

A multi-objective imperialist competitive algorithm for a capacitated hub covering location problem

M. Mohammadi^a, R. Tavakkoli-Moghaddam^{a*}, H. Rostami^b

^aDepartment of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran

^bDepartment of Industrial Engineering, College of Engineering, University of Tabriz, Tabriz, Iran

ARTICLE INFO

Article history:

Received 20 October 2010

Received in revised form

20 December 2010

Accepted 28 December 2010

Available online

28 December 2010

Keywords:

Hub covering location

Multi-objective problem

Capacitated single allocation

Service time

Imperialist competitive algorithm

ABSTRACT

The hub location problem appears in a variety of applications, including airline systems, cargo delivery systems and telecommunication network design. Hub location problems deal with finding the location of hub facilities and the allocation of demand nodes to these located hub facilities. In this paper, a new model for the capacitated single allocation hub covering location problem is presented. Instead of using capacity constraints to limit the amount of flow received by the hubs, the second objective function is introduced to minimize service times in the hubs. The service time in the hubs includes the waiting time of received flows in a queue and the time to get services. Due to the NP-hardness of the problem, a new weight-based multi-objective imperialist competitive algorithm (MOICA) is designed to find near-optimal solutions. To validate the performance of the proposed algorithm, the solutions obtained by the MOICA are compared by the exact solutions of the mathematical programming model.

© 2011 Growing Science Ltd. All rights reserved

1. Introduction

Hub-and-spoke networks are common in many areas of everyday life from passenger travel through an airline's network of airports, to postal delivery, communication, cargo and public transportation networks. Therefore, the application of this problem appears whenever it is impossible or too expensive to establish a direct link between each pair of origins and destinations. In this situation, some nodes are considered as a hub and remaining nodes are allocated to the located hubs. The problem defines the basic setting of a hub location network problem in order to decide which nodes should become hubs and how the flow in the network should be consolidated and redistributed.

In particular, hub networks use a set of hub nodes to consolidate and reroute the flows, and a reduced number of links, where economies of scale are applied, to connect the large set of origins/destination points. In the literature, four major types of hub location problems exist: (1) capacitated and

* Corresponding author. Tel:(+9821) 82084183, Fax:(+9821) 88013102
E-mail: tavakoli@ut.ac.ir (R. Tavakkoli-Moghaddam)

uncapacitated hub location problem (UHLP), (2) p -hub median problem, (3) p -hub center problem, and (4) hub covering location problem.

In a hub location problem (HLP), the objective is to minimize the total cost of locating hubs and transporting cargo flows through the hub network. In a p -hub median problem (p HMP), the objective is to locate p hubs in the network so that the total cost of transporting flows through the network is minimized. Unlike the UHLP the number of hub is given as input. In a p -hub center problem (p HCP), the objective is to find the optimal location of p hubs and the allocation of non-hub nodes to the hubs and minimize the longest path in the network. Finally, in a hub covering location problem (HCLP), the number of hubs is not given and the objective is to find the best location of hubs in the network and allocation of nodes to hubs such that the total cost of locating the hubs is minimized. The HCLP contains cover constraints, which limit the number of non-hub nodes that can allocate to each hub.

Three coverage criteria for hubs were defined by Campbell (1994). The origin destination pair (i, j) is covered by hubs k and m if:

- the cost from i to j via k and m does not exceed a specified value,
- the cost for each link in the path from i to j via k and m does not exceed a specified value, and
- each of the origin-hub and hub-destination links meets separate specified values.

All of the above types are divided in two major parts: namely single allocation and multiple allocations hub location problems. In single allocation hub networks, each non-hub node is allocated to exactly one hub (Ernst et al., 1996a; Ernst et al., 1996b; O'Kelly, 1987). In multiple allocation networks (Ernst, 1998), a non-hub node can be allocated to more than one hub. In this study, it is considered that hubs network is a complete graph and that the non-hub nodes can be linked directly to, at least, one hub (Aykin, 1988, Campbell, 1996, Campbell, 2002). Nevertheless, some other studies can be found in the literature that consider the incomplete network for a hub location problem (Alumur et al., 2009; Campbell, 2005; Contreras et al., 2009a; Contreras et al., 2010; Nickel et al., 2001; Yaman, 2008).

Another common assumption in the literature is that there are no direct links between each pair of non-hub nodes. Therefore, a traffic should be routed at least one hub. Hub location problems can be classified as a capacitated or uncapacitated problem depending on whether there are limits on flows transferred through the network. These limitations may be applied on links or hub nodes (Campbell, 2002). Capacities on hub nodes can limit the volume of flow into the hub (Ebery et al., 2000; Ernst et al., 1999; Aykin, 1994), or for the total flow through the hub.

The single-allocation hub location problem, in which hubs are capacitated, is known in the literature as the capacitated single-allocation hub location problem (CSAHL). A number of studies exist in the literature considering this type of problem. Campbell (1994) presented the first mixed-integer linear programming (MILP) formulation for the CSAHL. There is a limit on the entire flow entering into the hubs. The costs include establish costs for opening the hubs and flow transporting costs namely consolidation, distribution and transfer costs.

The first formulation for the multiple allocation case was given by Campbell (1994). The rest of the literature on hub location problems were primarily focused on the linearization of the quadratic model proposed in O'Kelly (1987); for example, Campbell (1996), Ernst and Krishnamoorthy (1996b), O'Kelly et al. (1995), and Skorin-Kapov et al. (1996). These studies introduced different mathematical formulations and solution procedures for the minimization of the total transportation cost. The first computational results for single allocation hub covering problem were presented by Kara et al. (2003) and Ernst et al. (2005) who proposed a better mathematical formulation for the hub covering problem using the "radius" idea. For the single allocation uncapacitated hub location

problem, Labbe' and Yaman (2004) derived a family of valid inequalities that generalizes the facet-defining inequalities and can be separated in a polynomial time. The capacitated multiple allocation case was studied by Aykin (1994), Ebery et al. (2000), Boland et al. (2004), and Marin (2005).

Costa et al. (2008) presented a bi-objective approach where the model proposed by Ernst and Krishnamoorthy (1999) is enlarged with the inclusion of a second objective function to be minimized and quantifies the time to process the flow entering the hubs. Contreras et al. (2009b) proposed a similar formulation for the same problem studied by Ernst and Krishnamoorthy (1999). In addition, Jabal-Ameli et al. (2011) presented a location-routing problem that can be used for new contribution in this area.

This paper presents the capacitated single allocation hub covering location problem (CSAHCLP) and it is organized as follows. Section 2 describes the basic definition of multi-objective optimization problems. Section 3 elucidates two multi-objective models for the hub covering location problem. Section 4 explains the weight-based imperialist competitive algorithm in order to find an optimal or near optimal solution. Section 5 presents and analyzes the computational results, and finally some conclusions are drawn in Section 6.

