Computers & Industrial Engineering 76 (2014) 204–221

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

Solving a new bi-objective location-routing-inventory problem in a distribution network by meta-heuristics ☆

N. Nekooghadirli [a,*], R. Tavakkoli-Moghaddam [b,c], V.R. Ghezavati [a], S. Javanmard [a]

[a] School of Industrial Engineering, South Tehran Branch, Islamic Azad University, Tehran, Iran
[b] School of Industrial Engineering and Research Institute of Energy Management & Planning, College of Engineering, University of Tehran, Tehran, Iran
[c] Research Center for Organizational Processes Improvement, Sari, Iran

ARTICLE INFO

Article history:
Received 17 May 2013
Received in revised form 1 May 2014
Accepted 3 August 2014
Available online 13 August 2014

Keywords:
Supply chain network
Bi-objective model
Facility location
Vehicle routing
Inventory control

ABSTRACT

This paper presents a novel bi-objective location-routing-inventory (LRI) model that considers a multi-period and multi-product system. The model considers the probabilistic travelling time among customers. This model also considers stochastic demands representing the customers' requirement. Location and inventory-routing decisions are made in strategic and tactical levels, respectively. The customers' uncertain demand follows a normal distribution. Each vehicle can carry all kind of products to meet the customer's demand, and each distribution center holds a certain amount of safety stock. In addition, shortage is not allowed. The considered two objectives aim to minimize the total cost and the maximum mean time for delivering commodities to customers. Because of NP-hardness of the given problem, we apply four multi-objective meta-heuristic algorithms, namely multi-objective imperialist competitive algorithm (MOICA), multi-objective parallel simulated annealing (MOPSA), non-dominated sorting genetic algorithm II (NSGA-II) and Pareto archived evolution strategy (PAES). A comparative study of the forgoing algorithms demonstrates the effectiveness of the proposed MOICA with respect to four existing performance measures for numerous test problems.

1. Introduction

Nowadays, the efficiency of industries is the bottleneck of progress in the competitive environment of marketing. That is why the companies have to increase their efficiency in all fields, specifically, in their logistics' operations. The present study considers the new integrated multi-objective model for the location-routing-inventory problem in a multi-product and multi-period supply chain system. Additionally, the model considers the probabilistic travelling time among customers. Considering these complexities make the problem more similar to real-life problems. In real world, there exist numerous industries making three location-routing-inventory (LRI) decisions and their interactions in multi-products and multi-periods system simultaneously. To the best of our knowledge, this problem has not been surveyed these assumptions altogether.

According to above-mentioned reasons, the main contributions of this paper, which distinguishes from other papers in the related literature review, are as follows:

- Considering an integrated LRI problem
- Considering a bi-objective integrated LRI problem
- Considering a multi-product supply chain system
- Considering a multi-period supply chain system
- Considering uncertainty in a multi-product and multi-period supply chain system
- Considering the probabilistic time among customers
- Considering the transportation cost consisting of the travelling distance related cost and vehicle fixed cost for determining the usage of vehicles.
- Solving the model by new multi-objective meta-heuristic algorithms (e.g., MOICA and MOPSA).
- Comparing the proposed meta-heuristics with two well-known evolutionary meta-heuristics (e.g., NSGA II and PEAS) in terms of four comparison metrics.

A logistic system consists of three important elements, namely facility location, vehicle routing and inventory control decisions. Since, these key elements are highly dependent, an integrated decision problem is considered as an integrated logistic system. This problem is represented under different assumptions in the related literature review.

Liu and Lee (2003) proposed a single product, multi-depot LRI problem and applied a two phase heuristic method to solve the

☆ This manuscript was processed by Area Editor Qiuhong Zhao.
* Corresponding author. Tel.: +98 912 2565715.
E-mail address: najme.qadiri@gmail.com (N. Nekooghadirli).

http://dx.doi.org/10.1016/j.cie.2014.08.004
0360-8352/

problem. Gaur and Fisher (2004) studied a periodic inventory-routing problem in a super market chain. Liu and Lin (2005) developed the same model solved by a combined tabu search and simulated annealing algorithms. Kang and Kim (2010) considered an integrating inventory replenishment model and delivery planning in a two-level supply chain consisting of a supplier and a retailer. Shen and Qi (2007) proposed a single-product, single-period LRI problem with an approximate routing cost and solved the LRI model by a Lagrangian relaxation based solution algorithm, their model was introduced as a modified inventory-location model given in Daskin, Coullard, and Shen (2002). Chanchan, Zujun, and Huajun (2008) formulated a dynamic LRI problem in a closed loop supply chain solved by a two-phase heuristic algorithm. Ahmadi Javid and Azad (2010) developed the model presented by Shen and Qi (2007). Their model simultaneously optimizes location, inventory and routing decisions without approximation, and is solved by a heuristic method based on a hybridization of tabu search and simulated annealing.

