Research Article

Metaheuristic Algorithm for Solving Biobjective Possibility Planning Model of Location-Allocation in Disaster Relief Logistics

Farnaz Barzinpour, Mohsen Saffarian, Ahmad Makoui, and Ebrahim Teimoury

Department of Industrial Engineering, Iran University of Science and Technology, Tehran 1684613114, Iran

Correspondence should be addressed to Mohsen Saffarian; mo_saffarian@iust.ac.ir

Thousands of victims and millions of affected people are hurt by natural disasters every year. Therefore, it is essential to prepare proper response programs that consider early activities of disaster management. In this paper, a multiobjective model for distribution centers which are located and allocated periodically to the damaged areas in order to distribute relief commodities is offered. The main objectives of this model are minimizing the total costs and maximizing the least rate of the satisfaction in the sense of being fair while distributing the items. The model simultaneously determines the location of relief distribution centers and the allocation of affected areas to relief distribution centers. Furthermore, an efficient solution approach based on genetic algorithm has been developed in order to solve the proposed mathematical model. The results of genetic algorithm are compared with the results provided by simulated annealing algorithm and LINGO software. The computational results show that the proposed genetic algorithm provides relatively good solutions in a reasonable time.

1. Introduction

It is inevitable to have an integrated scientific system for crisis' logistics management with clearly defined functions and duties. Optimizing logistics and relief chain can affect issues such as relief logistics management [1]. Logistics can create greater coordination for delivering relief commodities while accelerating the delivery speed and reducing the response time [2, 3]. One of the logistics subproblems with strategic roles in a critical situation is to locate the depots and their supply savings [2]. In the depot locating, factors such as cost, response time, and justice in distribution have to be considered in preparedness phase [4]. Although researches in facilities location problems have been widely done both theoretically and practically, the problems have not been used in humanitarian logistics [2].

The application of operation research in disaster management programs has been one of the main issues in recent decades. Supply chain and logistics management are recently being used as analytical tools and techniques to provide efficient and effective relief to people who need help in devastated areas with optimized functions and activities. Thus, emergency logistics management has been emerged as a worldwide noticeable theme as disasters may occur anytime around the world with enormous consequences. The aim of response to disasters in humanitarian relief chains is to provide a quick relief to the affected areas in order to minimize the death and pain of people. Proper design and operation of a relief chain are an essential element to achieve an effective and efficient response, but, only in recent years, humanitarian organizations have paid attention to its importance [5].

In urban areas, earthquakes can cause serious damage to inhabitance and force people to leave their homes. Therefore, at the preparation planning phase of disaster management, municipal authorities should consider emergency shelters, proper equipment, and supplies for the affected people in order to reduce the number of casualties and people's sufferings and bring relief to survivors [5].

The number and location of the facilities and distribution centers decrease the response time and the operation costs in relief chain. Therefore, Location-Allocation is an important issue in this process. Some of the applications of these locations in the relief logistics problem are listed as follows:

(i) the location of critical management centers in the city or province and the allocation of emergency commodities such as water, food packages, and tents from these centers to damaged areas,

(ii) Red Crescent depots' locating and these depots allocation to damaged cities and provinces in the situation of facing natural crisis or even war,

(iii) locating ambulance centers inside and between the cities and provinces.

Actually in crisis management, we deal with locating process already in preparedness and response phase. Locating process and facility capacity determination are the two main topics in the disaster management, and a few researches have been focusing on this subject [6].

Barbarosoğlu and Arda [7] modeled the transportation planning in earthquake response by a scenario-based two stage stochastic programming over a multicommodity and multimodal network flow with uncertainty in demand, supply, and route capacities. The inclusion of uncertainties is an important advance in the analysis, but the focus is still postdisaster response. They did not consider the location of facilities or the inventory decisions in their model. Akkihal [8] has proposed a model for crisis management-center location in order to manage relief items. Tzeng et al. [9] have presented a certain multicriteria model in order to distribute relief commodities to damaged areas considering costs, response time, and damaged people satisfaction and solved it by fuzzy multiobjective programming. Jia et al. [10] presented an incapacitated facility location model to locate emergency service facilities in the event of a large-scale emergency.

Balcik and Beamon [2] considered the facility location problem for humanitarian relief chains in order to respond to quick onset disasters. Their proposed model aimed to locate and determine the situation and the number of distribution centers in relief network and the number of stored commodities in order to meet the demand of affected people.

Beraldi and Bruni [11] also offered a probable model to have a relief facility to optimize locating in an uncertain environment. Mete and Zabinsky [12] introduced a stochastic optimization model for disaster preparedness and response under demand/cost uncertainty in order to assist deciding on the location and allocation of medical supplies which are used during emergencies. They also offered a mixed-integer programming transportation model that is potentially useful in routing decisions during the response phase. Rawls and Turnquist [13] developed a two-stage, stochastic, mixed-integer program that determined the locations and quantities of various types of emergency commodities; their model also contained the transportation network availability following a disaster under demand/cost uncertainty.

Moreover, Jabal-Ameli et al. [14] presented a location and distribution model and solved it by fuzzy methods. Duran et al. [15] studied inventory and location model and offered a mathematical model which investigates the effect of location on the average response time. Zeng and Wu [16] considered location and routing in relief situation in two steps. It means that they first solved the location problem and directly solved the routing problem in a way to minimize total costs and then they utilized a two-step heuristic algorithm to solve it.

Bozorgi-Amiri et al. [17] considered an uncertain model for relief logistics. They suggested that some prepositioned relief items might be destroyed after disasters based on this idea; they developed a robust multiobjective approach to their uncertain location problem and solved it for a real case study in Iran. Bozorgi-Amiri et al. [18] investigated uncertainty in many parameters of a relief operation like demand, supply, and operational costs associated with it. Location of relief centers and allocation of affected areas to these centers can be determined in situation described in their model. Yazdian and Shahanaghi [19] presented a multiobjective possibility programming approach for locating distribution centers and allocating customers' demands in supply chains.

In 2011, Barzinpour and Esmaeili [5] developed a multiobjective MILP model for a two-echelon relief chain in order to maximize coverage of urban populations and minimize logistics costs. Tancrez et al. [20] considered inventory-location problem for three-level supply chain in which distribution decisions of distribution centers, allocation, and number of the transported commodities were made altogether.

In 2012, Schmid [21] offered a probability model for relocation problem and dynamic distribution and solved it by dynamic programming. Chrétienne et al. [22] suggested location and dispatching problem which included minimizing setup and availability costs and then solved it through branch and cut algorithm. Toro-Diaz et al. [23] offered a mathematical model for location and distribution problem and solved it through genetic algorithm. In 2012, Guillermo Cabrera et al. [24] presented a hybrid approach using an artificial bee algorithm with mixed-integer programming applied to a large-scale capacitated facility location problem. Murali et al. [25] considered a facility location problem to determine the points in a city where medicine should be handed out to the population. They consider locating capacitated facilities in order to maximize the coverage, considering a distance-dependent coverage function.

In 2013, Xi et al. [26] developed a modified p-median problem model that accounts for rescue time limitations. A variable neighborhood search- (VNS-) based algorithm is developed for the model considered. Gharegozloo Hamedani et al. [27] studied a multiobjective location problem in a three-level supply chain network under uncertain environment, considering inventory decisions. The proposed model presents a robust optimization model, which specifies locations of distribution centers to be opened, inventory control parameters, and allocation of supply chain components, concurrently.

Abounacer et al. [28] proposed a three-objective location-transportation problem for disaster response. The location problem aims at determining the number, the position, and the mission of required humanitarian aid distribution centers (HADC) within the disaster region. Three conflicting objectives are considered. The first objective minimizes the total transportation duration of needed products from the distribution centers to the demand points. The second objective minimizes the number of agents (first-aiders) required to open and operate the selected distribution centers. The third objective minimizes the noncovered demand for all demand points within the affected area. They proposed an epsilon-constraint method for this problem. Esmaeili and Barzinpour [29] based on a real world case study for a municipal district in Tehran developed a multiobjective mathematical model for the location-distribution problem.

Most of the researches in location and distribution of relief logistics are dependent on preparedness phase on the basis of predicting programming conditions. Now, the question is whether the same criterion in preparedness phase can be operation criterion or, based on the new conditions and available information, new programs should be suggested. Usually after any casualty, in the first 72 hours, based on the available commodities in the distribution centers, services are offered. Therefore, it would be possible to program for damaged areas during that time based on the received information. In this paper, the goal is to present a model to locate distribution centers periodically and to consider good distribution manner toward damaged areas.

This paper focuses on the logistics aspect of the response phase and more precisely on two important related problems: location and allocation. The location problem aims at designing a network for distributing humanitarian aids (e.g., water, food, medical commodities, and survival equipment). It mainly consists of determining the position and the mission of required humanitarian aid distribution centers (HADC) within the disaster region. The allocation problem deals with the distribution of humanitarian aid from HADCs to demand points. When both problems are solved simultaneously, we discuss a location-allocation problem. Based on this particular context, this paper has provided a biobjective location-allocation problem. The first objective is to minimize the total costs. The second objective is to maximize the covered demand for all demand points.

We propose a genetic algorithm and a simulated annealing algorithm to biobjective location–allocation problem addressed.

2. Problem Definition and Model Assumptions

A two-level system including distributers and damaged areas is considered. There are regions as candidate regions to establish distribution centers in this system. And also the number of commodities in distribution centers and demands of damaged areas in different periods are available. Our goal is to determine active distribution centers and their distribution manner in different periods. Location and allocation have to be done so that total costs become minimized and commodities have to be distributed in a fair manner.

2.1. Model Assumptions

(i) The inventories of distribution centers in different periods are available in the form of triangular fuzzy numbers.