2. Multi-objective optimization problem

The main difference between single and multi-objective optimization problems is the number of the obtained optimal solutions. In a single-objective optimization algorithm, the decision maker (DM) is looking for one and only one optimal solution, while in multi-objective optimization problems, a set of solutions depending on non-dominance criterion are found that is named the Pareto sense. In the following section, we provide a summary of some basic definitions to better understand the multi-objective optimization problem. A multi-objective linear problem (MOLP) can be described as follows.

$$\begin{aligned} \min_x & [f_1(x), f_2(x), \dots, f_k(x)]^T \\ \text{subject to} & \\ & x \in S, \end{aligned}$$

where k is the number of objectives, $f_i(x)$ is the i th objective function ($i = 1, 2, \dots, k$) and S is the feasible region.

Definition 1. Let $z^1 = (f_1(x^1), f_2(x^1), \dots, f_k(x^1))$ and $z^2 = (f_1(x^2), f_2(x^2), \dots, f_k(x^2)) \in R^k$ be two objective vectors. Then, z^1 dominates z^2 if and only if $z^1 \leq z^2$ and $z^1 \neq z^2$ (i.e., $f_i(x^1) \leq f_i(x^2)$ for all i and $f_i(x^1) < f_i(x^2)$ for at least one i).

Definition 2. A feasible solution x^e to the MOLP is called to be locally optimal in the Pareto sense if there exists a real $\varepsilon > 0$ such that there is no solution x^l that dominates the solution x^e with $x^l \in R^k \cap B(x^e, \varepsilon)$, where $B(x^e, \varepsilon)$ shows a bowl shape with a center of x^e and of radius ε .

Definition 3. A solution x^e is globally optimal in the Pareto sense if there is no vector x^l such that x^l dominates the vector x^e .

The main difference between definition 3 and the definition of local optimality lies in the fact that we do not have a restriction on the set R^k anymore.

3. Multi-objective mathematical formulation

Multi-objective models are especially ample to be considered in hub location problems, namely because of the usual conflict between the quality and the cost of the solutions. In this paper, we introduce a new multi-objective model to deal with the limitations of a traditional capacitated hub

location problem. As a matter of fact, if a single-objective model with capacity constraints is chosen, the number of options of the decision maker (DM) is limited to the acceptance or rejection of the optimal solutions. The use of proposed model allows the DM to choose the favorite solution among the optimal solutions obtained by the model with considering different combination of weights for the objective functions. In all the studies mentioned in Section 1, the objective functions are mostly about minimizing the total transportation cost and total shipment time in the hub network through connection links between each pair of nodes. However, none of those papers has considered the total waiting time in hubs. The total waiting time is equal to the time when entered flows to each hub are obliged to wait in a queue to received services. This time includes both the waiting time in a queue and the service time in each hub for preparing flows to transfer them to destinations.

In addition to transportation cost, it considers alternatively the minimization of the time, which flows spend in the hubs, to receive services or the minimization of the maximum spent time in the hubs. Therefore, minimizing the waiting time is considered as the second objective function. Furthermore, the classical capacitated hub covering location problems usually generates solutions determined by an excessive concentration on the flows entering to the hubs. Also, this fact leads to more CPU time for solving the problems. To avoid this situation, the traditional model is modified by using the second objective function instead of the capacity constraint; however, this new object added to the model is not free of inconveniences. Before formulating the problem, the following notations are introduced.

$N = \{1, \dots, n\}$	Set of nodes.
W_{ij}	Flow to be sent from node i to node j ($i, j \in N$).
C_{ij}	Transportation cost of a unit of the flow between i and j
F_k	Fixed cost of opening a hub at node k
Γ_k	Capacity of collecting flow at hub k (flow into hub k).
r_k	Maximum collection/distribution cost between hub k and nodes that are allocated to hub k (radius of hub k).
P_k	Preparation service time of hub k .
T_k	Total spent time in hub k .
$O_i = \sum_{j \in N} W_{ij}$	Total flow originating at node i .
$D_i = \sum_{j \in N} W_{ji}$	Total flow destined for node i .
X_{ik}	is 1 if node i is allocated to hub at node k ; otherwise, it is 0.
Y_{kl}^i	Total amount of flow from location i (origin) that is routed via hubs k and l
$\alpha \in [0,1]$	Cost discount factor for between two hubs.
$\delta \in [0,1]$	Cost discount factor for between non-hub nodes and hubs.
M	A large value

Now, the waiting time in each hub is calculated as follows. According to Fig. 1, circles are entered flows to a hub illustrated by a square. The first arrived unit flow does not wait in the queue and is transferred after P_k time unit. But, the other flows should wait until the prior flows receive services. For example, the second unit flow should wait as long as $2P_k$ before being transferred. This waiting time includes: (1) time P_k when the first unit receives services in the hub and (2) time P_k for getting services for the second unit flow. So the time, in which the second flow spends in the hub k , is equal to $2 \times P_k$. Similarly, we have $3 \times P_k$, $(n-1) P_k$ and $n \times P_k$ for third, the $(n-1)$ th and n th flow unit, respectively. Therefore the total spent time in hub k is as follows:

$$T_k = P_k + 2P_k + 3P_k + \dots + (n-1)P_k + nP_k. \quad (1)$$

Eq. (1) can be written as follows.

$$T_k = (1 + 2 + 3 + \dots + (n - 1) + n) \times P_k \tag{2}$$

Therefore, Eq. (2) is written as follows.

$$T_k = P_k \times \sum_{i=1}^n n, \tag{3}$$

$$T_k = P_k \times \frac{n \times (n + 1)}{2}. \tag{4}$$

Finally, Eq. (4) can be written as follows,

$$T_k = P_k \times \frac{[\sum_{i=1}^n (O_i + D_i)X_{ik}]^2 + \sum_{i=1}^n (O_i + D_i)X_{ik}}{2}, \tag{5}$$

where $\sum_{i=1}^n (O_i + D_i)X_{ik}$ is the total flows entering to hub k . Therefore, Eq. (5) can be the second objective function.

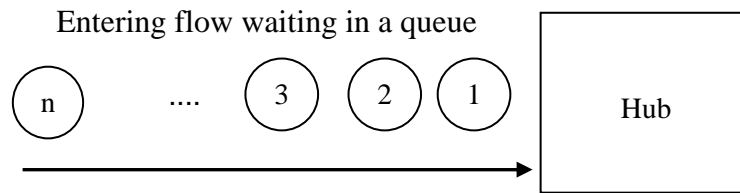


Fig. 1. Entering flows to the hub waiting in a queue

A number of researchers have proposed many formulations for the CSAHLP. Our model is based on the formulation presented by Ernst and Krishnamoorthy (1999), which seems to be the most effective formulation for this kind of problem.

