	45.1
			20	1942.58	33.7	2017.01	47.2	2140.41	39.1
			50	2064.81	47.8	2126.77	40.1	2314.32	42.7
		0.5	1	1903.73	33.9	1961.79	30.5	1934.52	43.9
			20	2041.66	37.7	2066.18	45.9	2160.76	43.7
			50	2185.07	47.9	2218.77	26.7	2405.62	41.2
		0.8	1	1971.40	21.2	1995.21	47.9	2025.18	41.3
			20	2115.29	45.7	2161.47	45.9	2278.08	43.1
			50	2271.01	23.9	2320.57	32.3	2523.84	41.6
	4	0.2	1	2271.74	30.7	2285.88	42.4	2303.59	33.6
			20	2363.02	33.6	2425.52	43.2	2542.50	47.3
			50	2556.36	43.7	2597.44	30.9	2762.33	37.9
		0.5	1	2368.75	45.9	2371.68	47.1	2433.39	47.2
			20	2572.51	47.4	2563.03	33.9	2687.46	45.3
			50	2649.83	45.3	2685.44	47.5	2940.50	33.9
		0.8	1	2433.18	31.7	2458.68	22.8	2496.33	47.0
			20	2653.47	27.1	2656.11	41.3	2773.78	35.8
			50	2763.63	45.1	2825.42	47.2	3018.56	45.4
81	3	0.2	1	1898.06	41.1	1899.45	45.9	1921.38	41.2
			20	2034.87	65.3	2110.34	51.7	2236.96	67.6
			50	2208.89	61.0	2268.09	65.6	2462.70	58.6
		0.5	1	1973.70	61.5	1983.95	40.8	2019.37	55.9
			20	2151.69	65.9	2186.79	38.4	2336.75	65.4
			50	2295.94	67.7	2359.37	61.7	2526.97	67.8
		0.8	1	2048.42	45.3	2060.32	53.2	2078.53	61.5
			20	2242.46	61.6	2268.94	61.6	2382.08	45.8
			50	2345.55	67.8	2401.46	55.2	2621.96	57.6
	4	0.2	1	2411.48	47.5	2374.28	65.4	2428.58	53.7
			20	2544.80	65.9	2601.51	60.6	2640.86	69.2
			50	2699.15	64.5	2697.16	61.5	2897.84	65.8
		0.5	1	2525.46	61.7	2499.34	30.9	2542.62	61.0
			20	2629.19	67.6	2707.04	65.8	2785.98	67.1
			50	2784.37	65.3	2844.74	65.2	3042.90	61.7
		0.8	1	2613.59	64.5	2662.17	65.9	2681.15	67.5
			20	2764.09	66.9	2778.81	67.2	2895.41	58.5
			50	2865.76	67.6	2948.16	54.8	3142.16	66.4
	Min			1675.97	7.00	1686.17	4.10	1695.25	5.7
	Average			2257.20	38.92	2299.00	37.06	2410.10	40.29
	Max			2865.76	67.80	2948.16	67.20	3142.16	69.20

6 Summary and Conclusion

In this paper, we present a model that captures the trade-off between transportation cost savings induced by the economies of scale and the congestion costs due to the variability of arrival and service rates of the flow at hub facilities. We modelled and analysed the effect of congestion on the design of logistics systems with hub-and-spoke topologies. Hubs are modelled as single server queues with Poisson arrivals and general service time distributions. The congestion is captured using the number of users at hubs. We present two solution approaches: an exact method and an approximation technique. In the first approach we linearize the initial nonlinear model and use a cutting plane algorithm to solve small to medium size problem instances to optimality. As the second solution approach, we propose a genetic algorithm based heuristic to solve large instances of the problem.

In order to mitigate the effects of congestion, the proposed model redistribute the flow across hubs to achieve maximize hub utilization and/or decide suitable hub capacities to achieve higher relative difference of hub flow and hub capacities. Our computational results demonstrate that substantial reduction in congestion can be achieved with relatively small increase in total costs. We further illustrate that network configurations offered by the model that include congestion cost could be very different from those proposed by a traditional model that ignores congestion. Our computational experiments on CAB and TR datasets confirms the efficiency and stability of both cutting plain and GA based heuristic approaches in locating optimal/best solutions to various problem instances. For CAB dataset, the GA provides solutions that are, on average, within 3.4% of the optimality in short computing times (<10 second). For the TR dataset (with up to 81 nodes), GA provides solutions within 40 second on average.