(ii) The damaged area requests in different periods are available in the form of triangular fuzzy numbers.

(iii) The maximum number of distribution centers that can be active in each period is determined.

(iv) Some types of relief commodities are available.

(v) It is possible to transfer inventories from one period to the next one.

(vi) Unsatisfied request does not transfer from one period to the next one.

2.2. Model Indexes

I_t: set of distribution centers in period t,

J_t: set of damaged areas in period t,

C: set of relief commodities,

i: indexes related to distribution centers,

j: indexes related to damaged area,

c: indexes related to relief commodities.

2.3. Model Parameters

$\widetilde{q}_{it}$: setup cost of distribution center i in period t that is a triangular fuzzy number,

p_t: the maximum number of distribution centers that can be active in period t,

$\widetilde{cc}_{cijt}$: transporting cost of c commodity unit between distribution center i and damaged area j in period t that is a triangular fuzzy number,

$\widetilde{cc1}_{cijt}$: transporting cost of c commodity unit between distribution center i and distribution center j in period t that is a triangular fuzzy number,

$\widetilde{inv}_{cit}$: the number of commodities c in distribution center i in period t that is a triangular fuzzy number,

$\widetilde{d}_{cjt}$: the number demand commodities c in damaged area j in period t that is a triangular fuzzy number,

β_c: penalty costs for the shortage of good c,

Mbig: a big number.

2.4. Decision Variables

x_{cijt}: number of commodities c that can be delivered in period t from distribution center i to damaged area j,

$x1_{cijt}$: binary variable, it is 1 if commodity c transfers from distribution center i to distribution center j in period t; otherwise, it is 0,

y_{it}: binary variable, it is 1 if distribution center i becomes active in period t; otherwise, it is 0,

inv1_{cit}: amount of transferred supply of commodity c in period t to period $t + 1$ in distribution center i.

2.5. The Proposed Mathematical Model. Consider

$$\min f_1 = \sum_{c \in C} \sum_{t \in T} \sum_{i \in I_t} \sum_{j \in J_t} \widetilde{cc}_{cijt} x_{cijt}$$

$$+ \sum_{t \in T} \sum_{i \in I_t} \max \left(0, y_{it} - \sum_{k=1}^{t-1} y_{ik} \right) \widetilde{q}_{it}$$

$$+ \sum_{t \in T} \sum_{c \in C} \sum_{j \in J_t} \beta_c \left(\widetilde{d}_{cjt} - \sum_{i \in I_t} x_{cijt} \right) \tag{1}$$

$$+ \sum_{c \in C} \sum_{t \in T} \sum_{i \in I_t} \sum_{j \in I_t} \widetilde{cc1}_{cijt} \widetilde{\text{inv}}_{cit} x1_{cijt},$$

$$\max f_2 = \sum_{t \in T} \sum_{c \in C} \min_{j \in J_t} \frac{\sum_{i \in I_t} x_{cijt}}{\widetilde{d}_{cjt}}, \tag{2}$$

$$\sum_{i \in I_t} y_{it} \le p_t, \quad \forall t \in T, \tag{3}$$

$$\sum_{c \in C} \sum_{j \in J_t} x_{cijt} + \sum_{c \in C} \sum_{j \in I_t} x1_{cjit} \le M\text{big}\, y_{it}, \quad \forall t \in T, \ i \in I_t, \tag{4}$$

$$\text{inv1}_{cit} = \widetilde{\text{inv}}_{cit} + \text{inv1}_{ci(t-1)} y_{i(t-1)}$$

$$+ \sum_{j \in I_t} \widetilde{\text{inv}}_{cjt} x1_{cjit} - \sum_{j \in J_t} x_{cijt}, \quad \forall t \in T, \ c \in C, \ i \in I_t, \tag{5}$$

$$\sum_{i \in I_t} x_{cijt} \le \widetilde{d}_{cjt}, \quad \forall t \in T, \ c \in C, \ j \in J_t, \tag{6}$$

$$\sum_{c \in C} \sum_{j \in I_t} x1_{cijt} = 1, \quad \forall t \in T, \ i \in I_t \tag{7}$$

$$x_{cijt} \in \text{Integer}, \quad \forall t \in T, \ c \in C, \ i \in I_t, \ j \in J_t, \tag{8}$$

$$x1_{cijt} \in \{0, 1\}, \quad \forall t \in T, \ c \in C, \ i \in I_t, \ j \in I_t, \tag{9}$$

$$y_{it} \in \{0, 1\}, \quad \forall t \in T, \ i \in I_t, \tag{10}$$

$$\text{inv1}_{cit} \in \text{Integer}, \quad \forall t \in T, \ c \in C, \ i \in I_t. \tag{11}$$

There are two objective functions in the above model.

Objective 1. It aims at minimizing total transportation costs between distribution centers and damaged areas and also minimizing setup costs and penalty costs for the shortage of goods.

Objective 2. It aims at maximizing damaged area satisfaction through maximizing satisfied requests of damaged areas.

Constraint (3) indicates the number of distribution centers that can be active. Constraint (4) shows that it will be possible to send goods from one distribution center to a damaged area if that center becomes active in that period. Constraint (5) shows the way in which it will be possible to calculate transported inventories from one period to the next period. Constraint (6) indicates that the number of received commodities to one damaged area should not be more than its demand. Constraint (7) shows that it will be possible to send goods from one distribution center to another distribution center if that destination center becomes active in that period. Constraints (8)–(11) express the nature of decision variables used in the model.

2.6. Linearization of the Proposed Model.
The second objective function is nonlinear term. To make it linear, we act as follows:

$$\max f_2 = \sum_{t \in T} \sum_{c \in C} k_{tc},$$

$$k_{tc} \le \frac{\sum_{i \in I} x_{cijt}}{\widetilde{d}_{cjt}}, \quad \forall t \in T, \ c \in C, \ j \in J_t. \tag{12}$$

In constraint (5), $\text{inv1}_{ci(t-1)} y_{i(t-1)}$ is nonlinear. So, for linearization, it can be replaced with $\text{inv}y_{ci(t-1)}$ and three constraints are added to models as follows:

$$\text{inv}y_{ci(t-1)} \le \text{inv1}_{ci(t-1)}, \quad \forall t \in T, \ c \in C, \ i \in I_t,$$

$$\text{inv}y_{ci(t-1)} \le M\text{big}\, y_{i(t-1)}, \quad \forall t \in T, \ c \in C, \ i \in I_t,$$

$$\text{inv}y_{ci(t-1)} \ge \text{inv1}_{ci(t-1)} - M\text{big}\, (1 - y_{i(t-1)}), \tag{13}$$

$$\forall t \in T, \quad c \in C, \quad i \in I_t,$$

$$\text{inv}y_{cit} \in \text{Integer}, \quad \forall t \in T, \ c \in C, \ i \in I_t.$$

2.7. Multipurpose Linear Programming Approach.
Considering the following multiobjective model, using Zimmerman method, this problem can be changed into a single objective linear programming model using max-min operator [14, 30]:

$$\max \quad Z = [c_1 x, c_2 x, \ldots, c_l x]^T,$$

$$\min \quad W = [d_1 x, d_2 x, \ldots, d_r x]^T, \tag{14}$$

$$\text{s.t.} \quad Ax \le B; \quad x \ge 0.$$

Membership functions for objective functions are defined as follows:

$$\mu_k (z_k) = \frac{z_k (x) - z_k^{\text{nis}}}{z_k^{\text{pis}} - z_k^{\text{nis}}}, \quad k = \{1, 2, \ldots, l\},$$

$$\mu_s (w_s) = \frac{w_s^{\text{nis}} - w_s (x)}{w_s^{\text{nis}} - w_s^{\text{pis}}}, \quad s = \{1, 2, \ldots, r\}. \tag{15}$$

In the above equations, $w_s^{\text{pis}}, z_k^{\text{pis}}$ are positive ideal solutions and $w_s^{\text{nis}}, z_k^{\text{nis}}$ are negative ideal solutions. Now, by max-min operator and satisfied degree, the multiobjective linear

programming problem has been written as a single-objective programming problem

$$
\begin{aligned}
\max \quad & \lambda \\
\text{s.t.} \quad & \lambda \le \frac{z_k(x) - z_k^{\text{nis}}}{z_k^{\text{pis}} - z_k^{\text{nis}}}, \quad k = 1 \cdots l \\
& \lambda \le \frac{w_s^{\text{nis}} - w_s(x)}{w_s^{\text{nis}} - w_s^{\text{pis}}}, \quad s = 1 \cdots r \\
& Ax \le B; \quad x \ge 0.
\end{aligned}
\tag{16}
$$