CSAHCLP Model:

$$\min \sum_{i=1}^n \sum_{k=1}^n \delta C_{ij} (O_i + D_i) X_{ik} + \sum_{i=1}^n \sum_{k=1}^n \sum_{l=1}^n \alpha C_{kl} Y_{kl}^i + \sum_{k=1}^n F_k X_{kk} \tag{6}$$

subject to

$$\sum_{k=1}^n X_{ik} = 1 \quad i = 1, \dots, n \tag{7}$$

$$X_{ik} \leq X_{kk} \quad i = 1, \dots, n; k = 1, \dots, n \tag{8}$$

$$C_{ik} X_{ik} \leq r_k \quad i = 1, \dots, n; k = 1, \dots, n \tag{9}$$

$$\sum_{i=1}^n (O_i + D_i) X_{ik} \leq \Gamma_k \quad k = 1, \dots, n \tag{10}$$

$$\sum_{l=1}^n Y_{kl}^i + \sum_{j=1}^n W_{ij} X_{jk} = O_i X_{ik} + \sum_{l=1}^n Y_{lk}^i \quad i = 1, \dots, n; k = 1, \dots, n \tag{11}$$

$$Y_{kl}^i \leq M X_{kk} \quad i = 1, \dots, n; k = 1, \dots, n; l = 1, \dots, n \tag{12}$$

$$Y_{kl}^i \leq M X_{ll} \quad i = 1, \dots, n; k = 1, \dots, n; l = 1, \dots, n \tag{13}$$

$$X_{ik} \in \{0, 1\} \quad i = 1, \dots, n; k = 1, \dots, n \quad (14)$$

$$Y_{kl}^i \geq 0 \quad i = 1, \dots, n; k = 1, \dots, n; l = 1, \dots, n \quad (15)$$

The objective function sums the transportation cost over all (i,j) pairs and the fixed cost of establishing a hub. Eq. (7) together with constraint (14) enforces single allocation for each node. Constraints (8) assure that no node is assigned to a location unless a hub is opened at that site. Constraint (9) makes sure that node i can only be allocated to k , if cost C_{ik} between i and k is at most the radius r_k . Constraint (10) shows the capacity constraint that limits the amount of flows processed by hub k . Eq. (11) is the divergence equations for commodity i at node k in a complete graph, when the demand and supply at the node is determined by the allocations X_{ik} . Constraints (12) and (13) ensure that the value of Y_{kl}^i can be more than zero if nodes k and l are the valid hubs. Constraint (15) is a domain constraint.

In this paper, the main idea is to eliminate the capacity constraint given in Eq. (10) and to introduce the second objective function that measures the flow-time spent in the hubs. Two approaches are considered for the second objective function: (1) summing the total time spent for processing the flow gathered by the hubs that, of course, has to be minimized (SAHCLP-1) and (2) minimizing the maximum spent time on the hubs (SAHCLP-2). The formulations for the two proposed models in this paper are as follows.

SAHCLP-1 Model:

$$\min \sum_{i=1}^n \sum_{k=1}^n \delta C_{ij} (O_i + D_i) X_{ik} + \sum_{i=1}^n \sum_{k=1}^n \sum_{l=1}^n \alpha C_{kl} Y_{kl}^i + \sum_{k=1}^n F_k X_{kk} \quad (6)$$

$$\min \sum_{k=1}^n P_k \times \frac{[\sum_{i=1}^n (O_i + D_i) X_{ik}]^2 + \sum_{i=1}^n (O_i + D_i) X_{ik}}{2} \quad (14)$$

subject to

$$\sum_{k=1}^n X_{ik} = 1 \quad i = 1, \dots, n \quad (7)$$

$$X_{ik} \leq X_{kk} \quad i = 1, \dots, n; k = 1, \dots, n \quad (8)$$

$$C_{ik} X_{ik} \leq r_k \quad i = 1, \dots, n; k = 1, \dots, n \quad (9)$$

$$\sum_{l=1}^n Y_{kl}^i + \sum_{j=1}^n W_{ij} X_{jk} = O_i X_{ik} + \sum_{l=1}^n Y_{lk}^i \quad i = 1, \dots, n; k = 1, \dots, n \quad (11)$$

$$Y_{kl}^i \leq M X_{kk} \quad i = 1, \dots, n; k = 1, \dots, n; l = 1, \dots, n \quad (12)$$

$$Y_{kl}^i \leq M X_{ll} \quad i = 1, \dots, n; k = 1, \dots, n; l = 1, \dots, n \quad (13)$$

$$X_{ik} \in \{0, 1\} \quad i = 1, \dots, n; k = 1, \dots, n \quad (14)$$

$$Y_{kl}^i \geq 0 \quad i = 1, \dots, n; k = 1, \dots, n; l = 1, \dots, n \quad (15)$$

SAHCLP-2 Model:

$$\min \sum_{i=1}^n \sum_{k=1}^n \delta C_{ij} (O_i + D_i) X_{ik} + \sum_{i=1}^n \sum_{k=1}^n \sum_{l=1}^n \alpha C_{kl} Y_{kl}^i + \sum_{k=1}^n F_k X_{kk} \quad (6)$$

$$\min \max_k \left(\frac{[\sum_{i=1}^n (O_i + D_i) X_{ik}]^2 + \sum_{i=1}^n (O_i + D_i) X_{ik}}{2} \times P_k \right) \quad (16)$$

subject to

$$\sum_{k=1}^n X_{ik} = 1 \quad i = 1, \dots, n \quad (7)$$

$$X_{ik} \leq X_{kk} \quad i = 1, \dots, n; k = 1, \dots, n \quad (8)$$

$$C_{ik} X_{ik} \leq r_k \quad i = 1, \dots, n; k = 1, \dots, n \quad (9)$$

$$\sum_{l=1}^n Y_{kl}^i + \sum_{j=1}^n W_{ij} X_{jk} = O_i X_{ik} + \sum_{l=1}^n Y_{lk}^i \quad i = 1, \dots, n; k = 1, \dots, n \quad (11)$$

$$Y_{kl}^i \leq M X_{kk} \quad i = 1, \dots, n; k = 1, \dots, n; l = 1, \dots, n \quad (12)$$

$$Y_{kl}^i \leq M X_{ll} \quad i = 1, \dots, n; k = 1, \dots, n; l = 1, \dots, n \quad (13)$$

$$X_{ik} \in \{0, 1\} \quad i = 1, \dots, n; k = 1, \dots, n \quad (14)$$

$$Y_{kl}^i \geq 0 \quad i = 1, \dots, n; k = 1, \dots, n; l = 1, \dots, n \quad (15)$$

For the second model, after linearizing the second objective function, we have:

