A HYBRID ALGORITHM FOR THE UNCERTAIN INVERSE p-MEDIAN LOCATION PROBLEM

Akram Soltanpour, Fahimeh Baroughi and Behrooz Alizadeh

Abstract. In this paper, we investigate the inverse p-median location problem with variable edge lengths and variable vertex weights on networks in which the vertex weights and modification costs are the independent uncertain variables. We propose a model for the uncertain inverse p-median location problem with tail value at risk objective. Then, we show that it is NP-hard. Therefore, a hybrid particle swarm optimization algorithm has been presented to obtain the approximate optimal solution of the proposed model. The algorithm contains expected value simulation and tail value at risk simulation.
Keywords: p-median location problem; inverse optimization; Hybrid algorithm; nonlinear programming.

1. Introduction

One of the important aspects of location problems which has recently been studied by many researchers is the p-median location problem which can be stated as follows. Let $N = (V, E)$ be an undirected connected network with vertex set V, $|V| = n$, and edge set E, $|E| = m$. The distance between two points on N is equal to the length of the shortest path connecting these two points. Each vertex is associated with a nonnegative weight that is the demand of the client at this vertex. In a p-median problem on a network, the aim is to find p locations for establishing facilities on edges or vertices of the network such that the sum of the weighted distances from the clients to the closest facility becomes minimum. In the context of the p-median location problems, the interested reader is referred to papers [1, 7, 8, 11, 12, 13, 17, 20, 21, 28, 37, 39].

In recent years, inverse location problems have found an increasing interest. In an inverse location problem the goal is to modify parameters of the problem at

minimum cost such that a prespecified solution becomes optimal. Burkard et al. investigated the inverse 1-median problem with variable vertex weights on a tree network and also on a plane and presented algorithms in $O(n \log n)$ time for them [9]. Also they proposed an algorithm in $O(n^2)$ time for the problem under investigation on cycles [10]. Baroughi et al. [3] proved that the inverse p-median location problem (IpMLP) on general networks is NP-hard. For a survey on the inverse p-median location problems, we refer the interested reader to [16, 18, 19, 23, 30, 36].

In the real life, we are usually faced with various types of uncertainty. For example, in location problems, we are usually not sure of the vertex weights, the travel times between vertices, the establishing costs of facilities and the vertex weights or edge lengths modification costs of a network. The uncertainty theory that proposed by Liu [25] is a suitable tool to deal with these parameters. Some researchers applied the uncertainty theory to deal with the location problems,for example Gao [14] modeled the single facility location problems with uncertain demands. Wen et al. [43] investigated the capacitated facility location-allocation problem with uncertain demands and also Nguyen and Chi [31] studied inverse 1-median problem on a tree with uncertain costs and showed that the inverse distribution function of the minimum cost can be obtained at $O(n^2 \log n)$ time. For a survey on uncertain location problems, we refer the interested reader to [15, 22, 27, 34, 40, 46].

The uncertainty leads to the risk. Liu in [26] introduced the risk concept in the uncertain environment. Measuring the risk is one of the important steps in the decision making process. The risk metrics contain techniques and data sets used to calculate the risk value of the problem under investigation. Tail value at risk (TVaR) metric [32] is one of the measures of the risk that is widely acceptable among industry segments and market participants.

In the risk management related to location problems, Berman et al. [6] studied the effect of a decision maker's risk attitude on the median and center location problems, with uncertain demand in the mean-variance framework. Wang et al. [41] investigated a two-stage fuzzy facility location problem with value at risk. Wagner et al. [42] developed and examined a new algorithm for solving the p-median problem when the demands are probabilistic and correlated. For a survey on the risk management in the location problems with fuzzy variables, see, e.g. [5, 44].

In this paper, we concentrate on IpMLP with variable edge lengths and variable vertex weights on networks. We assume that the vertex weights and modification costs are the independent uncertain variables. We propose a model for the uncertain inverse p-median location problem with tail value at risk objective and expected value constraints and show that the problem is NP-hard. Considering the uncertain and NP-hard nature in uncertain IpMLP (UIpMLP), evolutionary and meta-heuristics algorithms can be used to UIpMLP for successful generation of optimal solutions. Hence, we present a hybrid particle swarm optimization algorithm which contains expected value simulation and tail value at risk simulation to obtain the approximate optimal solution of the proposed model.

Based on our knowledge, there are two papers on the implementation of meta-heuristic algorithms to the inverse location problems until now. Alizadeh and

Bakhteh [2] studied the general IpMLPs on networks and presented a modified firefly algorithm for the problem under investigation. Mirzapolis Adeh et al. [29] investigated the general inverse ordered p-median location problem on crisp networks and designed a modified particle swarm optimization (PSO) algorithm for it. There is no scientific paper on implementation of hybrid metaheuristic algorithms on IpMLPs in uncertain networks. However, many papers can be found in the literature for other classical location problems on uncertain networks. Bashiri et al. [4] modeled fuzzy capacitated p-hub center problem and presented a genetic algorithm for the problem. Huang and Hao [22] modeled uncapacitated facility location problem with uncertain customers positions and provided a hybrid intelligent algorithm for solving it. In 2018 Rahmaniani et al. [35] proposed an efficient hybrid solution algorithm for the capacitated facility location-allocation problem under uncertainty. Yang et al. [45] presented an improved hybrid particle swarm optimization algorithm for fuzzy p-hub center problem.

The article is organized as follows: In the next section, we first introduce uncertainty theory and TVaR metric in an uncertain environment. Then, we discuss uncertain optimization model and present a new model with TVaR objective and expected value constraints. In Section 3., we first introduce IpMLP with variable edge lengths and variable vertex weights on networks and then investigate the problem with uncertain vertex weights and uncertain modification costs. A model for the uncertain inverse p-median location problem (UIpMLP) with TVaR objective is presented and it is shown that the problem under investigation is NP-hard. Then, we present a hybrid PSO algorithm to obtain the approximate optimal solution of the proposed model, which it contains expected value simulation and TVaR simulation. Finally, to show the effectiveness of the proposed hybrid PSO algorithm, we give a numerical example. Section 4. gives a brief conclusion to this paper.

2.   Preliminaries

In this section, we first present some definitions and theorems of the uncertainty theory and TVaR metric in an uncertain environment. Then, we introduce the uncertain optimization model and present a new model with TVaR objective and expected value constraints.

2.1.   Uncertainty theory

Let Γ be a nonempty set and Θ be a σ-algebra over Γ. An uncertain measure is a set function $\mathcal{M} : \Theta \to [0, 1]$ that satisfies in normality, duality and subadditivity axioms. The triple $(\Gamma, \Theta, \mathcal{M})$ is called an uncertainty space.