2.8. Solving Proposed Model. The first objective function of the proposed model is a function with triangular fuzzy coefficient. Therefore, we can change it to 3 objective functions as follows:

$$
\begin{aligned}
\min \ f_{11} = & \sum_{c \in C} \sum_{t \in T} \sum_{i \in I_t} \sum_{j \in J_t} \left(cc^u_{cijt} - cc^m_{cijt} \right) x_{cijt} \\
& + \sum_{t \in T} \sum_{i \in I_t} \max \left(0, y_{it} - \sum_{k=1}^{t-1} y_{ik} \right) \left(q^u_{it} - q^m_{it} \right) \\
& + \sum_{t \in T} \sum_{c \in C} \sum_{j \in J_t} \beta_c \left[\left(d^u_{cjt} - d^m_{cjt} \right) - \sum_{i \in I_t} x_{cijt} \right] \\
& + \sum_{c \in C} \sum_{t \in T} \sum_{i \in I_t} \sum_{j \in I_t} \left(cc1^u_{cijt} - cc1^m_{cijt} \right) \\
& \qquad \times \left(\text{inv}^u_{cit} - \text{inv}^m_{cit} \right) x1_{cijt},
\end{aligned}
$$

$$
\begin{aligned}
\min \ f_{12} = & \sum_{c \in C} \sum_{t \in T} \sum_{i \in I_t} \sum_{j \in J_t} cc^m_{cijt} x_{cijt} \\
& + \sum_{t \in T} \sum_{i \in I_t} \max \left(0, y_{it} - \sum_{k=1}^{t-1} y_{ik} \right) q^m_{it} \\
& + \sum_{t \in T} \sum_{c \in C} \sum_{j \in J_t} \beta_c \left(d^m_{cjt} - \sum_{i \in I_t} x_{cijt} \right) \\
& + \sum_{c \in C} \sum_{t \in T} \sum_{i \in I_t} \sum_{j \in I_t} cc1^m_{cijt} \text{inv}^m_{cit} x1_{cijt},
\end{aligned}
\tag{17}
$$

$$
\begin{aligned}
\max \ f_{13} = & \sum_{c \in C} \sum_{t \in T} \sum_{i \in I_t} \sum_{j \in J_t} \left(cc^m_{cijt} - cc^l_{cijt} \right) x_{cijt} \\
& + \sum_{t \in T} \sum_{i \in I_t} \max \left(0, y_{it} - \sum_{k=1}^{t-1} y_{ik} \right) \left(q^m_{it} - q^l_{it} \right) \\
& + \sum_{t \in T} \sum_{c \in C} \sum_{j \in J_t} \beta_c \left[\left(d^m_{cjt} - d^l_{cjt} \right) - \sum_{i \in I_t} x_{cijt} \right] \\
& + \sum_{c \in C} \sum_{t \in T} \sum_{i \in I_t} \sum_{j \in I_t} \left(cc1^m_{cijt} - cc1^l_{cijt} \right) \\
& \qquad \times \left(\text{inv}^m_{cit} - \text{inv}^l_{cit} \right) x1_{cijt}.
\end{aligned}
$$

There are three fuzzy constraints in the presented model. Therefore, we can change it to 3 certain constraints as follows [10]:

$$
k_{tc} \le \frac{\sum_{i \in I_t} x_{cijt}}{\left(\left(d^u_{cjt} - d^l_{cjt} \right) + \left(d^m_{cjt} - d^l_{cjt} \right) \right) / 3 + d^l_{cjt}},
$$
$$
\forall t \in T, \quad c \in C, \quad j \in J_t,
$$

$$
\begin{aligned}
\text{inv1}_{cit} & \\
= & \frac{\left(\text{inv}^u_{cit} - \text{inv}^l_{cit} \right) + \left(\text{inv}^m_{cit} - \text{inv}^l_{cit} \right)}{3} \\
& + \text{inv}^l_{cit} + \text{inv1}_{ci(t-1)} y_{i(t-1)} \\
& + \sum_{j \in I_t} \left[\frac{\left(\text{inv}^u_{cjt} - \text{inv}^l_{cjt} \right) + \left(\text{inv}^m_{cjt} - \text{inv}^l_{cjt} \right)}{3} + \text{inv}^l_{cjt} \right] x1_{cjit} \\
& - \sum_{j \in J} x_{cijt}, \quad \forall t \in T, \ c \in C, \ i \in I_t,
\end{aligned}
$$

$$
\sum_{i \in I_t} x_{cijt} \le \frac{\left(d^u_{cjt} - d^l_{cjt} \right) + \left(d^m_{cjt} - d^l_{cjt} \right)}{3} + d^l_{cjt},
$$
$$
\forall t \in T, \quad c \in C, \quad j \in J_t.
\tag{18}
$$

Now, a certain linear multiobjective model can be achieved as follows:

MOLP:

$$
\begin{aligned}
\text{Min} \quad & Z = [f_{11}, f_{12}] \\
\text{Max} \quad & W = [f_{13}, f_2] \\
\text{S.t.} \quad & (3), (4), (7), (8), (9), (10), \\
& (11), (13), (18).
\end{aligned}
\tag{19}
$$

To solve this linear multiobjective model through Zimmermann fuzzy sets (15), membership functions are defined as follows for the objective functions of the problem [30]:

$$
\mu_{f_{11}} = \begin{cases} 1 & f_{11} \le f_{11}^{\text{pis}} \\ \dfrac{f_{11}^{\text{nis}} - f_{11}}{f_{11}^{\text{nis}} - f_{11}^{\text{pis}}} & f_{11}^{\text{pis}} \le f_{11} \le f_{11}^{\text{nis}} \\ 0 & f_{11} \ge f_{11}^{\text{nis}}, \end{cases}
$$
$$
\mu_{f_2} = \begin{cases} 1 & f_2 \ge f_2^{\text{pis}} \\ \dfrac{f_2 - f_2^{\text{nis}}}{f_2^{\text{pis}} - f_2^{\text{nis}}} & f_2^{\text{nis}} \le f_2 \le f_2^{\text{pis}} \\ 0 & f_2 \ge f_2^{\text{nis}}. \end{cases}
\tag{20}
$$

To calculate μ_{f12} and μ_{f13}, a procedure the same as that of μ_{f11} and μ_{f2} would be followed respectively. Now, we use

the following equations to obtain an ideal positive value (pis) and a negative value (nis):

$$
\begin{aligned}
f_{11}^{\text{pis}} &= \min f_{11}, & f_{11}^{\text{nis}} &= \max f_{11}, \\
f_{12}^{\text{pis}} &= \min f_{12}, & f_{12}^{\text{nis}} &= \max f_{12}, \\
f_{13}^{\text{pis}} &= \max f_{13}, & f_{13}^{\text{nis}} &= \min f_{13}, \\
f_{2}^{\text{pis}} &= \max f_{2}, & f_{2}^{\text{nis}} &= \min f_{2}.
\end{aligned}
\tag{21}
$$

Now, we change the multiobjective problem (19) to the single-objective problem (22) via (16). The model is defined as follows:

$$
\begin{aligned}
\max \quad & \lambda \\
\text{S.t.} \quad & \lambda \le \frac{f_{11}^{\text{nis}} - f_{11}}{f_{11}^{\text{nis}} - f_{11}^{\text{pis}}} \\
& \lambda \le \frac{f_{12}^{\text{nis}} - f_{12}}{f_{12}^{\text{nis}} - f_{12}^{\text{pis}}} \\
& \lambda \le \frac{f_{13}^{\text{pis}} - f_{13}}{f_{13}^{\text{pis}} - f_{13}^{\text{nis}}} \\
& \lambda \le \frac{f_{2}^{\text{pis}} - f_{2}}{f_{2}^{\text{pis}} - f_{2}^{\text{nis}}} \\
& (3), (4), (7), (8), (9), \\
& (10), (11), (13), (18).
\end{aligned}
\tag{22}
$$

3. Genetic Algorithm

In the last decade, we have seen a growing interest in biologically motivated approaches such as evolutionary strategies and genetic algorithms (GAs) being applied to many complex optimization problems. The processes occurring in the natural systems have inspired the development of these algorithms. Evolutionary strategies and genetic algorithms are constructed based on the observation of evolutionary processes in biological systems. Evolutionary processes such as adaptation, selection, reproduction, mutation, and competition are closely studied and translated into the form of computer simulations. Although these algorithms are a crude simplification of the natural processes, they have been successfully applied to many complex problems that were once intractable [31, 32].

GA is a stochastic search technique that explores the problem domain by maintaining a population of individuals, which represents a set of potential solutions in the search space. GA attempts to combine the good features found in each individual using a structured yet randomized information exchange in order to construct individuals who are better suited to their environment than the individuals that were created through the evolution of better individuals; it is anticipated that the desired solution will be found. The following steps are defined to design this algorithm.

3.1. Solution Representation. To show the chromosome, we used three matrices; the first matrix amount is zero or one and the rest of them are positive integer numbers.

The first matrix is called location matrix that is related to distribution centers. Its lines are related to a time period and its columns are related to distribution centers. Its dimension is $|T| \times \max_{t \in T} |I_t|$ in which lines are periods and columns are distribution centers. This matrix's elements are zero and one in a way that zero and one correspond to inactiveness and activeness of the distribution center, respectively,

$$
\text{Location} = \begin{bmatrix}
L_{11} L_{12} \cdots L_{1|I_1|} \\
L_{21} L_{22} \cdots L_{1|I_2|} \\
\vdots \\
L_{|T|1} L_{|T|2} \cdots L_{|T||I_{|T|}|}
\end{bmatrix}.
\tag{23}
$$

The second matrix is called Commodity_Allocated matrix. To each commodity, there is a combined matrix including $|T|$ submatrix with $|I_t| \times |J_t|$ dimension. Each element in these submatrices indicates the number of commodities which are sent to corresponding damaged areas from the distribution center

Commodity_Allocated_1

$$
= \left[\begin{bmatrix}
i_{11} i_{12} \cdots i_{1|J_1|} \\
i_{21} i_{22} \cdots i_{2|J_1|} \\
\vdots \\
i_{|I_1|1} i_{|I_1|2} \cdots i_{|I_1||J_1|}
\end{bmatrix}^{1} \right.
$$

$$
\left. \cdots \begin{bmatrix}
i_{11} i_{12} \cdots i_{1|J_{|T|}|} \\
i_{21} i_{22} \cdots i_{2|J_{|T|}|} \\
\vdots \\
i_{|I_{|T|}|1} i_{|I_{|T|}|2} \cdots i_{|I_{|T|}||J_{|T|}|}
\end{bmatrix}^{|T|} \right], \dots,
$$