SAHCLP-2* Model:

$$\min \sum_{i=1}^n \sum_{k=1}^n \delta C_{ij} (O_i + D_i) X_{ik} + \sum_{i=1}^n \sum_{k=1}^n \sum_{l=1}^n \alpha C_{kl} Y_{kl}^i + \sum_{k=1}^n F_k X_{kk} \quad (6)$$

$$\min \beta \quad (17)$$

subject to

$$\beta \geq \frac{[\sum_{i=1}^n (O_i + D_i) X_{ik}]^2 + \sum_{i=1}^n (O_i + D_i) X_{ik}}{2} \times P_k \quad k = 1, \dots, n \quad (18)$$

$$\sum_{k=1}^n X_{ik} = 1 \quad i = 1, \dots, n \quad (7)$$

$$X_{ik} \leq X_{kk} \quad i = 1, \dots, n; k = 1, \dots, n \quad (8)$$

$$C_{ik} X_{ik} \leq r_k \quad i = 1, \dots, n; k = 1, \dots, n \quad (9)$$

$$\sum_{l=1}^n Y_{kl}^i + \sum_{j=1}^n W_{ij} X_{jk} = O_i X_{ik} + \sum_{l=1}^n Y_{lk}^i \quad i = 1, \dots, n; k = 1, \dots, n \quad (11)$$

$$Y_{kl}^i \leq M X_{kk} \quad i = 1, \dots, n; k = 1, \dots, n; l = 1, \dots, n \quad (12)$$

$$Y_{kl}^i \leq M X_{ll} \quad i = 1, \dots, n; k = 1, \dots, n; l = 1, \dots, n \quad (13)$$

$$X_{ik} \in \{0, 1\} \quad i = 1, \dots, n; k = 1, \dots, n \quad (14)$$

$$Y_{kl}^i \geq 0 \quad i = 1, \dots, n; k = 1, \dots, n; l = 1, \dots, n \quad (15)$$

4. Proposed weight-based imperialist competitive algorithm (MOICA)

One of the approaches to solve a multi-objective optimization problem is to assign a weight w_i to each normalized objective function $f'_i(x)$ such that the problem is converted into a single objective problem with a scalar objective function as follows.

$$\min f = w_1 f'_1(x) + w_2 f'_2(x) + \dots + w_k f'_k(x) \quad (19)$$

where $f'_i(x)$ is the normalized objective function $f_i(x)$ with $\sum w_i = 1$. Solving such a problem with the objective function (17) for a given weight vector $w = \{w_1, w_2, \dots, w_k\}$ yields a single solution. If multiple solutions are desired, the problem should be solved different times with different weight combinations. For the given problem at hand, even finding a feasible solution is challenging. With this motivation, in this section, we develop an imperialist competitive algorithm (ICA) to solve the problem. The ICA is a new meta-heuristic algorithm introduced by Atashpas-Gargari and Lucas (2007) for solving continuous optimization problems. The general form of the ICA and the steps of the proposed solution algorithm are described in Sections 4.1 and 4.2, respectively.

4.1. Imperialist competitive algorithm

Evolutionary optimization algorithms are generally inspired by modelling the natural processes and other aspects of species evolution, especially human evolution. However, imperialist competitive algorithm (ICA) uses socio-political evolution of human as a source of inspiration for developing a strong optimization strategy (Atashpas-Gargari & Lucas, 2007). Imperialism is the policy of extending the power and the rule of a government beyond its own boundaries. A country may attempt to dominate others by direct rule or by less obvious means such as a control of markets for goods or raw materials. Atashpas-Gargari and Lucas (2007) described the algorithm as like other evolutionary ones, the method is initiated using an initial population and any individual of the population is called a country. Some of the best countries (in optimization terminology, countries with the least cost) are selected to be the imperialist states and the rest form the colonies of these imperialists. All the remained colonies of initial countries are divided among the mentioned imperialists based on their power. The power of each country is measured based on the objective function of the proposed model.

After dividing all colonies among imperialists and creating the initial empires, these colonies start moving toward their relevant imperialist country. This movement is a simple model of assimilation policy which is pursued by some of the imperialist states. The total power of an empire depends on both the power of the imperialist country and the power of its colonies. For this fact, we consider the total power of an empire as the sum of power of imperialist country and a percentage of mean power of its colonies. Then, the imperialistic competition begins among all the empires. Any empire that is not able to succeed in this competition will be eliminated from the competition. The imperialistic competition will gradually result in an increase in the power of powerful empires and a decrease in the power of weaker ones. Weak empires will lose their power, and ultimately they will collapse. The

movement of colonies toward their relevant imperialists along with competition among empires, and also the collapse mechanism will hopefully cause all the countries to converge to a state in which there is one empire in the world and all the other countries are colonies of that empire. In this ideal new world, colonies have the same position and power as the imperialist.

4.2. Improved imperialist competitive Algorithm

Since basic ICA is used for continuous problem, not only do we adopt this algorithm with discrete problems, but also we improve this algorithm by adding some new assimilation methods. For this fact, we use a crossover function of the genetic algorithm (GA) and the new near building policy. These methods lead to better approximated solutions closer to an optimal solution and less computational time in comparison with the traditional GA. Following, we explain the steps of the proposed ICA.

4.2.1. Representation of the solution

Any solution encoding procedure should show the location of hub nodes and the allocation of non-hub nodes to the hubs. One of the procedures uses integer numbers for presenting the given network. The solutions are presented as a vector, in which the length of the vector is equal to the number of nodes in the network. Each bit shows a node in the network, in which its value explains the number of the hub or the node is allocated to. Further, when the value of the bit is equal to the number of the node, the node is considered as a hub. For example, a sample solution is obtained as follows:

Ind= [1 7 3 7 3 3 7 1 9 9]

In this solution nodes 1, 3, 7 and 9 are hubs. Also, nodes 2 and 4 are allocated to hub 7, nodes 5 and 6 are allocated to hub 3, node 8 is allocated to hub 1, and node 10 is allocated to hub 9.

4.2.2. Initial population

The creation of the initial population is processed in three steps. In the first step, the number of hubs, p , is determined, randomly. The number of hubs in this study is not given so it can be any number of nodes in the network with minimum 2 and maximum n . In the second stage (i.e., location stage), p number of hubs are located randomly among n nodes of the network. Finally, in the allocation step, the remaining nodes are allocated to the located hubs based on their shortest distance from located hubs and the capacity of the mentioned hubs. The above process is applied iteratively to create the entire population.