In this research hub facilities are modelled as single-server queues (M/G/1). Nevertheless it would be beneficial, from both academic and practical point of view, to extend this study and model hubs as multiple servers and explore exact and other solution approaches that can handle problems with such complexity. Another promising avenue that can be explored is to extend the queuing-based congestion modelling framework to deal with congestion on links (and link capacity selection) in the hub-and-spoke network. Future research can also explore the possibility of embedding the proposed cutting plane based exact solution procedure within the Lagrangean relaxation/Benders decomposition framework to solve large-scale instances of the hub-and-spoke problems with congestion.

References

Abdinnour-Helm, S. (2001). Using simulated annealing to solve the p-hub median problem. *International Journal of Physical Distribution and Logistics Management*, 31(3), 203–220.

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

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

Alumur, S.A., Nickel, S., Saldanha da Gama, F. (2012). Hub location under uncertainty. *Transportation Research: Part B*, 46, 529–543.

Azizi, N. (2017). Managing Facility Disruption in Hub-and-Spoke Networks: Formulations and Efficient Solution Methods, *Annals of Operations Research*, DOI: 10.1007/s10479-017-2517-0.

Azizi, N., Chauhan, S., Salhi, S., Vidyarthi, N. (2016). The impact of hub failure in hub-and-spoke networks: Mathematical formulations and solution techniques. *Computers & Operations Research*, 65, 74–188.

Boffey, B., Galvao, R., Espejo, L. (2007). A review of congestion models in the location of facilities with immobile servers. *European Journal of Operational Research*, 178(3), 643–662.

Bryan, D.L., O'Kelly, M.E. (1999). Hub-and-spoke networks in air transportation: An analytical review. *Journal of Regional Science*, 39(2), 275–295.

Camargo, R.S.de, Miranda. G. Jr., Ferreira, R.P.M. (2011). A hybrid outer approximation/benders decomposition algorithm for the single allocation hub location problem under congestion. *Operations Research Letters*, 39(12), 329–337.

Camargo, R.S.de, Miranda, G. Jr., Ferreira, R.P.M., Luna, H.P. (2009a). Multiple allocation hub-and-spoke network design under hub congestion. *Computers and Operations Research*, 36(12), 3097–3106.

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

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

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

Carello, G., Della Croce, F., Ghirardi, M., Tadei, R. (2004). Solving the hub location problem in telecommunication network design: A local search approach. *Networks*, 44(2), 94–105.

Cetiner, S., Sepil, C., Sural, H. (2010). Hubbing and routing in postal delivery systems. *Annals of Operations Research*, 181 109–124.

Cheung, R. K., and Muralidharan, B. (1999). Impact of dynamic decision making on hub-and-spoke freight transportation networks. *Annals of Operations Research*, 8749–71.

Contreras, I., Cordeau, J.-F., Laporte, G. (2012). Exact solution of large-scale hub location problems with multiple capacity levels. *Transportation Science*, 46, 439 - 459.

Contreras, I., Diaz, J. A., Fernandez, E. (2011c). Branch-and-price for large-scale capacitated problems with single assignment. *INFORMS Journal on Computing*, 23(1), 41–55.

Contreras I., Díaz J., Fernández E. Lagrangean relaxation for the capacitated hub location problem with single assignment. OR Spectrum (2009) 31(3):483–505.

Correia, I., Nickel, S., Saldanha-da-Gama, F. (2010). Single-assignment hub location problems with multiple capacity levels. *Transportation Research: Part B*, 44, 1047–1066.

Cunha, C.B., Silva, M.R. (2007). A genetic algorithm for the problem of configuring a hub-and-spoke network for a LTL trucking company in Brazil. *European Journal of Operational Research*, 179, 747–758.

Ebery, J. 2001. Solving large single allocation p-hub location problems with two or three