Commodity_Allocated_$|c|$

$$
= \left[\begin{bmatrix}
i_{11} i_{12} \cdots i_{1|J_1|} \\
i_{21} i_{22} \cdots i_{2|J_1|} \\
\vdots \\
i_{|I_1|1} i_{|I_1|2} \cdots i_{|I_1||J_1|}
\end{bmatrix}^{1} \cdots \begin{bmatrix}
i_{11} i_{12} \cdots i_{1|J_{|T|}|} \\
i_{21} i_{22} \cdots i_{2|J_{|T|}|} \\
\vdots \\
i_{|I_{|T|}|1} i_{|I_{|T|}|2} \cdots i_{|I_{|T|}||J_{|T|}|}
\end{bmatrix}^{|T|} \right].
\tag{24}
$$

The third matrix is called inventory matrix that indicates the amount of transported inventory from one period to the next one in each distribution center

$$
\text{Inventory\_1} = \begin{bmatrix}
\text{inv}_{11} \text{inv}_{12} \cdots \text{inv}_{1|T|} \\
\text{inv}_{21} \text{inv}_{22} \cdots \text{inv}_{2|T|} \\
\vdots \\
\text{inv}_{|I_1|1} \text{inv}_{|I_2|2} \cdots \text{inv}_{|I_{|T|}||T|}
\end{bmatrix}_{\max_{t \in T} |I_t| \times |T|}, \dots,
$$

$$
\text{Inventory\_}|c| = \begin{bmatrix}
\text{inv}_{11} \text{inv}_{12} \dots \text{inv}_{1|T|} \\
\text{inv}_{21} \text{inv}_{22} \dots \text{inv}_{2|T|} \\
\vdots \\
\text{inv}_{|I_1|1} \text{inv}_{|I_2|2} \dots \text{inv}_{|I_{|T|}||T|}
\end{bmatrix}_{\max_{t \in T} |I_t| \times |T|}.
\tag{25}
$$

0	1	0	1	0	0	0	1
0	0	1	1	1	0	0	0
1	1	1	0	0	0	1	0
1	1	1	1	0	0	0	0

Location matrix of parent 1

(a)

0	0	0	1	1	0	1	0
0	0	1	1	0	0	0	1
1	0	1	0	1	1	0	0
0	1	1	1	1	0	0	0

Location matrix of parent 2

(b)

0	1	0	1	0	0	0	1
0	0	1	1	0	0	0	1
1	1	1	0	0	0	1	0
1	1	1	1	0	0	0	0

Location matrix of child 1

(c)

0	0	0	1	1	0	1	0
0	0	1	1	1	0	0	0
1	0	1	0	1	1	0	0
0	1	1	1	1	0	0	0

Location matrix of child 1

(d)
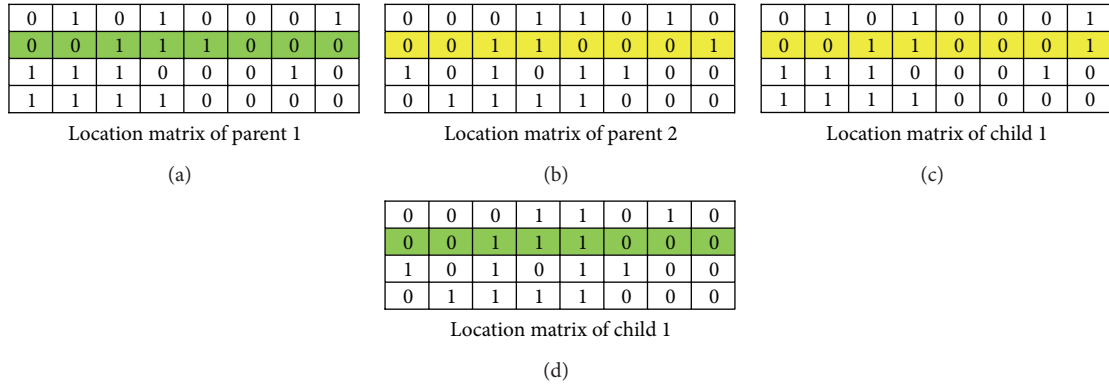
Figure 1: Crossover operator.

0	1	0	1	0	0	0	1
0	0	1	1	1	0	0	0
1	1	1	0	0	0	1	0
1	1	1	1	0	0	0	0

Location matrix before mutation

(a)

0	0	0	1	0	1	0	1
0	0	1	1	1	0	0	0
1	1	1	0	1	0	0	0
1	1	1	1	0	0	0	0

Location matrix after mutation

(b)
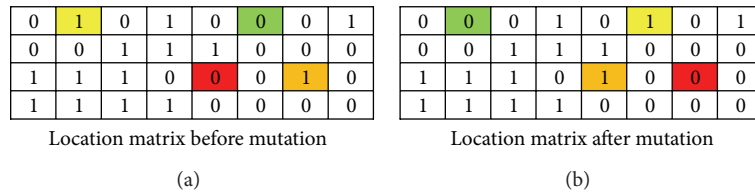
Figure 2: Mutation operator.

Begin

Generate a random primary population equals with population size

Calculate chromosomes fitness function

Do sampling through tournament strategy from primary population and put them in mating pool

Do crossover operator with RC rate

Do mutation operator with RM rate

Do elitism with 1-RC-RM rate

Stop criterion

End
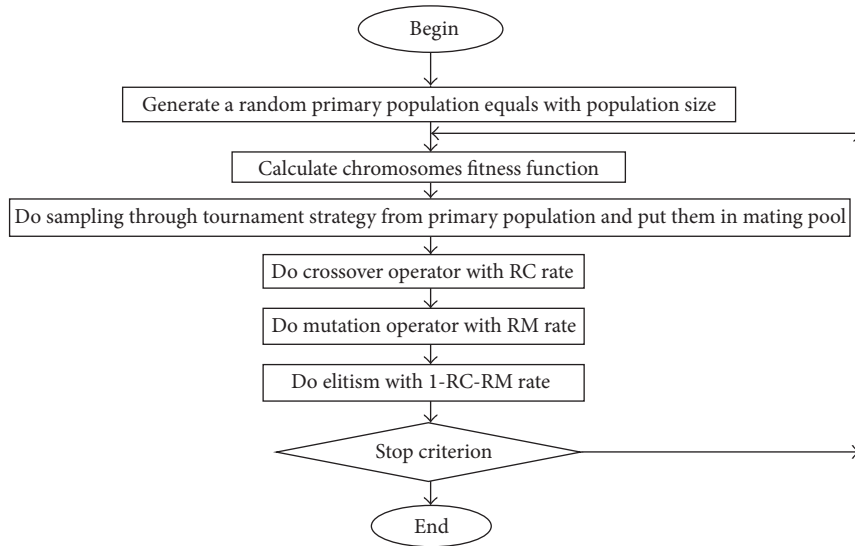
Figure 3: Genetic algorithm flowchart.

Table 1: Simulated annealing algorithm parameters and their levels.

Problem size	Levels	A Cooling rate (α)	B Number of repetitions in each temperature (N)	C Final temperature (T_f)	D Initial temperature (T_0)
Small	1	0.9	20	0.0000001	100
	2	0.925	40	0.00001	10000
	3	0.95	60	0.001	1000000
Large	1	0.94	50	0.0000001	100
	2	0.96	75	0.00001	10000
	3	0.98	100	0.001	1000000

Figure 4: Simulated annealing flowchart.

Figure 5: Diagram of mean effect of the S/N ratio for small problem in genetic algorithm.

TABLE 2: Genetic algorithm parameters and their levels.