Definition 2.1. (Liu[25]). Let $(\Gamma, \Theta, \mathcal{M})$ be an uncertainty space. A measurable function θ from $(\Gamma, \Theta, \mathcal{M})$ to the set of real numbers is called an uncertain variable.

Definition 2.2. (Liu[25]). Let θ be an uncertain variable. For any real number x, the function $\Upsilon(x) = \mathcal{M}\{\theta \leq x\}$ is called an uncertainty distribution of θ.

Definition 2.3. (Liu[25]). Let θ_i, $i = 1, \ldots, n$, be the uncertain variables. We call θ_i, $i = 1, \ldots, n$, independent if for any Borel sets $B_1, B_2, \ldots, B_n$ of real numbers,

$$\mathcal{M}\left\{\bigcap_{i=1}^{n}\{\theta_i \in B_i\}\right\} = \bigwedge_{i=1}^{n} \mathcal{M}\{\theta_i \in B_i\}.$$

Definition 2.4. (Liu[25]). The expected value of the uncertain variable θ is defined as

$$E[\theta] = \int_{0}^{+\infty} \mathcal{M}\{\theta \geq r\}dr - \int_{-\infty}^{0} \mathcal{M}\{\theta \leq r\}dr,$$

provided that at least one of the two integral is finite.

A real valued function $f(x_1, x_2, ..., x_n)$ is said to be strictly increasing if $f(x_1, x_2, \ldots, x_n) > f(y_1, y_2, \ldots, y_n)$ when $x_i > y_i$ for $i = 1, 2, \ldots, n$.

Theorem 2.1. (Liu[25]). *Let θ_i, $i = 1, 2, \ldots, n$, be the independent uncertain variables and Υ_i^{-1}, $i = 1, 2, \ldots, n$, be the inverse uncertainty distributions of θ_i. Also, let $f(x_1, x_2, \ldots, x_n)$ be a strictly increasing function with respect to x_i, $i = 1, 2, \ldots, n$. Then the uncertain variable $\vartheta = f(\theta_1, \theta_2, \ldots, \theta_n)$ has the following inverse uncertainty distribution*

$$\Upsilon^{-1}(\alpha) = f(\Upsilon_1^{-1}(\alpha), \ldots, \Upsilon_n^{-1}(\alpha)),$$

and also it has the following expected value

$$E[\vartheta] = \int_{0}^{1} f(\Upsilon_1^{-1}(\alpha), \ldots, \Upsilon_n^{-1}(\alpha))d\alpha.$$

2.2. TVaR metric in an uncertain environment

The risk demonstrates a situation, in which there is a chance of loss or danger. The quantification of the risk is a key step towards the management and mitigation of the risk. In this section, we introduce the definition of the TVaR metric to account the probability of loss and the severity of the loss in an uncertain environment [32].

In order to define the TVaR metric, we first introduce the definition of the loss function.

Definition 2.5. (Liu[26]). Consider θ_i, $i = 1, 2, \ldots, n$, as the uncertain factors of a system. A function f is said to be a loss function if some specified loss occurs if and only if

$$f(\theta_1, \theta_2, \ldots, \theta_n) > 0.$$

In the uncertain environment, TVaR of the loss function is defined as follows.

Definition 2.6. (*Peng*[32]). Let θ_i, $i = 1, 2, \ldots, n$, be the uncertain factors and f be the loss function of a system. Then TVaR of f is defined as

$$TVaR_\beta = \frac{1}{\beta} \int_0^\beta sup\left\{\lambda \mid \mathcal{M}\left\{f(\theta_1, \theta_2, \ldots, \theta_n) \geq \lambda\right\} \geq \gamma\right\} d\gamma,$$

for each given risk confidence level $\beta \in (0, 1]$.

Theorem 2.2. (*Peng*[32]). *Let θ_i, $i = 1, 2, \ldots, n$, be the uncertain factors of a system and Υ_i^{-1}, $i = 1, 2, \ldots, n$, be the inverse uncertainty distributions of θ_i. Also assume that the loss function $f(x_1, x_2, \ldots, x_n)$ is a strictly increasing function with respect to x_i, $i = 1, 2, \ldots, n$. Then, for each risk confidence level $\beta \in (0, 1]$, we have*

$$TVaR_\beta = \frac{1}{\beta} \int_0^\beta f(\Upsilon_1^{-1}(1 - \gamma), \Upsilon_2^{-1}(1 - \gamma), \ldots, \Upsilon_n^{-1}(1 - \gamma)) d\gamma.$$

2.3. Uncertainty optimization

Let $x = (x_1, x_2, \ldots, x_n)$ be a decision vector, and $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$ be an uncertain vector. Consider the following optimization model.

(2.1)
$$\begin{aligned}
\min \quad & f(x, \theta) \\
\text{s.t.} \quad & g_j(x, \theta) \leq 0 \quad j = 1, \ldots, p, \\
& z_l(x) \leq 0 \qquad l = 1, \ldots, m, \\
& x \geq 0,
\end{aligned}$$

where f and g_j, $j = 1, \ldots, p$ are uncertain functions and z_l, $l = 1, \ldots, m$ are crisp functions.

Since the objective function of the model (2.1) involves uncertainty, it cannot be directly optimized. Therefore, by considering $f(x, \theta)$ as a loss function, we minimize its TVaR. In addition, since the uncertain constraints do not define a crisp feasible set, we use the expected value of constraints. Thus, the model (2.1) can be reformulated as

(2.2)
$$\begin{aligned}
\min \quad & TVaR_\alpha(f(x, \theta)) \\
\text{s.t.} \quad & E(g_j(x, \theta)) \leq 0 \quad j = 1, \ldots, p, \\
& z_l(x) \leq 0 \qquad l = 1, \ldots, m, \\
& x \geq 0.
\end{aligned}$$

According to Theorems 2.1 and 2.2, we can rewrite the problem (2.2) as follows:

(2.3)
$$\begin{aligned}
\min \quad & \frac{1}{\beta} \int_0^\beta f(x, \Upsilon_1^{-1}(1 - \gamma), \Upsilon_2^{-1}(1 - \gamma), \ldots, \Upsilon_n^{-1}(1 - \gamma)) d\gamma \\
\text{s.t.} \quad & \int_0^1 g_j(x, \Upsilon_1^{-1}(\alpha), \Upsilon_2^{-1}(\alpha), \ldots, \Upsilon_n^{-1}(\alpha)) d\alpha \leq 0 \qquad j = 1, \ldots, p, \\
& z_l(x) \leq 0 \quad l = 1, \ldots, m, \\
& x \geq 0,
\end{aligned}$$

where $g_j(x, \theta_1, \theta_2, \ldots, \theta_n)$ is strictly increasing with respect to $\theta_1, \theta_2, \ldots, \theta_n$.