4.2.3. Evaluation function

The solutions in the imperialist competitive algorithm are named as country. The evaluation function is an operation to evaluate how good the network configuration of each country is. Also, it makes the comparison between different solutions possible. In this paper, the evaluation function includes of calculating the value of the summation of the objective function of the network represented by each country. In order to prevent an object from being dominated by the other object, the authors used normalized weight base objective function. Let Z_1^* and Z_2^* be the optimal objective functions that are obtained by considering only one of the objective functions Z_1 and Z_2 (Malekinezhad et al., 2011). Therefore the evaluation function in the proposed algorithm is as follows.

$$c_i = w_1 \left\{ \frac{\left(\sum_{i=1}^n \sum_{k=1}^n \delta C_{ij} (O_i + D_i) X_{ik} + \sum_{i=1}^n \sum_{k=1}^n \sum_{l=1}^n \alpha C_{kl} Y_{kl}^i + \sum_{k=1}^n F_k X_{kk} \right) - Z_1^*}{Z_1^*} \right\} \\ + w_2 \left\{ \frac{\left(\sum_{k=1}^n P_k \times \frac{[\sum_{i=1}^n (O_i + D_i) X_{ik}]^2 + \sum_{i=1}^n (O_i + D_i) X_{ik}}{2} \right) - Z_2^*}{Z_2^*} \right\} \quad (20)$$

where w_i is generated randomly or determined by the DM, so that $\sum w_i = 1$ and c_i is the objective function value of the i th solution.

4.2.4. Imperialism

To produce the initial imperialist, we consider some of the best countries (those countries have the minimum cost $=N_{imp}$) as emperors. We consider the remaining countries (N_{col}) as colonies. After that, the remaining countries are allocated to emperors according to authority of each emperor, which is obtained by computing the normalized cost for each emperor as follows,

$$C_n = c_n - \max_i \{c_i\},$$

where c_n is the cost of the n th emperor, and C_n is the normalized cost for the n th emperor. Now, a proportional power for each emperor for allocating the colonies to them is computed as follows,

$$P_n = \frac{C_n}{\sum_{i=1}^{N_{imp}} C_i}.$$

Finally, a number of colonies, in which each emperor can seize, are:

$$N \times C_n = \text{round} \{P_n \times (N_{col})\},$$

where $N \times C_n$ is the number of colonies that the n th emperor can seize. After allocating the colonies to emperors, the imperialism competition will be started. Each emperor tries to develop its colonies.

4.2.5. Assimilation methods

After forming initial empires, the colonies in each of them start moving toward their relevant imperialist country. This movement is called the assimilation policy that was pursued by some of the imperialist states. In the following, three methods for assimilating colonies are presented.

4.2.5.1. Near building policy

In this section, a new method for assimilating colonies is explained. At first, all hubs of the emperor are determined. Second, half of them will be chosen and located instead of the same number of hubs in each colony, in which we want to assimilate. Those colonies, which will be selected for assimilating, are limited and they are chosen randomly among weaker colonies. After the hubs are located in colonies, we allocate the remaining nodes to the located hubs. In this process, if a colony gains better position (i.e., lower cost) than the emperor, the position of emperor and the best assimilated colony will be exchanged. For example, if the position of an emperor in search space is $\text{Emp} = [1 \ 2 \ 7 \ 6 \ 6 \ 7 \ 2 \ 2 \ 9 \ 10]$, the hub locations will be $[1 \ 2 \ 6 \ 7 \ 9 \ 10]$. Now we select half of the hub locations matrix, randomly. For example: hubs 6, 9 and 10 have been chosen. Then for each chosen, we locate nodes 6, 9 and 10 as hubs. Finally, we allocate the remaining nodes to these hubs.

4.2.5.2. Association policy

In this method, each emperor combines its features with colonies. The crossover in the GA is used. After each association, the country passes from the filtering step in the program to guarantee that the country will have a valid structure at the end of this process. For example, suppose that the emperor is $Emp = [1\ 2\ 7\ 6\ 6\ 7\ 2\ 2\ 9\ 10]$ and the colony is $Col = [2\ 2\ 3\ 4\ 7\ 3\ 7\ 8\ 8\ 2]$. Then, we choose the association point randomly. For example, if the association point is 6, then two solutions ($[1\ 2\ 7\ 6\ 6\ 7]$ and $[7\ 8\ 8\ 2]$) will be combined. Therefore, the colony are changed to $[1\ 2\ 7\ 6\ 6\ 7\ 7\ 8\ 8\ 2]$.

4.2.5.3. Revolution policy

In our proposed ICA, the revolution policy consists of creating a new solution. It likes the mutation operator in the GA. This method is similar to create the initial solution as mentioned in Section 3.2.2. In any imperialism, a number of colonies, which will be revolted, are limited. The authors set the revolution rate close to 0.1 (i.e., if the imperialism has 20 members, then just two of them will be revolted).

4.2.6. Cost of a imperialism

The total cost of imperialism is computed by:

$$T.C_n = Cost(emperor_n) + \gamma \cdot mean\{cost(Colonies\ of\ emperor_n)\},$$

where γ is the positive factor between 0 and 1. Also, $T \times C_n$ is the total cost of the n th imperialism. If the γ value is low, the cost of the n th imperialism closes to the cost of the n th emperor. The usual value γ is 0.3.

4.2.7. Imperialist competition

As all empires try to take the possession of the colonies of the other empires and control them, the imperialistic competition gradually brings about a decrease in the power of weaker empires and an increase in the power of more powerful ones (Atashpas-Gargari & Lucas, 2007). The imperialistic competition is modelled by just picking one (or some) of the weakest colonies of the weakest empires and making a competition among all empires to possess this (or these) colony (or colonies). During this competition, each emperor, who cannot develop its colonies, will be eliminated and seized with a stronger imperialism. This process is continued until it remains only one emperor. Also, the algorithm can be stopped after predefined iterations.

4.2.8. Pseudo code for imperialist competitive algorithm (ICA)

The main steps of the proposed ICA are summarized in the pseudo code shown in Fig. 2.

5. Experimental evaluation

This section evaluates the performances of the proposed imperialist competitive algorithm. These algorithms are coded and implemented in MATLAB 7.0 running on a PC with Pentium 4, CPU 1.4GHz, RAM 512 MB. Furthermore, for the optimal solution in small sizes, the authors use the CPLEX software.

5.1. Parameters tuning

It is worthy to note that the quality of an algorithm is significantly affected by the values of its parameters. In this section, the authors study the behaviour of the different parameters of the ICA. In order to tune the proposed ICA, the authors apply a full factorial design in the design of the experiment (DOE) method (Montgomery, 2000). Then four instances for tuning each set of

parameters are randomly generated, resulting in the total of 240 instances. The stopping criterion is $n \times m \times 0.4$ seconds of the computational time. This stopping criterion allows for more time as the number of nodes increases. We use the relative percentage deviation (RPD) for the objective function value as the common performance measure in order to compare the levels of parameters. The RPD is calculated by:

$$RPD = \frac{Alg_{sol} - Min_{sol}}{Min_{sol}} \times 100$$

where Alg_{sol} is the objective function value obtained for the proposed ICA, and Min_{sol} is the best solutions obtained for each instance by the CPLEX software.