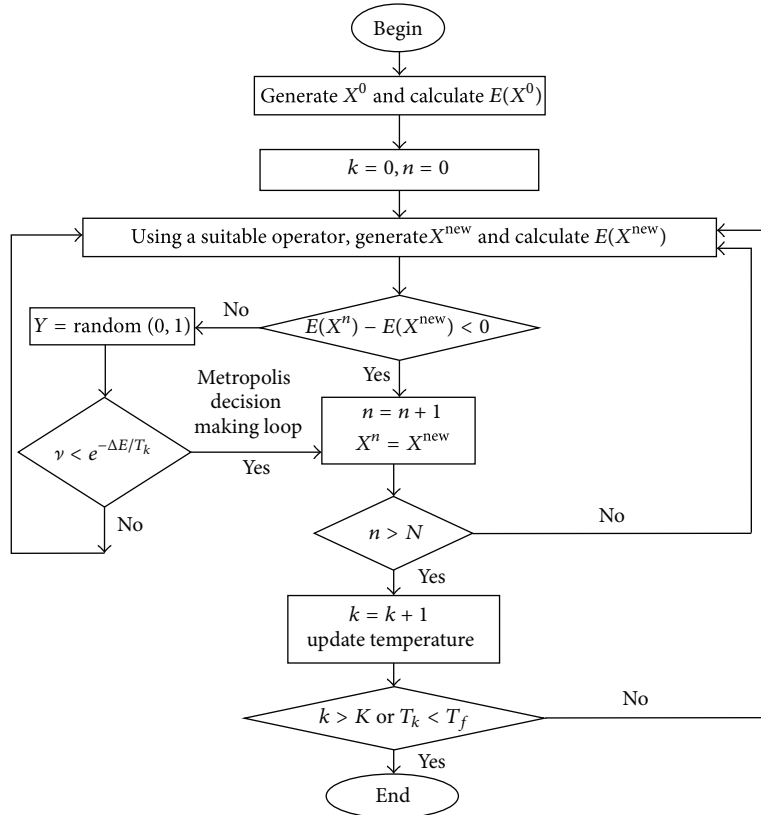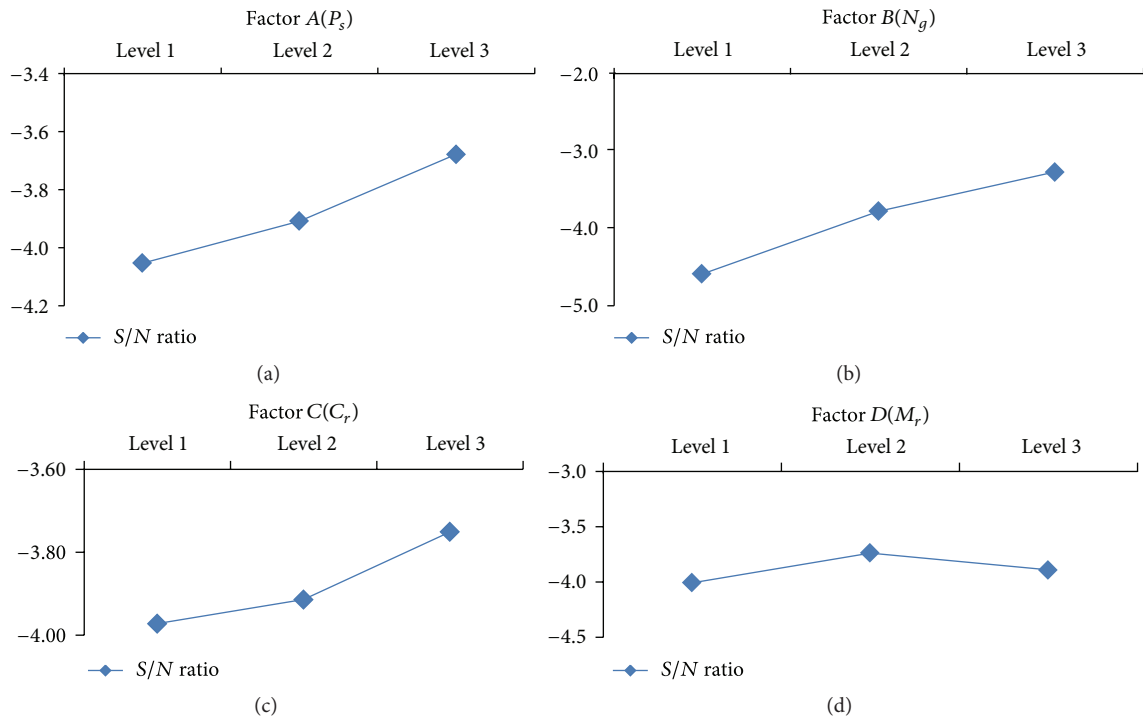| Problem size	Levels	A	B	C	D
		Population size (P_s)	Number of generations (N_g)	Crossover rate (C_r)	Mutation rate (M_r)
Small	1	100	400	0.7	0.01
	2	120	500	0.75	0.02
	3	150	600	0.8	0.05
Large	1	180	600	0.8	0.01
	2	200	800	0.9	0.02
	3	220	1000	0.95	0.05

TABLE 3: Orthogonal array $L9$.

Experiments	A	B	C	D
1	A(1)	B(1)	C(1)	D(1)
2	A(1)	B(2)	C(2)	D(2)
3	A(1)	B(3)	C(3)	D(3)
4	A(2)	B(1)	C(2)	D(3)
5	A(2)	B(2)	C(3)	D(1)
6	A(2)	B(3)	C(1)	D(2)
7	A(3)	B(1)	C(3)	D(2)
8	A(3)	B(2)	C(1)	D(3)
9	A(3)	B(3)	C(2)	D(1)

TABLE 4: Characteristics of instances.

Problem size	Instance name	Number of periods	Number of damaged areas	Number of areas for distribution center	Maximum number of distribution centers in periods
Small	I1	5	(5, 5, 5, 6, 6)	(4, 4, 5, 5, 5)	(2, 2, 2, 2, 2)
	I2	5	(6, 6, 6, 6, 6)	(4, 4, 5, 5, 5)	(2, 2, 3, 3, 3)
	I3	5	(6, 6, 8, 8, 8)	(5, 5, 5, 6, 6)	(2, 2, 3, 3, 3)
	I4	3	(10, 10, 10)	(5, 6, 6)	(3, 4, 4)
	I5	3	(10, 12, 12)	(6, 6, 8)	(3, 4, 4)
	I6	3	(12, 12, 14)	(6, 8, 8)	(3, 5, 5)
	I7	3	(14, 14, 14)	(8, 10, 10)	(4, 5, 5)
	I8	3	(14, 14, 16)	(12, 14, 14)	(4, 5, 5)
Large	I9	5	(30, 30, 30, 35, 35)	(12, 12, 14, 14, 14)	(4, 4, 5, 5, 5)
	I10	5	(30, 30, 35, 35, 35)	(14, 14, 14, 15, 15)	(5, 5, 5, 5, 5)
	I11	5	(35, 35, 35, 40, 40)	(15, 15, 18, 18, 18)	(5, 5, 5, 5, 5)
	I12	5	(40, 40, 40, 45, 45)	(16, 16, 18, 20, 20)	(5, 5, 6, 6, 6)
	I13	5	(45, 45, 45, 48, 50)	(20, 20, 22, 22, 22)	(6, 6, 6, 6, 6)
	I14	5	(45, 48, 48, 48, 50)	(22, 24, 24, 25, 25)	(6, 6, 7, 7, 7)
	I15	5	(50, 55, 55, 50, 50)	(25, 28, 28, 28, 28)	(6, 6, 7, 7, 7)
	I16	5	(50, 55, 55, 55, 55)	(28, 30, 30, 30, 30)	(7, 7, 7, 7, 7)
	I17	3	(80, 100, 100)	(30, 30, 32)	(8, 8, 8)
	I18	3	(100, 100, 120)	(32, 35, 35)	(8, 10, 10)
	I19	3	(120, 120, 140)	(35, 35, 38)	(10, 10, 10)
	I20	3	(140, 140, 160)	(38, 40, 42)	(10, 10, 12)
	I21	3	(150, 160, 150)	(42, 45, 45)	(10, 12, 12)
	I22	3	(160, 160, 170)	(45, 50, 50)	(12, 12, 12)
	I23	3	(180, 180, 200)	(50, 50, 60)	(12, 12, 12)
	I24	3	(200, 220, 220)	(60, 60, 62)	(12, 14, 14)
	I25	3	(220, 250, 250)	(60, 65, 65)	(12, 14, 14)

TABLE 5: Characteristics of sample problems that have been used for Taguchi method.

Problem size	Number of periods	Number of damaged areas	Number of areas for distribution center	Maximum number of distribution centers in periods
Small	3	(10, 10, 10)	(5, 6, 6)	(3, 4, 4)
Large	5	(35, 35, 35, 40, 40)	(15, 15, 18, 18, 18)	(5, 5, 5, 5, 5)

TABLE 6: The optimum factor for genetic algorithm and simulated annealing.

		Problem size	Factors	Level 1	Level 2	Level 3	Optimum factor
Genetic algorithm	Mean S/N ratio	Small	A	−4.0524	−3.9078	−3.6780	$A(3)$
			B	−4.5873	−3.7776	−3.2731	$B(3)$
			C	−3.9725	−3.9144	−3.7512	$C(3)$
			D	−4.0079	−3.7381	−3.8921	$D(2)$
		Large	A	−5.5750	−5.4837	−5.4807	$A(3)$
			B	−6.0750	−5.4906	−4.9739	$B(3)$
			C	−5.6615	−5.4169	−5.4610	$C(2)$
			D	−5.5713	−5.3525	−5.6156	$D(2)$
	Mean objective function	Small	A	0.6301	0.6395	0.6558	$A(3)$
			B	0.5914	0.6479	0.6861	$B(3)$
			C	0.6360	0.6391	0.6502	$C(3)$
			D	0.6334	0.6514	0.6405	$D(2)$
		Large	A	0.5286	0.5339	0.5345	$A(2), A(3)$
			B	0.4995	0.5323	0.5641	$B(3)$
			C	0.5235	0.5380	0.5345	$C(2)$
			D	0.5290	0.5412	0.5258	$D(2)$
Simulated annealing	Mean S/N ratio	Small	A	−4.1877	−4.0033	−3.7003	$A(3)$
			B	−4.1164	−3.9694	−3.8055	$B(3)$
			C	−4.1038	−3.7080	−4.0795	$C(2)$
			D	−4.4123	−4.0730	−3.4060	$D(3)$
		Large	A	−6.4144	−6.0149	−5.4369	$A(3)$
			B	−6.2095	−6.0824	−5.5743	$B(3)$
			C	−5.9112	−5.8779	−6.0771	$C(2)$
			D	−6.3202	−5.8310	−5.7150	$D(3)$
	Mean objective function	Small	A	0.6209	0.6327	0.6544	$A(3)$
			B	0.6262	0.6353	0.6464	$B(3)$
			C	0.6272	0.6537	0.6270	$C(2)$
			D	0.6048	0.6272	0.6760	$D(3)$
		Large	A	0.4817	0.5032	0.5351	$A(3)$
			B	0.4925	0.5006	0.5269	$B(3)$
			C	0.5100	0.5108	0.4993	$C(1), C(2)$
			D	0.4880	0.5131	0.5189	$D(3)$

TABLE 7: Tuned values of the simulated annealing parameters.