3. Problem definition

In this section, we first introduce IpMLP with variable edge lengths and variable vertex weights on networks and then investigate the problem with uncertain vertex weights and uncertain modification costs. A model for UIpMLP with TVaR objective and expected value constraints is presented. To solve the proposed model, we present a hybrid PSO algorithm which contains expected value simulation and TVaR simulation.

3.1. UIpMLP on networks

We can express IpMLP with variable edge lengths and variable vertex weights as follows: Let $N = (V, E)$ with $|V| = n$ and $|E| = m$ be a connected network. Also let vertex $v \in V$ have a positive weight $w(v)$ and edge $e \in E$ have a positive length ℓ_e. In an IpMLP on networks, a set of vertices $\{m_1, \ldots, m_p\}$ is given. The goal is to modify $w(v)$, $v \in V$, and ℓ_e, $e \in E$, at minimum total cost such that the given set becomes a p-median of modified location problem. Let us consider nonnegative costs c_e^+ and c_v^+, if ℓ_e and $w(v)$ are increased by one unit, respectively. Also we consider nonnegative costs c_e^- and c_v^-, if ℓ_e and $w(v)$ are decreased by one unit, respectively. Let p_e, q_e, p_v and q_v be the amounts by which the edge length ℓ_e and the vertex weight $w(v)$ are increased and decreased, respectively. We let p_e, q_e, p_v and q_v obey the upper bounds u_e^+, u_e^-, u_v^+, u_v^-. In addition, assume that $\mathcal{S}$ is the set of all subsets $S \subseteq V$ with $|S| = p$. Thus, IpMLP on N can be stated as follows.

Change ℓ_e, $e \in E$, to $\tilde{\ell}_e = \ell_e + p_e - q_e$ and $w(v)$, $v \in V$, to $\tilde{w}(v) = w(v) + p_v - q_v$ such that

(i) The set $\{m_1, \ldots, m_p\}$ becomes a p-median of N with respect to $\tilde{\ell}$ and $\tilde{w}(v)$, i.e.,

$$(3.1) \qquad \sum_{v \in V} \tilde{w}(v) \min_{i=1,\ldots,p} d_{\tilde{\ell}}(v, m_i) \leq \sum_{v \in V} \tilde{w}(v) \min_{k \in S} d_{\tilde{\ell}}(v, v_k) \quad \forall \, S \in \mathcal{S},$$

(ii) The bound constraints are satisfied:

$$(3.2) \qquad\qquad 0 \leq p_e \leq u_e^+, \quad 0 \leq q_e \leq u_e^- \quad \forall \, e \in E,$$

$$(3.3) \qquad\qquad 0 \leq p_v \leq u_v^+, \quad 0 \leq q_v \leq u_v^- \quad \forall \, v \in V,$$

(iii) The objective function

$$\sum_{e \in E} (c_e^+ p_e + c_e^- q_e) + \sum_{v \in V} (c_v^+ p_v + c_v^- q_v)$$

becomes minimum.

This formulation of IpMLP is a nonlinear programming model. In the following, we consider IpMLP with uncertain vertex weights and uncertain modification costs.

Let $N = (V, E)$ be a network with independent uncertain vertex weights $\theta_v, v \in V$. Also let $w(v)$ be a parameter on each vertex $v \in V$, which will be changed to $\tilde{w}(v)$. In addition, suppose that θ_v relates to this parameter, i.e., for each vertex $v \in V$, we have an original weight $\theta(w(v))$ and also a new weight $\theta(\tilde{w}(v))$. Let ϑ_v^+ and ϑ_v^- be the independent uncertain variables with respect to the costs c_v^+ and c_v^-, for all $v \in V$, and ϑ_e^+ and ϑ_e^- be the independent uncertain variables with respect to the costs c_e^+ and c_e^-, for all $e \in E$, respectively.

Let us assume that we are given a set of vertices $\{m_1, \ldots, m_p\}$. In an UIpMLP, the goal is to find $\tilde{\ell}_e = \ell_e + p_e - q_e$ and $\tilde{w}(v) = w(v) + p_v - q_v$ such that $\{m_1, \ldots, m_p\}$ becomes a p-median of the problem with respect to $\theta_v(\tilde{w}(v))$ and $\tilde{\ell}_e$, $v \in V$, $e \in E$, and the total cost

$$\sum_{v \in V}(\vartheta_v^+ p_v + \vartheta_v^- q_v) + \sum_{e \in E}(\vartheta_e^+ p_e + \vartheta_e^- q_e)$$

is minimized.

Therefore, we can model UIpMLP as follows.

$$\min \quad \left[\sum_{v \in V}(\vartheta_v^+ p_v + \vartheta_v^- q_v) + \sum_{e \in E}(\vartheta_e^+ p_e + \vartheta_e^- q_e)\right]$$
$$\text{s.t.}$$
$$(3.4) \quad \left[\sum_{v \in V} \theta(\tilde{w}(v)) \left(\min_{i=1,\ldots,p} d_{\tilde{\ell}}(v, m_i) - \min_{k \in S} d_{\tilde{\ell}}(v, v_k)\right)\right] \leq 0 \quad \forall S \in \mathcal{S},$$
$$0 \leq p_e \leq u_e^+, \quad 0 \leq q_e \leq u_e^- \quad \forall e \in E,$$
$$0 \leq p_v \leq u_v^+, \quad 0 \leq q_v \leq u_v^- \quad \forall v \in V.$$

Definition 3.1. Let $p = (p_e)_{e \in E}$ and $q = (q_v)_{v \in V}$ be the vectors that satisfies in (3.2) and (3.3). Then (p, q) is called expected solution of (3.4) if and only if $\forall S \in \mathcal{S}$

$$\sum_{v \in V} E\left[\theta(\tilde{w}(v))\right] \left(\min_{i=1,\ldots,p} d_{\tilde{\ell}}(v, m_i) - \min_{k \in S} d_{\tilde{\ell}}(v, v_k)\right) \leq 0.$$

Now, let (p, q) be a expected solution of (3.4). Define

$$f(p, q) = \sum_{v \in V}(\vartheta_v^+ p_v + \vartheta_v^- q_v) + \sum_{e \in E}(\vartheta_e^+ p_e + \vartheta_e^- q_e).$$

Definition 3.2. For a risk confidence level $\beta \in (0, 1]$, a expected solution (p^*, q^*) is called optimal solution with minimum $TVaR$ if

$$TVaR_\beta(f(p^*, q^*)) \leq TVaR_\beta(f(p, q)),$$

holds for any expected solution (p, q).