The proposed ICA has three parameters, namely the number of imperialists (Imp-Num), the number of colonies (Popsiz), and the crossover point in assimilation procedure (CrossP). The considered levels of the parameters are shown in Table 1. The associated results are analyzed by the means of the analysis of the variance (ANOVA) method. The means plot and least significant differences (LSD) intervals at the 95% confidence level for the levels Imp-Num, Popsiz, and CrossP parameter factors are shown in Figs. 3 to 5, respectively.

1. Generate random sample of countries N
2. Initiate imperialism
 - a. **For** each country compute the evaluation function c_i
 - b. Sort the set N descending based on c_i
 - c. Select N_{imp} powerful countries as the emperor
 - d. Normalize the cost of each emperor C_i
 - e. Compute the authority of each emperor for seizing countries P_n
 - f. Assign N_{col} remained countries to emperor according to Step 2.e. by $N \times C_n$
3. **For** each imperialism **do**
 - a. Move the colony toward the relevant emperor (assimilating)
 - b. Compute the cost of assimilated countries
 - c. **If** the new colony gained lower cost than that of emperor **then**
 - ✓ Exchange the positions of that colony and the imperialist
 - d. Compute the total cost of imperialism $T \times C_n$
 - e. Select the weakest colony (colonies) from the weakest empire and give it (them) to the strongest empire
4. **If** there is an imperialism with no colonies **then**
 - ✓ Eliminate that imperialism
5. **If** stop conditions have been fulfilled **then** stop the algorithm or **go** to Step 3.

Fig. 2. Pseudo code for the CCA

In these figures, it can be seen that the Imp-Num = 8, Popsiz = 120, and CrossP = 2/4 results in a better output in comparison with the other values, statistically. Therefore, as the results of the analysis, we set the parameters: Imp-Num = 8, Popsiz = 120 and CrossP = 2/4.

Table 1
Considered levels of ICA parameters

Parameters	No. of levels	Levels			
Imp-Num	3	6	8	10	-
Popsiz	4	70	100	120	150
CrossP	3	1/4	2/4	3/4	-

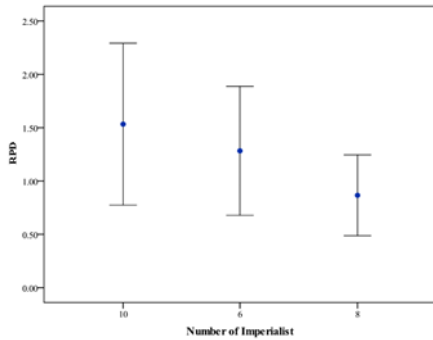


Fig. 3. Mean plot and LSD intervals for different levels of the Imp-Num

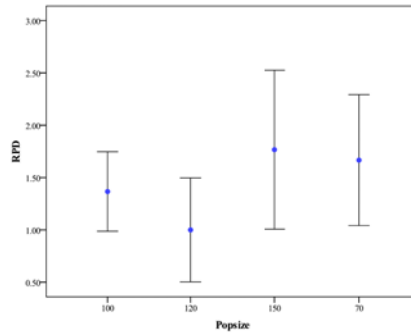


Fig. 4. Mean plot and LSD intervals for different levels of the Popsiz

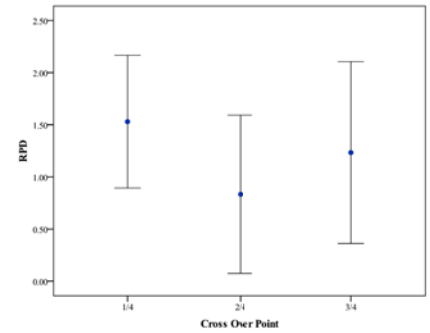


Fig. 5. Mean plot and LSD intervals for different levels of CrossP

5.2. Experimental results

To test the performance of the proposed ICA, we compare the quality of its solutions with those of optimal solutions obtained by the developed mathematical programming for small-sized instances. Table 2 shows the parameter values and probability distributions used for randomly generated test instances. We have 100 (i.e., 5×2×2×5) classes of the problem.

The results of the comparison with different values of parameters are shown in Tables 3 to 7. For each instance, we run the problem by the meta-heuristic algorithm four times and insert the best one of them in the tables. The first three columns are the input parameters of the problem. The fourth column is the different weights for achieving five Pareto solutions. The fifth column is the objective function of the optimal solution obtained by the CPLEX software. The sixth column is the objective function of our proposed ICA. The seventh column is the gap of the ICA in comparison with the optimal solution. This column is computed by:

$$\frac{Obj_{heuristic} - Obj_{optimal\ solution}}{Obj_{optimal\ solution}} \times 100.$$

Table 2
Generation of random data

	Parameters					<i>P</i>	<i>F</i>	<i>W</i>	<i>C</i>	<i>r</i>
	<i>n</i>	α	δ	<i>w</i>						
Value	10	0.75	0.90	(0.40, 0.60)	U ~ (10,20)	U ~ (200,600)	U ~ (1,10)	U ~ (1,10)	U ~ (1,10)	
	15	0.80	0.95	(0.45, 0.55)						
	20			(0.50, 0.50)						
	40			(0.55, 0.45)						
	70			(0.60, 0.40)						

Table 3Computational comparison of the IP model and the heuristic algorithm for $n = 10$

Parameters				CPLEX	ICA		
n	α	δ	$w(w_1, w_2)$	OFV*	OFV	Gap (%)	
10	0.75	0.90	(0.40, 0.60)	1.424	1.424	0	
			(0.45, 0.55)	1.470	1.470	0	
			(0.50, 0.50)	1.531	1.531	0	
			(0.55, 0.45)	1.433	1.433	0	
			(0.60, 0.40)	1.404	1.404	0	
	0.95		(0.40, 0.60)	1.309	1.309	0	
			(0.45, 0.55)	1.402	1.402	0	
			(0.50, 0.50)	1.418	1.418	0	
			(0.55, 0.45)	1.176	1.176	0	
			(0.60, 0.40)	1.038	1.038	0	
	0.80	0.90		(0.40, 0.60)	1.132	1.132	0
				(0.45, 0.55)	1.364	1.364	0
				(0.50, 0.50)	1.421	1.421	0
				(0.55, 0.45)	1.322	1.322	0
				(0.60, 0.40)	1.212	1.212	0
		0.95		(0.40, 0.60)	1.582	1.582	0
				(0.45, 0.55)	1.595	1.595	0
				(0.50, 0.50)	1.642	1.642	0
				(0.55, 0.45)	1.486	1.486	0
				(0.60, 0.40)	1.411	1.411	0
Average						0	

*Objective function value

Table 4Computational comparison of the IP model and the heuristic algorithm for $n = 15$