Problem size	α	N	T_f	T_0
Small	0.95	60	0.00001	1000000
Large	0.98	100	0.00001	1000000

Factor $A(\alpha)$

Factor $B(N)$

Factor $C(T_f)$
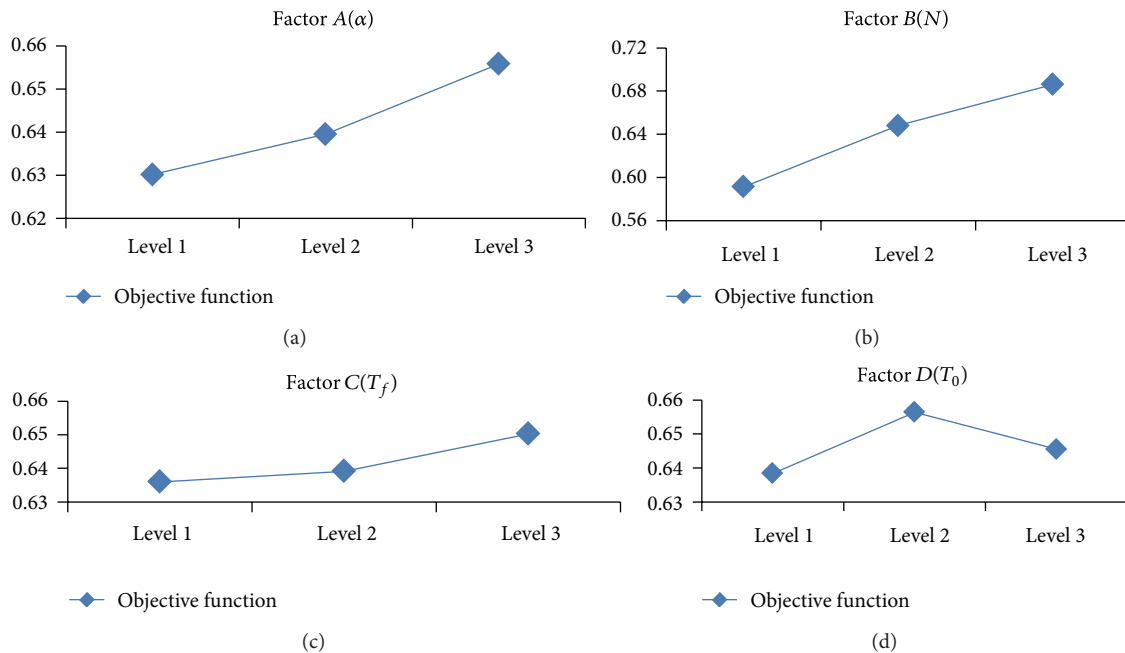
Factor $D(T_0)$

(a)

(b)

(c)

(d)

Figure 6: Diagram of mean effect of parameters based on the objective function for small problem in genetic algorithm.

Figure 7: Iran: South Khorasan.

3.2. Generating Initial Population. After determining a technique to assign a chromosome to each solution, one can create an initial population. In this process, the location matrix is generated randomly.

To initialize the Commodity_Allocated matrices, we use a heuristic method so that first, for each commodity type, the total damaged area requests and total supply of active distribution centers have been evaluated, and, then, considering justice in distribution, commodities are distributed among damaged areas equally. Now, to allocate active centers to damaged areas, an area has been chosen randomly and it would be allocated to an active distribution center obtained by the roulette wheel, which is generated based on the allocation costs. If this distribution center cannot satisfy all predicted demands, the next center obtained by the roulette wheel will meet the remaining demands of this damaged area. Then, one area is chosen randomly from the other damaged areas. All the above-mentioned activities are executed until allocation manner of centers to damaged areas is determined for all the areas.

3.3. Sampling Mechanisms. This mechanism is related to selection chromosomes' method. Tournament selection has been used in this research. Based on this method, first, the value of each chromosome's fitness functions in a population has been calculated, and a tournament size is chosen (here, tournament size is 3). Then, chromosome from one population has been chosen randomly equal with tournament size and, at the end, the best chromosome based on suitability will be transferred to mating pool. This has been repeated up to the number of population. It is possible to create several copies from a chromosome with higher fitness function. This shows that better chromosomes have greater chance to be chosen.

3.4. Genetic Operators. To generate a new generation of the present chromosome, the following genetic operators could be used.

3.4.1. Crossover Operator. In this crossover operator, a row of location matrix of one chromosome is exchanged with the

TABLE 8: Tuned values of the genetic algorithm parameters.

Problem size	P_s	N_g	C_r	M_r
Small	150	600	0.8	0.02
Large	220	1000	0.9	0.02

same row of location matrix of another one and also corresponding elements are exchanged in Commodity_Allocated matrices and inventory matrices. In other words, we choose a random number from $\{1, \ldots, T\}$ (T is the number of time periods), and we exchange corresponding row of location matrix of parent 1 with the same row of location matrix of parent 2 and also corresponding elements are exchanged in Commodity_Allocated matrices and inventory matrices in two chromosomes. Figure 1 shows the crossover operator.

3.4.2. Mutation Operators. Mutation operator operates on location matrix in the developed algorithm. In the first mutation operator, elements corresponding to two distribution centers exchange in a row of location matrix. As distribution center locating conditions change, Commodity_Allocated matrices are valued by the stated heuristic method in Section 3.2.

Figure 2 shows the mutation operator.

In the second mutation operator, one element corresponding to the distribution center in location matrix has been considered. It will be zero if it is one and it will be one if it is zero (provided that the number of active centers does not violate the number of authorized ones).

Now, as the location conditions of distribution center change, Commodity_Allocated matrices are updated by explained heuristic method in Section 3.2.

3.5. Elitism Selection. When genetic operators (mutation and crossover) are used, it would be possible to lose the best chromosomes. Elitism is a method to keep a copy from the best chromosomes in the new generations. The above mechanism makes genetic algorithm keep some of the best solutions in each generation. Previous experiences have proved that this mechanism optimizes genetic algorithm operation and shortens coverage time.

3.6. Algorithm Stop Criterion. Since metaheuristic algorithms do not have any criteria for the global optima, a maximum number of iterations have been considered in order to stop the algorithm. Figure 3 shows the genetic algorithm flowchart.

4. Simulated Annealing Algorithm

Simulated annealing is a probabilistic metaheuristic algorithm which is a local search method. The simulated annealing begins its search from a random initial solution. The iteration loop that characterizes the main procedure randomly generates in each iteration only one neighbor s' of the current solution s. The variation Δ for the value of the objective function $f(x)$ is tested for each neighbor generation. To

test this variation, $\Delta = f(s) - f(s')$ is computed. If the value of Δ is less than 0 (zero), then the new solution s' is automatically accepted to replace s. Otherwise, accepting the new solution s' will depend on the probability established by the Metropolis criteria, which is given by $e^{\Delta/T}$, where T is a temperature parameter, a key variable for the method.

Figure 4 shows the simulated annealing flowchart. In Figure 4, n is the number of internal circles, k is the number of external circles, and $E()$ is the objective function.

In the following, we discuss the proposed SA heuristic in detail, including the solution representation, the generation of the initial solution, and various types of neighborhood.

Solution representation: solution representation is similar to Section 3.1.

Generation of the initial solution: this is similar to Section 3.2.

Neighborhood generation mechanism: we produce neighborhood by mutation operators that is exhibited in Section 3.4.2.

5. Numerical Results

In this section, the developed algorithms have been used on the sample problems that have been generated randomly and the presented genetic algorithm efficacy has been studied. The characteristics of instances are shown in Table 3.

To solve the model used of LINGO 11 on a computer with the following characteristics: intel(R) core(Tm) i3 cpuM330@2.13 Ghz, 4 G RAM. Considering the intense effect of parameter configuration on the performance of algorithms, the Taguchi method is used for the configuration of parameters. Before calibration of the employed algorithms, we run some preliminary tests to find appropriate parameter levels. To obtain more accurate, as well as better sustained, results for the offered algorithm, the following four parameters were configured: T_0, T_f, N, and α for simulated annealing and C_r, M_r, N_g, and P_s for the genetic algorithm. These parameters and their levels are given in Tables 1 and 2. The square matrix with 4 parameters in 3 levels used in the Taguchi method is L9, which is given in Table 3. In Taguchi method, the variation of the output results is measured by means of signal-to-noise (S/N) ratio. Here, the larger value of S/N ratio leads to the smaller variation of the response variable. Factor levels that maximize the appropriate S/N ratio are optimal [33]. The value of S/N ratio is computed by

$$\frac{S}{N} = -10 \log_{10} \left[\frac{1}{n} \sum_{i=1}^{n} \frac{1}{y_i^2} \right], \qquad (26)$$

where n is the number of repetitions under the same experimental conditions and y_i denotes the experimental result at each repetition.

TABLE 9: A summary of the computational results of the small and large instances.