Therefore, we can find an optimal expected solution with minimum TVaR as follows:

Let $(\Psi_v^+)^{-1}$, $v \in V$, and $(\Psi_e^+)^{-1}$, $e \in E$ be the inverse uncertainty distributions of ϑ_v^+ and ϑ_e^+, respectively. Also let $(\Psi_v^-)^{-1}$, $v \in V$, and $(\Psi_e^-)^{-1}$, $e \in E$ be the inverse uncertainty distributions of ϑ_v^- and ϑ_e^-, respectively. Assume that Υ_v^{-1}, $v \in V$, is the inverse uncertainty distribution of θ_v. Then, for a risk confidence level $\beta \in (0,1]$, the optimal expected solution with minimum TVaR is the optimal solution of the following model:

$$
\begin{aligned}
\min \quad & \sum_{v \in V} \left[\left(\frac{1}{\beta} \int_0^\beta (\Psi_v^+)^{-1}(1-\gamma)d\gamma \right) p_v + \left(\frac{1}{\beta} \int_0^\beta (\Psi_v^-)^{-1}(1-\gamma)d\gamma \right) q_v \right] \\
& + \sum_{e \in E} \left[\left(\frac{1}{\beta} \int_0^\beta (\Psi_e^+)^{-1}(1-\gamma)d\gamma \right) p_e + \left(\frac{1}{\beta} \int_0^\beta (\Psi_e^-)^{-1}(1-\gamma)d\gamma \right) q_e \right]
\end{aligned}
$$

(3.5) $s.t.$

$$
\sum_{v \in V} \left(\int_0^1 \Upsilon_v^{-1}(\tilde{w}(v), \alpha)d\alpha \right) \left(\min_{i=1,\ldots,p} d_{\tilde{\ell}}(v, m_i) - \min_{k \in S} d_{\tilde{\ell}}(v, v_k) \right) \leq 0
$$
$$
\forall S \in \mathcal{S},
$$
$$
0 \leq p_e \leq u_e^+, \quad 0 \leq q_e \leq u_e^- \quad \forall e \in E,
$$
$$
0 \leq p_v \leq u_v^+, \quad 0 \leq q_v \leq u_v^- \quad \forall v \in V.
$$

The above model is a deterministic inverse p-median problem formulation with vertex weights

$$
\int_0^1 \Upsilon_v^{-1}(\tilde{w}(v), \alpha)d\alpha, \ \ \forall v \in V,
$$

vertex weight modification costs

$$
\frac{1}{\beta} \int_0^\beta (\Psi_v^+)^{-1}(1-\gamma)d\gamma, \ \frac{1}{\beta} \int_0^\beta (\Psi_v^-)^{-1}(1-\gamma)d\gamma,
$$

and edge length modification costs

$$
\frac{1}{\beta} \int_0^\beta (\Psi_v^+)^{-1}(1-\gamma)d\gamma, \ \frac{1}{\beta} \int_0^\beta (\Psi_v^-)^{-1}(1-\gamma)d\gamma.
$$

Baroughi et al. in [3] showed that IpMLP on general networks is NP-hard. Thus we immediately conclude the following proposition.

Proposition 3.1. *UIpMLP with TVaR criterion on general networks is NP-hard.*

The above proposition implies that it is not possible to present exact polynomial time methods to solve UIpMLP on general networks. Therefore, we propose an efficient hybrid PSO algorithm for approximating the optimal solution of UIpMP on networks.

3.2.   Hybrid PSO algorithm

Kennedy and Eberhart in 1995 [24] developed the PSO algorithm as a nature-inspired evolutionary computation algorithm. Consider the following model

$$\min \quad f(x)$$
$$s.t. \quad x \in X,$$

where X is the restricted region. In PSO algorithm, a potential solution is presented as a particle $x_j \in X$ and a direction $v_j \in \mathbb{R}$ in which the particle will move. A swarm of particles is defined as a set $\{x_1, x_2, \ldots, x_N\}$, in which N is number of particles. Each particle x_j retains a record of the position of its previous best performance in a vector called $P_{best,j}$. The particle with best performance r in the population has been maintained in a vector G_{best}. An iteration involves evaluating of each particle x_j, then randomly setting of v_j in the direction of particle x_j's best previous position $P_{best,j}$ and the best previous position G_{best} of any particle in the population.

Since in UIpMLP the aim is to modify the vertex weights and edge lengths with respect to modification bounds. Thus, we consider a particle of the problem as $x_j = (x_{1,j}, x_{2,j}, \ldots, x_{2m+2n,j})$ where

(3.6)
$$(x_{1,j}, x_{2,j}, \ldots, x_{m,j}) = (p_e)_{e \in E},$$
$$(x_{m+1,j}, x_{m+2,j}, \ldots, x_{2m,j}) = (q_e)_{e \in E},$$
$$(x_{2m+1,j}, x_{2m+2,j}, \ldots, x_{2m+n,j}) = (p_v)_{v \in V},$$
$$(x_{2m+n+1,j}, x_{2m+n+2,j}, \ldots, x_{2m+2n,j}) = (q_v)_{v \in V}.$$

Therefore, x_j represents the decision vector of UIpMLP that used in PSO. In addition, according to the orthogonality condition

- if $q_e > p_e$, then $q_e = q_e - p_e$, $p_e = 0$,

- if $q_e < p_e$, then $p_e = p_e - q_e$, $q_e = 0$,

- if $q_v > p_v$, then $q_v = q_v - p_v$, $p_v = 0$,

- if $q_v < p_v$, then $p_v = p_v - q_v$, $q_v = 0$.

For checking the feasibility of particle x_j, we calculate the expected value of constraints by using the following uncertain simulation algorithm [33]. Let $S \in \mathcal{S}$.

Algorithm 1 (Expected value simulation)

1. Set $E = 0$.

2. For $k = 1, \ldots, 99$ do

   compute

$$E_k = 0.01 \sum_{v \in V} \left(\Upsilon_v^{-1}(\tilde{w}(v), 0.0k) \right) \left(\min_{i=1,\ldots,p} d_{\tilde{\ell}}(v, m_i) - \min_{k \in S} d_{\tilde{\ell}}(v, v_k) \right),$$

and $E := E + E_k$.

3. Report E.

Therefore, if the particle $x_j = (x_{1,j}, x_{2,j}, \ldots, x_{2m+2n,j})$ is defined as (3.6) and for each $S \in \mathcal{S}$, $E \leq 0$, then x_j is feasibile.

Based on Theorem 2.2, we present the following uncertain simulation procedure for computing TVaR of objective function for each feasible particle x_j and given $\beta \in (0, 1]$.