Parameters				CPLEX	ICA		
n	α	δ	$w(w_1, w_2)$	OFV	OFV	Gap (%)	
15	0.75	0.90	(0.40, 0.60)	1.422	1.422	0	
			(0.45, 0.55)	1.445	1.445	0	
			(0.50, 0.50)	1.497	1.512	0.98	
			(0.55, 0.45)	1.434	1.434	0	
			(0.60, 0.40)	1.398	1.398	0	
	0.95		(0.40, 0.60)	1.327	1.341	1.02	
			(0.45, 0.55)	1.352	1.352	0	
			(0.50, 0.50)	1.375	1.375	0	
			(0.55, 0.45)	1.311	1.311	0	
			(0.60, 0.40)	1.294	1.294	0	
	0.80	0.90		(0.40, 0.60)	1.427	1.441	1.01
				(0.45, 0.55)	1.432	1.432	0
				(0.50, 0.50)	1.476	1.488	0.83
				(0.55, 0.45)	1.411	1.411	0
				(0.60, 0.40)	1.402	1.413	0.76
		0.95		(0.40, 0.60)	1.371	1.388	1.22
				(0.45, 0.55)	1.395	1.395	0
				(0.50, 0.50)	1.423	1.436	0.90
				(0.55, 0.45)	1.404	1.404	0
				(0.60, 0.40)	1.379	1.393	1.00
Average						0.38	

Table 5Computational comparison of the IP model and the heuristic algorithm for $n = 20$

Parameters				CPLEX	ICA				
n	α	δ	$w(w_1, w_2)$	OFV	OFV	Gap (%)			
20	0.75	0.90	(0.40, 0.60)	1.498	1.535	2.5			
			(0.45, 0.55)	1.525	1.525	0			
			(0.50, 0.50)	1.546	1.572	1.7			
			(0.55, 0.45)	1.473	1.473	0			
			(0.60, 0.40)	1.459	1.491	2.2			
	0.95			(0.40, 0.60)	1.257	1.273	1.3		
				(0.45, 0.55)	1.264	1.303	3.1		
				(0.50, 0.50)	1.285	1.312	2.1		
				(0.55, 0.45)	1.164	1.164	0		
				(0.60, 0.40)	1.153	1.186	2.9		
	0.80	0.90			(0.40, 0.60)	1.350	1.382	2.4	
					(0.45, 0.55)	1.365	1.365	0	
					(0.50, 0.50)	1.392	1.392	0	
					(0.55, 0.45)	1.263	1.302	3.1	
					(0.60, 0.40)	1.235	1.272	3.0	
		0.95				(0.40, 0.60)	1.517	1.556	2.6
						(0.45, 0.55)	1.532	1.566	2.2
						(0.50, 0.50)	1.570	1.570	0
						(0.55, 0.45)	1.523	1.523	0
						(0.60, 0.40)	1.497	1.551	3.6
Average						1.64			

Table 6Computational comparison of the IP model and the heuristic algorithm for $n = 40$

Parameters				CPLEX	ICA			
n	α	δ	$w(w_1, w_2)$	OFV	OFV			
40	0.75	0.90	(0.40, 0.60)	-	0.878			
			(0.45, 0.55)	-	0.922			
			(0.50, 0.50)	-	1.028			
			(0.55, 0.45)	-	0.591			
			(0.60, 0.40)	-	0.563			
		0.95				(0.40, 0.60)	-	0.651
						(0.45, 0.55)	-	0.853
						(0.50, 0.50)	-	0.914
						(0.55, 0.45)	-	0.579
						(0.60, 0.40)	-	0.561
	0.80	0.90			(0.40, 0.60)	-	0.986	
					(0.45, 0.55)	-	1.132	
					(0.50, 0.50)	-	1.210	
					(0.55, 0.45)	-	0.892	
					(0.60, 0.40)	-	0.826	
		0.95				(0.40, 0.60)	-	0.805
						(0.45, 0.55)	-	0.973
						(0.50, 0.50)	-	1.113
						(0.55, 0.45)	-	0.819
						(0.60, 0.40)	-	0.874

For large-sized problems, the CPLEX software spends more than 10 hours for achieving the optimal solutions. Therefore, values Z_1 and Z_2 are calculated by interrupting the CPLEX software after $n \times m \times 0.4$ seconds. So, the following equation is used to compute the normalized objective function instead of Eq. (20).

$$c_i = w_1 \left\{ \frac{\left(\sum_{i=1}^n \sum_{k=1}^n \delta C_{ij} (O_i + D_i) X_{ik} + \sum_{i=1}^n \sum_{k=1}^n \sum_{l=1}^n \alpha C_{kl} Y_{kl}^i + \sum_{k=1}^n F_k X_{kk} \right)}{Z_1} \right\} \\ + w_2 \left\{ \frac{\left(\sum_{k=1}^n P_k \times \frac{\left[\sum_{i=1}^n (O_i + D_i) X_{ik} \right]^2 + \sum_{i=1}^n (O_i + D_i) X_{ik}}{2} \right)}{Z_2} \right\} \quad (21)$$

The CPLEX software spends more than 10 hours in solving instances with 40 and 70 nodes. So, we run our proposed ICA alone and compare the results for these instances. The proposed algorithm spends less than 600 seconds for 70 nodes problems. So, it is more reasonable to use the meta-heuristic approaches for achieving the near-optimal solution with a little gap. The gaps of the proposed ICA are 0.0, 0.38 and 1.64 for 10, 15 and 20 nodes, respectively.

Table 7

Computational comparison of the IP model and the heuristic algorithm for $n = 70$

n	Parameters			CPLEX	ICA
	α	δ	$w(w_1, w_2)$	OFV	OFV
70	0.75	0.90	(0.40, 0.60)	-	0.638
			(0.45, 0.55)	-	0.659
			(0.50, 0.50)	-	0.714
			(0.55, 0.45)	-	0.624
			(0.60, 0.40)	-	0.621
	0.95	0.90	(0.40, 0.60)	-	0.751
			(0.45, 0.55)	-	0.803
			(0.50, 0.50)	-	0.838
			(0.55, 0.45)	-	0.779
			(0.60, 0.40)	-	0.783
	0.80	0.90	(0.40, 0.60)	-	0.953
			(0.45, 0.55)	-	0.974
			(0.50, 0.50)	-	1.093
			(0.55, 0.45)	-	0.948
			(0.60, 0.40)	-	0.953
	0.95	0.90	(0.40, 0.60)	-	0.898
			(0.45, 0.55)	-	0.943
			(0.50, 0.50)	-	0.987
			(0.55, 0.45)	-	0.879
			(0.60, 0.40)	-	0.778

A paired t -test is conducted to see whether the significant difference exists between the obtained solution of the proposed ICA and the optimal solution of the CPLEX software. Let D_i be equal to the difference between the computed values of two methods for test problem i . So the statistics are as follows:

$$t = \frac{\sqrt{n} \times \bar{D}}{S_D}, \quad (22)$$

$$\text{where } \bar{D} = \frac{\sum D_i}{n} \text{ and } S_D = \sqrt{\frac{\sum (D_i - \bar{D})^2}{n-1}}.$$

We conduct the paired t -test by 30 test problems in the SPSS software. The calculated statistic is equal to 4.059. By referencing to the t table, for 29 df (i.e., degree of freedom), the significance (2-tailed) is 0.00. The detailed statistics are shown in Table 8.