Instance name	LINGO		SA in 10 iterations					GA in 10 iterations					PRAS%	PRBS%
	Exact solution	Time	Average solution	Average time (s)	Standard deviation	Best solution	The best solution's error %	Average solution	Average time (s)	Standard deviation	Best solution	The best solution's error %		
I1	0.5647	15657	0.56385	132.1	0.00056	0.5647	0.000	0.5645	125.0	0.00028	0.5647	0.000	0.115	0.000
I2	0.77818	38956	0.77551	157.2	0.00151	0.7774	0.100	0.7757	140.7	0.00081	0.7765	0.220	0.024	−0.116
I3	0.6429	74378	0.63241	158.1	0.00367	0.6362	1.042	0.6304	156.0	0.00703	0.6362	1.058	−0.319	0.000
I4	0.6938	57478	0.67563	142.7	0.00764	0.6837	1.456	0.6798	135.0	0.00493	0.6874	0.922	0.613	0.538
I5	0.78345	104563	0.75765	167.0	0.01028	0.7687	1.883	0.7554	150.6	0.00838	0.7691	1.835	−0.298	0.052
I6	0.5973	195468	0.57159	203.0	0.00605	0.5798	2.930	0.5761	192.0	0.00722	0.5831	2.377	0.783	0.566
I7	0.6832	246378	0.65593	241.2	0.01078	0.6684	2.166	0.6594	215.1	0.00861	0.6701	1.917	0.526	0.254
I8	0.8843	310234	0.84158	257.4	0.02066	0.8685	1.787	0.8452	222.3	0.01939	0.8691	1.719	0.428	0.069
Average		130389		182.34	0.00764		1.421		167.09	0.00708		1.256	0.234	0.168
I9	—	—	0.74823	1521	0.01755	0.7761	—	0.7779	1431	0.00596	0.7834	—	3.81782	0.93183
I10	—	—	0.63553	1560	0.01813	0.6598	—	0.6633	1492	0.00632	0.6715	—	4.18664	1.74236
I11	—	—	0.51612	1625	0.02334	0.5465	—	0.5534	1553	0.00858	0.5698	—	6.73990	4.08915
I12	—	—	0.71373	1758	0.02807	0.7651	—	0.7353	1661	0.02576	0.7601	—	2.92821	−0.6578
I13	—	—	0.73424	1821	0.04430	0.7924	—	0.8055	1749	0.02185	0.8325	—	8.84441	4.81681
I14	—	—	0.62172	1972	0.01734	0.6538	—	0.6587	1881	0.03170	0.6947	—	5.60835	5.88743
I15	—	—	0.55941	2084	0.04037	0.6089	—	0.6099	2007	0.03593	0.6548	—	8.27840	7.00977
I16	—	—	0.66088	2218	0.04091	0.7191	—	0.6860	2132	0.03373	0.7165	—	3.66461	−0.3628
I17	—	—	0.51391	1732	0.03563	0.5606	—	0.5535	1652	0.03755	0.5963	—	7.14931	5.98691
I18	—	—	0.6941	1829	0.05634	0.7723	—	0.7729	1784	0.04592	0.8435	—	10.1895	8.44102
I19	—	—	0.59042	1997	0.05718	0.6773	—	0.6880	1903	0.03741	0.7326	—	14.1843	7.54845
I20	—	—	0.73554	2132	0.05335	0.8231	—	0.8227	2012	0.03815	0.8901	—	10.5943	7.52724
I21	—	—	0.47935	2217	0.05890	0.5469	—	0.5448	2158	0.03160	0.6035	—	12.0103	9.37862
I22	—	—	0.57803	2349	0.05441	0.6642	—	0.6463	2263	0.04003	0.7108	—	10.5659	6.55599
I23	—	—	0.62408	2494	0.06076	0.6838	—	0.6871	2321	0.04458	0.7593	—	9.16923	9.94336
I24	—	—	0.570474	2683	0.05570	0.6419	—	0.6260	2532	0.05311	0.6980	—	8.87142	8.03724
I25	—	—	0.71449	2793	0.06001	0.7892	—	0.7690	2671	0.054704	0.8326	—	7.08601	5.21258
Average	—	—		2046.18	0.04249		—		1953.06	0.03252		—	7.786	5.417

TABLE 10: The amount of inventory has been estimated in the candidate areas for the establishment of distribution centers ($*1000$).

		Sarayan	Qaen	Birjand
Period 1	Commodity 1	(19, 23, 30)	(54, 57, 59)	(128, 130, 134)
	Commodity 2	(96, 99, 107)	(144, 149, 162)	(274, 280, 291)
Period 2	Commodity 1	(47, 53, 55)	(67, 72, 79)	(175, 180, 182)
	Commodity 2	(102, 113, 124)	(192, 209, 223)	(389, 400, 419)
Period 3	Commodity 1	(41, 43, 48)	(69, 78, 85)	(192, 200, 212)
	Commodity 2	(89, 96, 109)	(117, 127, 139)	(270, 280, 296)

Characteristics of sample problems that have been used for Taguchi method are shown in Table 5. Calculating all test results using the Taguchi method, the mean rate of S/N for small-scale problem is presented in Figure 5 and the mean of the objective function for small-scale problem is shown in Figure 6.

Optimum factors for genetic algorithm and simulated annealing are given in Table 6. The final configurations of parameters for all algorithms are summarized in Tables 7 and 8.

Table 9 shows that the proposed genetic algorithm and simulated annealing algorithm operate faster than LINGO with little errors and the standard deviation is not considerable in ten iterations indicating that the solutions do not have deviation. Now, we generate a number of sample problems with small and large size that are exhibited in Table 4 and then operate the proposed genetic algorithm and simulated annealing algorithm on these problems. Table 9 shows more details on computational results for all test problems. The CPU times corresponding to the exact solution, as seen in Table 9, show that computational time grows exponentially by size of the problem. Unfortunately, for the large-scale calculations, LINGO cannot even find the feasible solution in the limited run time.

In Table 9, we present the following: average value, standard deviation, best solution, computational time, PRAS%, and PRBS% for each problem. As observed in Table 9, there are only 2 cases out of 25 for which the SA algorithm performs better than the GA. Also, in 2 cases out of 25, the SA algorithm and GA behave identically. Moreover, in the other cases than those mentioned, GA performs better than the SA. Finally, in an average sense, the genetic algorithm gives less standard deviation than the simulated annealing. These observations show that the GA gives more stable results than the SA. Solution average in genetic algorithm meets more suitable conditions especially in the case of large size problems. As can be seen, the percent reduction best solution (%PRBS) and the percent reduction average solution (%PRAS) are both positive values, so the GA performs better than the SA.

The percent reduction best solution (%PRBS) and the percent reduction average solution (%PRAS) are calculated using

$$\%\text{PRBS} = \frac{\text{Best}_{\text{GA}} - \text{Best}_{\text{SA}}}{\text{Best}_{\text{GA}}} \times 100\%,$$

$$\%\text{PRAS} = \frac{\text{Average}_{\text{GA}} - \text{Average}_{\text{SA}}}{\text{Average}_{\text{GA}}} \times 100\%. \tag{27}$$

For the proposed algorithms, the average results of the ten simulations are also presented. The efficiency of the two algorithms is measured by the quality of the produced solutions. The quality is given in terms of the relative deviation from the optimum solution that the best solution's error = (LINGO solution − the best solution)/LINGO solution, where LINGO solution denotes the solution of the LINGO software and the best solution is considered to be the solution of the metaheuristic algorithms. As it can be seen from Table 9, the algorithms lead to very good and stable results for most cases. Also, the genetic algorithm performs better than the simulated annealing, in each and every instance.

The comparison of LINGO with the proposed genetic algorithm shows that the GA can find approximately an optimal solution in a shorter time compared to the LINGO as presented in Table 9. In Table 9, it is also shown that the average gap between the optimal and the GA solutions is 1.421%. The maximum gap is related to the test problem I_6.

As observed in Table 9, the standard deviation has increased as the problem's dimension increases. To obtain better solution in large dimensions, population size and the number of generations should both increase.

A small size example is considered to test the reliability of the model. Figure 7 shows the map of south-khorasan province of Iran that has been used at this example. Three cities (Birjand, Qaen, and Sarayan) are selected to set up relief commodities distribution centers for different periods. Furthermore, nine cities including Birjand, Qaen, Darmiyan, Nehbandan, Ferdows, Zirkoh, Sarayan, Sarbisheh, and Boshruyeh are considered as damaged areas. The fixed cost to open distribution centers for all the areas is $25000, $26000, and $27500. The shortage penalty for each commodity is $1.5 for commodity 1 and $2.5 for commodity 2. These values are confirmed with experts' estimations. In this example, two types of relief commodities have been considered. In addition, there are two active centers for each period. The model has been solved for a situation that the number of planning periods is 3. The demand of damaged areas is available as triangular fuzzy numbers in Table 11. The amount of inventory in distribution centers is presented in Table 10. The proposed model tries to minimize the cost of commodities relocation and maximize the demand coverage ratio of damaged area for fair distribution. This is done by a multiobjective possibility linear programming model.

As shown in Tables 12 and 13, considering the inventories of distribution centers and damaged area demands in periods

TABLE 11: Demand of damaged areas ($*1000$).

		Ferdows	Boshruyeh	Darmiyan	Sarayan	Sarbisheh	Nehbandan	Zirkoh	Qaen	Birjand
Period 1	Commodity 1	(38, 40, 42)	(44, 50, 57)	(45, 50, 53)	(33, 36, 39)	(34, 38, 39)	(49, 50, 54)	(36, 40, 42)	(42, 45, 46)	(56, 60, 63)
	Commodity 2	(68, 70, 72)	(65, 69, 73)	(91, 95, 99)	(86, 89, 98)	(55, 60, 66)	(53, 55, 59)	(66, 70, 76)	(76, 80, 89)	(92, 100, 104)
Period 2	Commodity 1	(44, 45, 47)	(54, 55, 58)	(42, 45, 48)	(42, 46, 47)	(35, 39, 41)	(56, 60, 61)	(32, 35, 38)	(35, 38, 41)	(61, 65, 67)
	Commodity 2	(76, 80, 86)	(75, 79, 83)	(102, 110, 119)	76, 80, 88)	(72, 75, 79)	(66, 70, 78)	(71, 75, 79)	(74, 78, 84)	(101, 110, 112)
Period 3	Commodity 1	(48, 50, 52)	(47, 49, 51)	(56, 60, 63)	(36, 40, 42)	(27, 30, 32)	(43, 45, 47)	(41, 45, 49)	(54, 60, 64)	(64, 70, 74)
	Commodity 2	(91, 95, 99)	(85, 87, 94)	(76, 80, 84)	(77, 78, 83)	(62, 65, 69)	(71, 75, 79)	(80, 83, 88)	(87, 89, 99)	(119, 125, 129)

Table 12: Allocated commodity (type 1) (∗1000).