Algorithm 2 (TVaR simulation)

1. Set $T_\beta = 0$.

2. For $j = 1, \ldots, M$ do

    compute

$$T_\beta^j = \sum_{v \in V} \left[\left((\Psi_v^+)^{-1}(1 - \frac{j}{M}\beta) \right) p_v + \left((\Psi_v^-)^{-1}(1 - \frac{j}{M}\beta) \right) q_v \right]$$

$$+ \sum_{e \in E} \left[((\Psi_e^+)^{-1}(1 - \frac{j}{M}\beta))p_e + \left((\Psi_e^-)^{-1}(1 - \frac{j}{M}\beta) \right) q_e \right],$$

    and $T_\beta = T_\beta + \frac{j}{M}\beta T_\beta^j$.

3. Compute $TVaR_\beta = \frac{1}{\beta}T_\beta$.

4. Report $TVaR_\beta$.

To solve the model (3.5) with hybrid PSO algorithm, we first randomly generate the particle x_j by checking the feasibility of it using expected value simulation. Repeat this process N times. We get N initial feasible particles $x_1, x_2, \ldots, x_N$. Then, we assume that the fitness of each x_j is the minus of TVaR, i.e.,

$$Fit(x_j) = -TVaR_\beta(x_j).$$

Thus, the particle with higher fitness has smaller objective value. The fitness of each particle is obtained by using TVaR simulation.

In the process of updating $(i+1)$th iteration, we first denote $P_{best,j}(i)$ for each particle x_j and $G_{best}(i)$, then we obtain the new directs and the positions of the particles by using the following two equations:

(3.7) $v_j(i+1) = v_j(i) + C_1 r_1 [P_{best,j}(i) - x_j(i)] + C_2 r_2 [G_{best}(i) - x_j(i)],$

(3.8)
$$x_j(i+1) = x_j(i) + v_j(i+1),$$

where, $P_{best,j}(i) = x_j(i)$ if

$$Fit(x_j(i)) \geq Fit(x_j(i-1))$$

and $P_{best,j}(i) = P_{best,j}(i-1)$ otherwise, and $G_{best}(i) = P_{best,k}(i)$, with

$$k = argmin\{P_{best,j}(i) : j = 1, \ldots, N\}.$$

In addition r_1 and r_2 are uniformly distributed random numbers in the interval $[0, 1]$ and C_1 and C_2 are learning rates, to well adjust the convergence of the particles. The values of C_1 and C_2 are usually assumed to be 2.

If the updated x_j is feasible, then we consider it as a new particle of the next generation. Otherwise, as long as a feasible new particle is found, we re-update (3.7) and (3.8).

We obtain a new generation of particles by repeating the above process N times.

If MaxIt indicate the number of generations of the PSO algorithm, then based on all the explanations above, we summarize the hybrid PSO algorithm for solving the model (3.5) as follows.

Algorithm 3 (Hybrid PSO algorithm)

1. Initialize the feasibile particles $x_1, \ldots, x_N$ (use expected value simulation).

2. Compute the fitness for all particles by using TVaR simulation, and evaluate each particle according to it.

3. Update all the particles by using equations (3.7) and (3.8).

4. As long as a new feasible population is found, re-update (3.7) and (3.8).

5. Repeat Steps 2 to 4 for MaxIt times.

6. Return G_{best} as the optimal solution of the model (3.5), and

$$TVaR_\beta(G_{best}) = -Fit(G_{best})$$

as the corresponding optimal value.

3.3.   An illustrative example

In this subsection, we give a numerical example to illustrate the hybrid PSO algorithm. The result of the numerical experiment is obtained on a PC with processor Intel(R) Core(TM) i3 CPU 2.27GHZ and 4GB of RAM under windows 7.

We apply the hybrid PSO algorithm for solving UIpMLP with TVaR criteria at a risk confidence level of $\beta = 0.8$ on the given network N in Figure 3.1. Let the

cost coefficients be linear uncertain variables (see Table 3.3). Also let the vertex weights θ be the linear uncertain variables with respect to, $\tilde{w}(v)$, i.e.,

$$\theta = \theta(\tilde{w}(v)) = \mathcal{L}(\tilde{w}(v) - 10, \tilde{w}(v) + 10).$$

The input data of the network are given in Tables 3.1 and 3.3.

Note that if $\theta = \mathcal{L}(a, b)$ is the linear uncertain variable, then for a risk confidence level $\beta \in (0, 1]$

$$TVaR_\beta(\theta) = \frac{\beta}{2}(a - b) + b,$$

and

$$E[\theta] = \frac{(a + b)}{2}.$$

In the following, we show the computational results of the hybrid PSO algorithm's performance on an example of UI2MLP on the given network.

Note that the goal is to change $w(v)$ and ℓ_e with respect to modification bounds so that $\{v_2, v_3\}$ becomes a 2-median at minimum total cost under the new vertex weights and edge lengths.
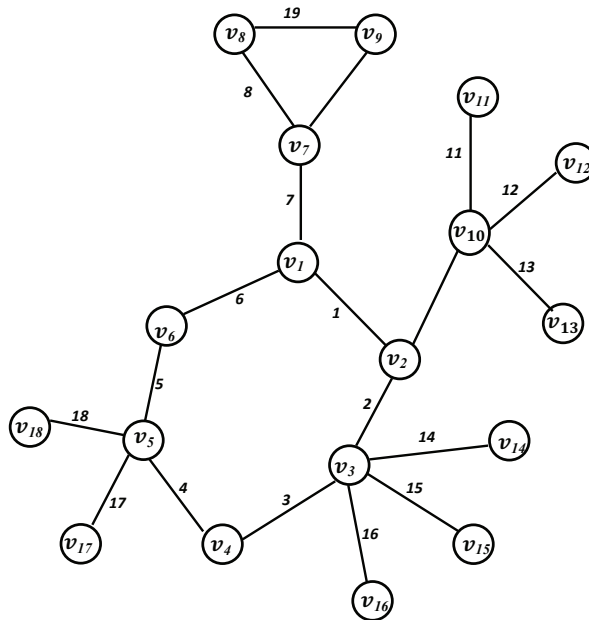

FIG. 3.1: Network N

Table 3.1: The input data for UI2MLP

ℓ_e	(14, 34, 25, 7, 22, 10, 8, 20, 12, 7, 10, 26, 12, 6, 10, 23, 31, 21, 22)
u_e^+	(5, 4, 5, 4, 7, 2, 5, 6, 3, 9, 2, 13, 1, 3, 5, 1, 7, 4, 1)
u_e^-	(10, 30, 15, 3, 17, 8, 4, 10, 8, 4, 5, 20, 6, 1, 8, 13, 2, 3, 13)
$w(v)$	(34, 18, 14, 13, 21, 11, 13, 20, 40, 22, 9, 17, 13, 6, 24, 14, 15, 12)
u_v^+	(20, 11, 4, 1, 2, 4, 7, 8, 15, 32, 13, 5, 15, 1, 2, 4, 1, 1)
u_v^-	(3, 2, 11, 9, 10, 8, 1, 1, 6, 7, 2, 2, 2, 4, 15, 10, 13, 11)

The hybrid PSO algorithm is run for the problem with 100, 200, 300 and 400 generations, respectively. Table 3.2 shows the best solutions of the problem.