Table 8

Detailed statistics of the paired t -test

Paired Differences	Mean	Std. deviation	Std. Error mean	95% confidence interval of the difference		t	df	Sig. (2-tailed)
				Lower	Upper			
				Pair-1 CPLEX - ICA	0.00925			

6. Conclusion

In this paper, a different model of the capacitated single allocation hub covering location problem has been presented. Instead of using capacity constraints to limit the amount of flows that can be received by the hubs, the second objective function has been introduced to the presented model (besides the traditional cost minimizing function), that tries to minimize both of the transportation time in the hub network and the service time in the hubs. According to the basic proposed model, the third model has been also proposed, in which the model has minimized the total transportation cost and minimized the maximum spent service time in the hubs. To solve the problem, the authors have proposed a new imperialist competitive algorithm (ICA). The performance of the proposed ICA has been compared with those optimal solutions obtained by the mathematical formulation. The related results have been shown in Tables 3 to 7. The maximum gap of the proposed algorithm is 3.3% for 20 nodes instance. For sizes 40 and 70, the CPLEX software has spent more than 4 hours in order to gain the optimal solutions. Therefore for sizes 40 and 70, the proposed algorithm has run alone. Developing other meta-heuristics for the problem given in this paper is an interesting further research direction. Also, analyzing the problem with arc constraints as the assumption can be a valuable research subject.

References

- Alumur, S., Kara, B.Y. (2009). The design of single allocation incomplete hub networks. *Transportation Research Part B*, 43 (10), 936–951.
- Atashpas-Gargari E. and Lucas C. (2007). Colonial competitive algorithm. IEEE Congress on Evolutionary Computation.
- Aykin T. (1994). Lagrangean relaxation based approaches to capacitated hub-and-spoke network design problems. *European Journal of Operational Research*, 79, 501–523.
- Aykin T. (1988). On the location of hub facilities. *Transportation Science*, 22 (2), 155–157.
- Boland, N., Krishnamoorthy, M., Ernst, A., T., Ebery, J. (2004). Preprocessing and cutting for multiple allocation hub location problems. *European Journal of Operational Research*, 155 (3), 638–653.
- Campbell, J.F. (1994). Integer programming formulations of discrete hub location problems. *European Journal of Operational Research*, 72, 387–405.

- Campbell, J.F. (1996). Hub location and the p-hub median problem. *Operations Research*, 44 (6), 923–935.
- Campbell, J.F., Ernst, A.T., Krishnamoorthy, M. (2002). *Hub location problems*. In: Z. Drezner and H. Hamacher (Eds.), *Facility location: Applications and theory*. Berlin: Springer, 373–406.
- Campbell, J.F., Ernst, A.T., Krishnamoorthy, M. (2005). Hub arc location problems: part I – introduction and results. *Management Science*, 51 (10), 1540–1555.
- Contreras, I., Fernández, E., Marín, A. (2009a). Tight bounds from a path based formulation for the tree of hub location problem. *Computers & Operations Research*, 36 (12), 3117–3127.
- Contreras, I., Díaz, J., Fernández, E. (2009b). Lagrangean relaxation for the capacitated hub location problem with single assignment. *OR Spectrum*, 31 (3), 483–505.
- Contreras, I., Fernández, E., Marín, A. (2010). The tree of hubs location problem. *European Journal of Operational Research*, 202 (2), 390–400.
- Costa, M., Captivo, M., Clímaco, J. (2008). Capacitated single allocation hub location problem – a bi-criteria approach. *Computers & Operations Research*, 35 (11), 3671–3695.
- Ebery, J., Krishnamoorthy, M., Ernst, A.T., Boland, N. (2000). The capacitated multiple allocation hub location problems: formulations and algorithms. *European Journal of Operational Research*, 120, 614–31.
- Ernst, A.T., Krishnamoorthy, M. (1996). Efficient algorithms for the uncapacitated single allocation p-hub median problem. *Location Science*, 4 (3), 139–54.
- Ernst, A.T., Krishnamoorthy, M. (1996). An exact solution approach based on shortest paths for p-hub median problems. *INFORMS Journal on Computing*, 10 (2), 149–62.
- Ernst, A.T., Krishnamoorthy, M. (1998). Exact and heuristic algorithms for the uncapacitated multiple allocation p-hub median problem. *European Journal of Operational Research*, 104, 100–12.
- Ernst, A.T., Krishnamoorthy, M. (1999). Solution algorithms for the capacitated single allocation hub location problem. *Annals of Operations Research*, 86, 141–59.
- Ernst, A.T., Jiang, H., Krishnamoorthy, M. (2005). Reformulations and computational results for uncapacitated single and multiple allocation hub covering problems. Unpublished Report, CSIRO Mathematical and Information Sciences, Australia.
- Jabal-Ameli, M.S., Aryanezhad, M.B., Ghaffari-Nasab, N. (2011). A variable neighborhood descent based heuristic to solve the capacitated location-routing problem. *International Journal of Industrial Engineering Computations*, 2, 141–154.
- Kara, B.Y., Tansel, B.C. (2003). The single-assignment hub covering problem: Models and linearizations. *Journal of the Operational Research Society*, 54, 59–64.
- Labbe', M., Yaman, H. (2004). Projecting flow variables for hub location problems. *Networks*, 44 (2), 84–93.
- Malekinezhad, A., Shirazi, E., Aryanezhad, M., B. (2011). A multi-objective set covering problem: A case study of warehouse allocation in truck industry. *Management Science Letters*, 1, 73–80.
- Marin, A. (2005). Formulating and solving splittable capacitated multiple allocation hub location problems. *Computers & Operational Research*, 32 (12), 3093–3109.
- Montgomery DC (2000). *Design and analysis of experiments*, 5th edn. Wiley, London.
- Nickel, S., Schobel, A., Sonneborn, T. (2001). *Hub location problems in urban traffic networks*. In: Niittymaki, J., Pursula, M. (Eds.), *Mathematics Methods and Optimization in Transportation Systems*. Kluwer Academic Publishers, 1–12.
- O'Kelly ME. (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, J. (1995). Lower bounds for the hub location problem. *Management Science*, 41 (4), 713–721.
- Skorin-Kapov, D., Skorin-Kapov, J., O'Kelly, M. (1996). Tight linear programming relaxations of uncapacitated p-hub median problems. *European Journal of Operational Research*, 94, 582–593.
- Yaman, H. Star p-hub median problem with modular arc capacities. (2008). *Computers & Operations Research*, 35 (9), 3009–3019.