hubs. *European Journal of Operational Research*, 128 447–458.
Elhedhli, S. (2005). Exact solution of a class of nonlinear knapsack problems. *Operations Research Letters*, 33, 615–624.
Elhedhli, S., Hu, F. X. (2005). Hub-and-spoke network design with congestion. *Computers and Operations Research*, 32, 1615–1632.
Elhedhli, S., Wu, H. (2010). A Lagrangean heuristic for hub-and-spoke system design with capacity selection and congestion. *INFORMS Journal of Computing*, 22(2), 282–296.
Ernst, A.T., Krishnamoorthy, M. (1996). Efficient algorithms for the uncapacitated single allocation p-hub median problem. *Location Science*, 4(3), 139–154.
Ernst, A.T., Krishnamoorthy, M. (1998b). Exact and heuristic algorithms for the uncapacitated multiple allocation p-hub median problem. *European Journal of Operational Research*, 104, 100–112.
Ernst, A.T., Krishnamoorthy, M. (1999). Solution algorithms for the capacitated single allocation hub location problem. *Annals of Operations Research*, 86, 141–159.
Grove, P. G., O'Kelly, M.E. (1986). Hub networks and simulated schedule delay. *Papers of the Regional Science Association*, 59, 103–119.
Guldmann, J.M., Shen, G. (1997). A general mixed integer nonlinear optimization model for hub network design. Working paper, Department of City and Regional Planning, The Ohio State University, Columbus, Ohio.
Ilic, A., Urosevic, D., Brimberg, J., Mladenovic, N. (2010). A general variable neighbourhood search for solving the uncapacitated single allocation p-hub median problem. *European Journal of Operational Research*, 206, 289–300.
Jeong, S-J., Lee, C.G., Bookbinder, J.H. (2007). The European freight railway system as a hub-and-spoke network. *Transportation Research Part A*, 41, 523–536.
Klincewicz, J.G. (1992). Avoiding local optima in the p-hub location problem using tabu search and GRASP. *Annals of Operations Research*, 40, 283–302.
Klincewicz, J.G. (1998). Hub location in backbone/tributary network design: A review. *Location Science*, 6, 307–335.
Koksalan, M., Soylu, B. (2010). Bicriteria p-hub location problems and evolutionary algorithms. *INFORMS Journal of Computing*, 22(4), 528–542.
Kratica, J., Stanimirovic, Z., Tosic, D., Filipovic, V. (2007). Two genetic algorithms for solving the uncapacitated single allocation p-hub median problem. *European Journal of Operational Research*, 182(1), 15–28.
Lapierre, S.D., Ruiz, A.B., Soriano, P. (2004). Designing distribution networks: Formulations and solution heuristic. *Transportation Science*, 38, 174–187.
Marianov, V., Serra, D. (2003). Location models for airline hubs behaving as M/D/c queues. *Computer and Operations Research*, 30, 983–1003.
Martin, J. C., Roman, C. (2004). Analyzing competition for hub location in intercontinental aviation markets. *Transportation Research Part E*, 40, 135–150.
Mayer, C., Sinai, T. (2003). Network effects, congestion externalities, and air traffic delays: Or why not all delays are evil. *American Economic Review*, 93(4), 1194–1215.
Nickel, S., Schobel, A., Sonneborn, T. (2001). Hub location problems in urban traffic networks. In J. Niittymaki, M. Pursula, eds., Mathematical Methods and Optimization in Transportation Systems. Kluwer Academic Publisher, Dordrecht, The Netherlands, 95–107.
O'Kelly, M.E. (1986a). The location of interacting hub facilities. *Transportation Science*, 20, 92–106.
O'Kelly, M.E. (1987). A quadratic integer program for the location of interacting hub facilities. *European Journal of Operational Research*, 32, 393–404.
O'Kelly, M.E., Skorin-Kapov, D., Skorin-Kapov, S. (1995). Lower bounds for the hub location problem. *Management Science*, 41(4), 713–721.

Salhi, S. (2017). Heuristic Search: The Emerging Science of Problem Solving. Germany: Springer.
Skorin-Kapov, D., Skorin-Kapov, J. (1994). On tabu search for the location of interacting hub facilities. *European Journal of Operational Research*, 73(3), 501–508.
Skorin-Kapov, D., Skorin-Kapov, J., O'Kelly, M.E. (1996). Tight linear programming relaxations of uncapacitated p-hub median problems. *European Journal of Operational Research*, 94, 582–593.
Smith, K., Krishnamoorthy, M., Palaniswami, M. (1996). Neural versus traditional approaches to the location of interacting hub facilities. *Location Science*, 4(3), 155–171.
Vidyarthi, N.K., Elhedhli, S., Jewkes, E. M. (2009). Response time reduction in make-to-order and assemble-to-order supply chain design. *IIE Transactions*, 41(5), 448–466.
Yaman, H., Carello, G. (2005). Solving the hub location problem with modular link capacities. *Computers and Operations Research*, 32, 3227–3245.
Yaman, H., Kara, B.Y., Tansel, B.C. (2007). The latest arrival hub location problem for cargo delivery systems with stopovers. *Transportation Research B*, 41(8), 906–919.