Table 3.4, shows the best solutions of UI2MLP using hybrid PSO algorithm. Furthermore, the convergence of the objective values with population sizes 10,15,20,25 and $MaxIt = 100$ is shown in Figure 3.2. The convergence of the objective values with $N = 10$ and $MaxIt = 100, 200, 300, 400$ is given in Figure 3.3.

Table 3.2: The results of the performance of hybrid PSO algorithm

N, MaxIt	Objective value	N, MaxIt	Objective value
10, 100	-6880	10, 200	-6635
15, 100	- 8240	10, 300	-7609
20, 100	-8520	10, 400	-9150
25, 100	-10496		

Table 3.3: Uncertain cost coefficients

ϑ_e^+	($\mathcal{L}(8,10), \mathcal{L}(18,21), \mathcal{L}(19,21), \mathcal{L}(4,6), \mathcal{L}(3,4), \mathcal{L}(14,16), \mathcal{L}(28,30), \mathcal{L}(10,12),$ $\mathcal{L}(17,18), \mathcal{L}(6,8), \mathcal{L}(22,24), \mathcal{L}(7,9), \mathcal{L}(15,17), \mathcal{L}(18,21), \mathcal{L}(26,28),$ $\mathcal{L}(28,30), \mathcal{L}(16,18), \mathcal{L}(4,6), \mathcal{L}(4,6))$
ϑ_e^-	($\mathcal{L}(18,20), \mathcal{L}(14,15), \mathcal{L}(10,12), \mathcal{L}(24,26), \mathcal{L}(17,18), \mathcal{L}(15,17), \mathcal{L}(27,29), \mathcal{L}(8,10),$ $\mathcal{L}(22,24), \mathcal{L}(22,24), \mathcal{L}(11,13), \mathcal{L}(2,4), \mathcal{L}(1,3), \mathcal{L}(15,17), \mathcal{L}(24,25),$ $\mathcal{L}(28,30), \mathcal{L}(3,5), \mathcal{L}(17,19), \mathcal{L}(17,19))$
ϑ_v^+	($\mathcal{L}(24,26), \mathcal{L}(27,28), \mathcal{L}(3,5), \mathcal{L}(27,28), \mathcal{L}(19,20), \mathcal{L}(1,4), \mathcal{L}(8,10), \mathcal{L}(16,18),$ $\mathcal{L}(29,30), \mathcal{L}(29,30), \mathcal{L}(4,6), \mathcal{L}(30,31), \mathcal{L}(29,30), \mathcal{L}(14,16), \mathcal{L}(24,26),$ $\mathcal{L}(4,6), \mathcal{L}(12,13), \mathcal{L}(27,28))$
ϑ_v^-	($\mathcal{L}(19,21), \mathcal{L}(22,24), \mathcal{L}(12,13), \mathcal{L}(19,21), \mathcal{L}(5,6), \mathcal{L}(20,22), \mathcal{L}(1,2), \mathcal{L}(7,10),$ $\mathcal{L}(2,3), \mathcal{L}(3,4), \mathcal{L}(24,26), \mathcal{L}(20,21), \mathcal{L}(19,20), \mathcal{L}(29,30), \mathcal{L}(2,3),$ $\mathcal{L}(13,15), \mathcal{L}(12,13), \mathcal{L}(22,24))$