		Ferdows	Boshruyeh	Darmiyan	Sarayan	Sarbisheh	Nehbandan	Zirkoh	Qaen	Birjand
Period 1	Birjand			25	8	20	27	20		31
	Qaen	21	26		11				23	
Period 2	Birjand			32	32	27	42			46
	Qaen	32	40					25	27	
Period 3	Birjand			46	23	23	34	22		53
	Qaen	38	38		7			13	45	

Table 13: Allocated commodity (type 2) (∗1000).

		Ferdows	Boshruyeh	Darmiyan	Sarayan	Sarbisheh	Nehbandan	Zirkoh	Qaen	Birjand
Period 1	Birjand			57	47	46	43	13		76
	Qaen	54	53	17	23			42	63	
Period 2	Birjand			105	55	72	68			103
	Qaen	77	76		22			71	75	
Period 3	Birjand			52	29	42	49	30		80
	Qaen	61	58		22			24	59	

Table 14: The amount of inventory in established distribution centers (∗1000).

		Birjand	Qaen
Period 1	Commodity 1	(128, 130, 134)	(73, 80, 89)
	Commodity 2	(274, 280, 291)	(240, 248, 269)
Period 2	Commodity 1	(175, 180, 182)	(114, 125, 134)
	Commodity 2	(389, 400, 419)	(294, 322, 347)
Period 3	Commodity 1	(192, 200, 212)	(110, 121, 133)
	Commodity 2	(270, 280, 296)	(206, 223, 248)

1, 2, and 3, Birjand and Qaen are considered as active distribution centers. The amount of inventory in active distribution centers is presented in Table 14. Tables 12 and 13 show how to allocate goods from distribution centers to the affected areas. As demonstrated, it has been tried as much as possible to use one distribution center as a closest center to damaged area. However, if the inventory of a distribution center is not sufficient, several distribution centers will be used to fulfill all demands. Reviewing the tables shows that almost all damaged areas receive commodities equally to have fair distribution of commodities. For instance, almost 52% of the 1st commodity demands and 77% of the 2nd commodity demands are met in period 1. Therefore, accuracy and reliability of the model are both acceptable.

6. Conclusions

It is clear that logistic activities are very important in response phase of the disaster management. Also, precise programming can improve the efficacy of the system. According to the crucial role of location-allocation in reducing cost and time, a multiobjective possibility programming model has been developed for relief logistics. This model follows two main objectives. The first one is minimizing good distribution cost and the second one is justice in good distribution process,

while the efforts are to maximize the minimum number of satisfied requests. Considering estimated information from problem conditions, some of the model parameters such as the number of damaged area requests and distribution center supply in different periods and transportation cost have been seeded as triangular fuzzy numbers. As observed earlier, the quality of the genetic algorithm is relatively higher, especially in large-scale problems.

Finally, for future research, the relief chain can be reconfigured with three echelons: central disaster facilities of the city as main suppliers, local emergency facilities as distributors, and urban areas as customer. Also, location and routing can be considered simultaneously.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] L. Douglas, "Logistics for disaster relief," *IIE Solutions*, vol. 29, no. 6, pp. 26–29, 1997.

[2] B. Balcik and B. M. Beamon, "Facility location in humanitarian relief," *International Journal of Logistics Research and Applications*, vol. 11, no. 2, pp. 101–121, 2008.

[3] A. Thomas, "Humanitarian logistics: enabling disaster response," The Fritz Institute, 2003, http://www.fritzinstitute.org/.

[4] L. Brotcorne, G. Laporte, and F. Semet, "Ambulance location and relocation models," European Journal of Operational Research, vol. 147, no. 3, pp. 451–463, 2003.

[5] F. Barzinpour and V. Esmaeili, "A multi-objective relief chain location distribution model for urban disaster management," International Journal of Advanced Manufacturing Technology, vol. 70, no. 5–8, pp. 1291–1302, 2014.

[6] C. G. Rawls and M. A. Turnquist, "Pre-positioning of emergency supplies for disaster response," Transportation Research B, vol. 44, no. 4, pp. 521–534, 2010.

[7] G. Barbarosoğlu and Y. Arda, "A two-stage stochastic programming framework for transportation planning in disaster response," Journal of the Operational Research Society, vol. 55, no. 1, pp. 43–53, 2004.

[8] A. Akkihal, Inventory pre-positioning for Humanitarian operations [M.S. thesis], MIT Center for Transportation & Logistics, 2006.

[9] G. H. Tzeng, H. J. Cheng, and T. D. Huang, "Multi-objective optimal planning for designing relief delivery systems," Transportation Research E, vol. 43, no. 6, pp. 673–686, 2007.

[10] H. Jia, F. Ordóñez, and M. Dessouky, "A modeling framework for facility location of medical services for large-scale emergencies," IIE Transactions, vol. 39, no. 1, pp. 41–55, 2007.

[11] P. Beraldi and M. E. Bruni, "A probabilistic model applied to emergency service vehicle location," European Journal of Operational Research, vol. 196, no. 1, pp. 323–331, 2009.

[12] H. O. Mete and Z. B. Zabinsky, "Stochastic optimization of medical supply location and distribution in disaster management," International Journal of Production Economics, vol. 126, no. 1, pp. 76–84, 2010.

[13] C. G. Rawls and M. A. Turnquist, "Pre-positioning of emergency supplies for disaster response," Transportation Research B, vol. 44, no. 4, pp. 521–534, 2010.

[14] M. S. Jabal-Ameli, A. Bozorgi-Amiri, and M. Heydari, "A multi-objective possibilistic programming model for relief logistics problem," International Journal of Industrial Engineering & Production Management, vol. 22, no. 1, pp. 65–75, 2011.

[15] S. Duran, M. A. Gutierrez, and P. Keskinocak, "Pre-positioning of emergency items for CARE international," Interfaces, vol. 41, no. 3, pp. 223–237, 2011.

[16] M. Zeng and X. Wu, "A location-routing problem model for emergency relief system of natural disaster," in Logistics, pp. 1643–1649, 2009.

[17] A. Bozorgi-Amiri, M. S. Jabalameli, and S. M. J. Mirzapour Al-e-Hashem, "A multi-objective robust stochastic programming model for disaster relief logistics under uncertainty," OR Spectrum, vol. 35, no. 4, pp. 905–933, 2013.

[18] A. Bozorgi-Amiri, M. S. Jabalameli, M. Alinaghian, and M. Heydari, "A modified particle swarm optimization for disaster relief logistics under uncertain environment," International Journal of Advanced Manufacturing Technology, vol. 60, no. 1–4, pp. 357–371, 2012.

[19] S. A. Yazdian and K. Shahanaghi, "A multi-objective possibilistic programming approach for locating distribution centers and allocating customers demands in supply chains," International Journal of Industrial Engineering Computations, vol. 2, no. 1, pp. 193–202, 2011.

[20] J. S. Tancrez, J. C. Lange, and P. Semal, "A location-inventory model for large three-level supply chains," Transportation Research E, vol. 48, no. 2, pp. 485–502, 2012.

[21] V. Schmid, "Solving the dynamic ambulance relocation and dispatching problem using approximate dynamic programming," European Journal of Operational Research, vol. 219, no. 3, pp. 611–621, 2012.

[22] P. Chrétienne, P. Fouilhoux, E. Gourdin, and J. M. Segura, "The location-dispatching problem: polyhedral results and content delivery network design," Electronic Notes in Discrete Mathematics C, vol. 36, pp. 867–874, 2010.

[23] H. Toro-Diaz, M. E. Mayorga, S. Chanta, and A. M. Laura, "Joint location and dispatching decisions for Emergency Medical Services," Computers & Industrial Engineering, vol. 64, no. 4, pp. 917–928, 2013.

[24] G. Guillermo Cabrera, E. Cabrera, R. Soto, L. Jose Miguel Rubio, B. Crawford, and F. Paredes, "A hybrid approach using an artificial bee algorithm with mixed integer programming applied to a large-scale capacitated facility location problem," Mathematical Problems in Engineering, vol. 2012, Article ID 954249, 14 pages, 2012.

[25] P. Murali, F. Ordóñez, and M. M. Dessouky, "Facility location under demand uncertainty: response to a large-scale bio-terror attack," Socio-Economic Planning Sciences, vol. 46, no. 1, pp. 78–87, 2012.

[26] M. Xi, F. Ye, Z. Yao, and Q. Zhao, "A modified p-median model for the emergency facilities location problem and its variable neighbourhood search-based algorithm," Journal of Applied Mathematics, vol. 2013, Article ID 375657, 10 pages, 2013.

[27] S. Gharegozloo Hamedani, M. S. Jabalameli, and A. Bozorgi-Amiri, "A multi-objective model for locating distribution centers in a supply chain network considering risk and inventory decisions," Management Science Letters, vol. 3, pp. 1077–1088, 2013.

[28] R. Abounacer, M. Rekik, and J. Renaud, "An exact solution approach for multi-objective location-transportation problem for disaster response," Computers & Operations Research, vol. 41, pp. 83–93, 2014.

[29] V. Esmaeili and F. Barzinpour, "Integrated decision making model for urban disaster management: a multi-objective genetic algorithm approach," International Journal of Industrial Engineering Computations, vol. 5, no. 1, pp. 55–70, 2014.

[30] H. J. Zimmermann, "Fuzzy programming and linear programming with several objective functions," Fuzzy Sets and Systems, vol. 1, no. 1, pp. 45–55, 1978.

[31] D. E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley Publishing Company, 1986.

[32] C. Cotta and V. Hemert, Studies in Computational Intelligence: Recent Advances in Evolutionary Computation for Combinatorial Optimization, Springer, 2008.

[33] B. Naderi, M. Zandieh, and S. M. T. Fatemi Ghomi, "Scheduling job shop problems with sequence-dependent setup times," International Journal of Production Research, vol. 47, no. 21, pp. 5959–5976, 2009.