Table 3.4: The obtained G_{best} for UI2MLP by using hybrid PSO algorithm

| N, MaxIt	G_{best}
10, 100	(3.92, 0.90, 4.88, 0, 4.84, 0, 0, 5.82, 1.25, 0, 0, 0.33, 0, 0, 0.99, 0.94, 0, 0.82, 0.69, 0, 0, 0, 2.75, 0, 2.50, 3.22, 0, 0, 2.80, 1.08, 0, 4.60, 0.23, 0, 0, 1.46, 0, 0, 0, 9.51, 3.40, 0.32, 1.38, 0, 0, 0, 0, 0, 0, 4.94, 0, 0, 1.54, 3.26, 0.61, 0, 0.22, 0, 0, 0, 0, 6.31, 0.34, 0.72, 4.28, 5.64, 0.01, 0, 0.34, 2.56, 0, 0, 0, 10.73)
15, 100	(0.24, 2.85, 4.07, 0, 4.07, 1.29, 1.95, 4.12, 0, 8.92, 0, 8.38, 0.97, 0.30, 3.2, 0, 3.42, 3.49, 0.08, 0, 0, 0, 2.12, 0, 0, 0, 0, 6.16, 0, 2.68, 0, 0, 0, 0, 7.84, 0, 0, 0, 0.50, 0, 1.95, 0.30, 0.94, 3.70, 6.81, 0, 0, 7.51, 5.21, 1.95, 0, 0, 0, 0, 0, 0, 0, 0.94, 0, 0, 0, 0, 0, 0.31, 0.48, 0, 0, 0, 0.8, 1.08, 4.82, 8.40, 5.80, 10.03)
20, 100	(3.43, 3.91, 1.20, 0, 4.02, 0, 0, 4, 2.57, 0, 1.48, 11.32, 0, 0, 3.03, 0, 0, 0, 0, 0, 0, 1.67, 0, 3.10, 1.55, 0, 0, 3.70, 0, 0, 0.86, 0.96, 0, 7.83, 1.09, 1.22, 2.76, 0.52, 0, 0.78, 0, 0.44, 1.79, 6.47, 0, 11.03, 0, 0, 0, 4.89, 0.02, 0, 0, 0.29, 0, 0, 0.11, 0, 4.76, 0, 0, 0, 0.10, 0, 5.28, 1.77, 0.68, 0, 0, 4.57, 5.31, 0, 10)
25, 100	(2.32, 2.36, 2.88, 0, 4.76, 0, 0.28, 0, 0.80, 0, 1.78, 0, 0.16, 0, 2.61, 0, 1.70, 0, 0.91, 0, 0, 0, 2.04, 0, 1.45, 0, 9.33, 0, 0.50, 0, 8.00, 0, 0.39, 0, 7.15, 0, 1.06, 0, 0, 8.81, 0.69, 0, 0, 0.93, 1.37, 4.04, 3.59, 26.46, 0, 0.82, 0, 0, 0, 0, 0.45, 0, 0.62, 0,0, 4.63, 4.86, 0, 0, 0, 0, 0, 1.35, 0, 0.12, 0.72, 12.62, 3.096, 0, 10.65)
10, 200	(3.25, 2.75, 3.77, 0, 5.40, 1.71, 2.35, 0, 2.47, 4.18, 0, 8.17, 0, 0, 4.96, 0, 0, 0, 0.56, 0, 0, 0, 1.39, 0, 0, 0, 9.88, 0, 0, 4.93, 0, 4.93, 0.29, 0, 3.01, 1.57, 1.16, 0, 0, 1.42, 0, 0.85, 0, 3.90, 2.67, 5.46, 0, 0, 0, 0.95, 0.58, 0.72, 0, 0, 0.53, 0, 2.24, 0, 4.77, 0, 4.97, 0, 0, 0, 5.12, 6.30, 1.55, 0, 0, 0, 9.99, 1.85, 0, 10.17)
10, 300	4.42, 0.91, 2.20, 0, 2.82, 0, 0, 0, 2.54, 0, 0, 1.42, 0, 0, 0, 0, 4.70, 0, 0.77, 0, 0, 0, 1.51, 0, 1.05, 1.30, 5.94, 0, 0.89, 2.83, 0, 2.91, 0.20, 1.74, 2.71, 0, 2.60, 0, 0, 3.66, 0, 0, 0, 0, 6.34, 6.11, 9.43, 0, 0, 1.10, 0, 0.20, 1.64, 0.36, 0, 0, 2.31, 0, 1.27, 2.86, 6.34, 5.40, 0, 0, 0, 4.33, 0.79, 0, 0.50, 0, 0, 0, 0.81, 10.66)
10, 400	(1.80, 2.94, 1.64, 0, 3.60, 0, 2.13, 4.58, 0.53, 0, 0.36, 11.20, 0.42, 1.08, 0, 0, 0, 2.98, 0, 0, 0, 0, 1.94, 0, 2.28, 0, 0, 0, 3.89, 0, 0, 0, 0, 7.56, 5.33, 1.80, 0, 9.03, 1.69, 8.02, 1.17, 0, 0, 3.97, 1.14, 7.81, 0, 1.61, 4.24, 1.71, 0, 0.71, 0.71, 0, 0.66, 0, 0, 0, 0, 8.40, 3.35, 0, 0, 0, 4.31, 0, 0, 0, 1.43, 0, 0, 4.72, 0, 10.97)
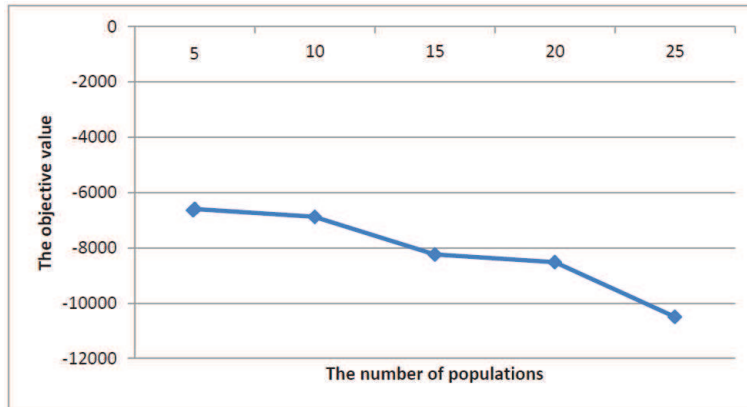
FIG. 3.2: The convergence of TVaR, MaxIt=100
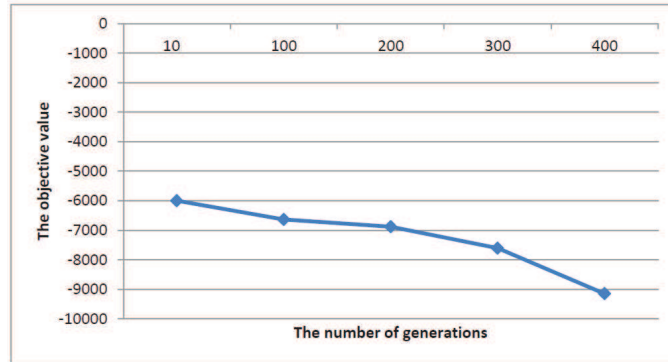
FIG. 3.3: The convergence of TVaR, N=10

4.   Conclusion

In this paper, we investigated IpMLP with variable edge lengths and variable vertex weights on a network in which the vertex weights and modification costs are the independent uncertain variables. We proposed a model for UIpMLP with TVaR objective and expected value constraints and showed that it is NP-hard. Thus, we presented a hybrid PSO algorithm for approximating the optimal solutions, which it contains expected value simulation and TVaR simulation. Finally, by computational experiments, the efficiency of the algorithm is illustrated.

REFERENCES

1. O. ALP, E. ERKUT and Z. DREZNER: *An efficient genetic algorithm for the p-median problem.* Ann. Oper. Res. **10** (2003), 1387-1395.

2. B. ALIZADEH and S. BAKHTEH: *A modified firefly algorithm for general inverse p-median location problems under different distance norms.* Opsearch. **54** (2017), 618-636.

3. F. BAROUGHI, R. E. BURKARD and E. GASSNER: *Inverse p-median problems with variable edge lengths.* Math. Meth. Oper. Res. **73** (2011), 263-280.

4. M. BASHIRI, M. MIRZAEI and M. RANDALL: *Modeling fuzzy capacitated p-hub center problem and a genetic algorithm solution.* Appl. Math. Model. **37(5)** (2013), 3513-3525.

5. X. BAI and Y. LIU: *Minimum risk facility location-allocation problem with type-2 fuzzy variables.* sci. world j. **2014** (2014), 1-9.

6. O. BERMAN, N. SANAJIAN and J. WANG: *Location choice and risk attitude of a decision maker.* Omega **66** (2017), 170-181.

7. R. BENKOCZI and B. BHATTACHARYA: *A new template for solving p-median problems for trees in sub-quadratic time (extended abstract).* LNCS. **3669** (2005), 271-282.

8. R. E. BURKARD and J. KRARUP: *A linear algorithm for the pos/neg-weighted 1-median problem on a cactus.* Computing. **60** (1998), 193-215.

9. R. E. BURKARD, C. PLESCHIUTSCHNIG and J. ZHAN: *Inverse median problems.* Discrete Optim. **1** (2004), 23-39.

10. R. E. BURKARD, C. PLESCHIUTSCHNIG and J. ZHAN: *The inverse 1-median problem on a cycle.* Discrete Optim. **5** (2008), 242-253.

11. Z. DREZNER: *The planar two-center and two-median problems.* Transportation Sci. **18** (1984), 351-361.

12. S. EILON, C. D. T. WATSON-GANDY and N. CHRISTOFIDES: *Distribution management: mathematical modeling and practical analysis.* New York: Hafner (1971).

13. H. A. EISELT and V. MARIANOV: *Foundations of location analysis.* OR/MS, NewYork: Springer (2011).

14. Y. GAO : *Uncertain models for single facility location problems on networks.* Appl. Math. Model. **36** (2012), 2592-2599.

15. Y. GAO and Z. QIN: *A chance constrained programming approach for uncertain p-hub center location problem.* Comput. Ind. Eng. **102** (2016), 10-20.

16. M. GALAVII: *The inverse 1-median problem on a tree and on a path.* Electron. Notes Discret. Math. **36** (2010), 1241-1248.

17. A. J. GOLDMAN: *Optimal center location in simple networks.* Transportation Sci. **5** (1971), 212-221.

18. X. GUAN and B. ZHANG: *Inverse 1-median problem on trees under weighted Hamming distance.* J Global Optim. **54** (2012), 75-82.

19. X. GUAN and B. ZHANG: *Inverse 1-median problem on trees under weighted l_∞ norm.* LNCS. **6124** (2010), 150-160.

20. S. L. HAKIMI: *Optimum locations of switching centers and the absolute centers and medians of a graph.* Oper. Res. **12** (1964), 450-459.

21. S. L. HAKIMI: *Optimum distribution of switching centers in a communication network and some related graph theoretic problems.* Oper. Res. **13** (1965), 462-475.

22. X. HUANG and D. HAO: *Modelling uncapacitated facility location problem with uncertain customers' positions.* J. Intell. Fuzzy Syst. **28** (2015), 2569-2577.

23. J. HATZL: *2-balanced flows and the inverse 1-median problem in the Chebyshev space.* Discrete Optim. **9** (2012), 137-148.

24. J. KENNEDY and R. C. EBERHART: *Particle swarm optimization.* ICNN'97. **4** (1995), 1942-1948.

25. B. LIU: *Uncertainty theory.* 2nd ed., Springer-Verlag, Berlin (2007).

26. B. LIU: *Uncertain risk analysis and uncertain reliability analysis.* J. Uncertain Syst. **4 (3)** (2010), 163-170.

27. X. LIU: *Uncertain programming model for location problem of multi-product logistics distribution centers.* Appl. Math. Sci. **9** (2015), 6543-6558.

28. R. F. LOVE: *Facilities location: models and methods.* Oper. Res. Ser. (1988).

29. I. MIRZAPOLIS ADEH, F. BAROUGHI and B. ALIZADEH: *A modified particle swarm optimization algorithm for general inverse ordered p-median location problem on network.* Facta Universitatis, Ser. Math. Inform. **32** (2017), 447-468.

30. K. T. NGUYEN: *Inverse 1-median problem on block graphs with variable vertex weights.* j. optimiz. theory appl. **168** (2016), 944-957.

31. K. T. NGUYEN and N. T. L. CHI: *A model for the inverse 1-median problem on trees under uncertain costs.* Opusc. Math. **36** (2016), 513-523.

32. J. PENG: *Risk metrics of loss function for uncertain system.* Fuzzy Optim. Decis. Ma. **12** (2013), 53-64.

33. J. PENG, B. ZHANG and S. LI: *Towards uncertain network optimization.* J. Uncertain. Anal. Appl. **3** (2015), Article 4.

34. Z. QIN and Y. GAO: *Uncapacitated p-hub location problem with fixed costs and uncertain flows.* J. Intell. Manuf. **28** (2017), 705-716.

35. R. RAHMANIANI, M. SAIDI-MEHRABAD and A. GHADERI: *An efficient hybrid solution algorithm for the capacitated facility location-allocation problem under uncertainty.* JOIE. DOI: 10.22094/joie.2018.538339 (2018).

36. A. R. SEPASIAN and F. RAHBARNIA: *An $O(n \log n)$ algorithm for the inverse 1-median problem on trees with variable vertex weights and edge reductions.* Optimization. **64** (2015), 595-602.

37. A. SCHOBEL and D. SCHOLZ: *The big cube small cube solution method for multi-dimensional facility location problems.* Comput. Oper. Res. **37** (2010), 115-122.

38. H. D. SHERALI and C. H. TUNCBILEK: *A squared Euclidean distance location-allocation problem.* NRL. **39** (1992), 447-469.

39. A. TAMIR: *An $O(pn^2)$ algorithm for the p-median and related problems on tree graphs.* Oper. Res. Lett. **19** (1996), 59-64.

40. K. E. WANG and Q. YANG: *Hierarchical facility location for the reverse logistics network design under uncertainty.* J. Uncertain Syst. **8** (2014), 255-270.

41. SH. WANG, J. WATADA and W. PEDRYCZ: *Value at risk based two-stage fuzzy facility location problems.* TII. **5** (2009), 465-482.

42. M. R. WAGNER, J. BHADURY and S. PENG: *Risk management in uncapacitated facility location models with random demands.* Comput. Oper. Res. **36** (2009), 1002-1011.

43. M. WEN, Z. QIN, R. KANG and Y. YANG: *The capacitated facility location-allocation problem under uncertain environment.* J. Intell. Fuzzy Syst. **29** (2015), 2217-2226.

44. K. YANG, Y. LIU and G. YANG: *Optimizing fuzzy p-hub center problem with generalizedvalue at risk criterion.* Appl. Math. Model. **38** (2014), 3987-4005.

45. K. YANG, Y. LIU and G. YANG: *An improved hybrid particle swarm optimization algorithm for fuzzy p-hub center problem.* Comput. Ind. Eng. **64** (2013a), 133-142.

46. B. ZHANG, J. PENG and S. LI: *Covering location problem of emergency service facilities in an uncertain environment.* Appl. Math. Model. **51** (2017), 429-447.

Akram Soltanpour
Faculty of Basic Sciences
Department of Applied Mathematics
Sahand University of Technology, Tabriz, Iran

Fahimeh Baroughi
Faculty of Basic Sciences
Department of Applied Mathematics
Sahand University of Technology, Tabriz, Iran
baroughi@sut.ac.ir
(Corresponding author)

Behrooz Alizadeh
Faculty of Basic Sciences
Department of Applied Mathematics
Sahand University of Technology, Tabriz, Iran