Ambrosino and Scutella (2005) proposed the dynamic version of a model consisting of three integrated decisions. Moin, Salhi, and Aziz (2011) addressed an inventory routing, many-to-many distribution network consisting of an assembly plant and many distinct suppliers where each supplies a distinct product. Hiassat and Diabat (2011) studied the LRI problem with perishable product, through a multi-period model. Bard and Nananukul (2009) suggested a periodic inventory routing problem, with the objective of maximizing the net benefits associated with making deliveries in a specific time period, in which backlogging is not permitted. Also, the inventories can be accumulated at the customer sites. Al Dhaheri and Diabat (2010) formulated a new model of an inventory-location problem with multiple products and risk pooling. Abdelmaguid, Dessouky, and Ordóñez (2009) studied a multi-period inventory routing problem that shortage is permitted and solved by a developed constructive and improvement heuristics and obtained to approximate the solutions. Liao, Hsieh, and Lai (2011) presented a location-inventory problem based on vendor-managed inventory setup. In this model, they minimized the total system costs and maximized the customer service by two performance measurements and they solved the model by a multi-objective evolutionary approach. Hanczar (2012) proposed an application of a multi-item inventory routing problem in a fuel distribution problem. Recently Ahmadi Javid and Seddighi (2012) studied a location-routing-inventory model with a multi-source distribution network. Their model considers the LRI model in a three-level distribution network and a multiphase heuristic algorithm based on simulated annealing (SA) and ant colony system (ACS) is proposed to solve their model.

Based on the above-mentioned studies, there are a few studies, in which all the three problems (i.e., LRI) are simultaneously considered.

In this paper, we consider the LRI problem in a multi-period and multi-product supply chain network. We also assume probabilistic travelling times among customers. In this paper, an extension of the model presented by Ahmadi Javid and Azad (2010) is discussed and solved by four multi-objective meta-heuristics.

2. Problem statement

The presented LRI problem accounts for a two-echelon logistic distribution system that locates and allocates a set of distribution centers to serve a set of customers geographically dispersed in a specific region. The associated model determines an optimal inventory policy and routes of vehicles to satisfy the customers' demand through periods of planning horizon for each kind of product. Additionally, the main aims in this model are to minimize the total cost and the maximum customer service time. The total cost consists of locating open distribution centers costs, transportation costs, inventory holding and ordering costs. Different capacity levels for each distribution center are assumed, leading to a lower cost of capacity distribution centers. Assumptions considered in this study are given in Section 2.1.

2.1. Assumptions

The following are the main assumption of the presented model.

- A two-echelon distribution system is considered.
- Some distribution centers (DCs) are taken into account to meet the customer demands with multi products.
- Each customer has an uncertain independent demand and follows a normal distribution, which should be satisfied for each product in each period.
- Vehicle fleet is heterogeneous which means that vehicles do not have the same capacity.
- Each DC $_j$ follows a (Q_{jpt}, R_{jpt}) inventory policy, in this policy when the inventory level of product p in period t at distribution center j gets to or below a reorder point R_{jpt}, a fixed quantity Q_{jpt} is ordered to the supplier. In addition, each distribution center holds a safety stock from each product in each period.
- The transportation cost includes travelling distance related cost and vehicle fixed cost for determining usage of vehicle v.
- There exist a limited capacity for DCs and vehicles.
- Each DC has four capacity levels with four different costs.
- Numerous DCs are opened.
- No route is considered between DCs.
- Locating and allocating decisions are strategic and not related to periods.
- Shortage is not permitted.
- Travelling times among customers are assumed to be probabilistic nature.

2.2. Model formulation

The notations used in this model are explained below:

Sets	
K and L	Set of customers
J and J'	Set of potential distribution centers
N_j	Set of capacity levels available to distribution center $j (j \in J)$
V	Set of vehicles
M	Aggregate set of customers and potential distribution centers $(k \cup j)$
P	Set of products
T	Set of periods along time horizon
Parameters	
μ_{kpt}	Mean of customer k demand in period t from product p $(\forall k \in K, \forall p \in P, \forall t \in T)$
δ^2_{kpt}	Variance of customer k demand in period t from product p $(\forall k \in K, \forall p \in P, \forall t \in T)$
f^n_j	Establishing cost of DC j with capacity level n in time horizon$(\forall j \in J, \forall n \in Nj)$
b^n_j	Capacity with level n for DC j in time horizon $(\forall j \in J, \forall n \in Nj)$
$d_k l$	Transportation cost for travelling from node k to node $l(\forall k, l \in M)$
Vc_v	Maximum capacity of vehicle $v(\forall v \in V)$

(continued on next page)

q	Number of visits of each customer in a year
H_{jpt}	Inventory holding cost of product p in period t at DC j $(\forall j \in J, \forall p \in P, \forall t \in T)$
P_j	Fixed ordering cost to the supplier by DC j $(\forall j \in J)$
Lt_{jt}	Lead time of DC j in period t $(\forall j \in J, \forall t \in T)$
g_j	Fixed shipping cost for transferring products from supplier to DC j per shipment $(\forall j \in J)$
a_{jp}	Shipment cost for transferring product p from supplier to DC j $(\forall j \in J, \forall p \in P)$
B	Number of customers contained in set K, (i.e., $B = \lvert K \rvert$)
Pr_v	Fixed usage cost of vehicle $v(\forall v \in V)$
t_{kl}	Travelling time from node k to node $l(\forall k, l \in M)$
$E[t_{kl}]$	Mean travelling time from node k to node $l(\forall k, l \in M)$
α	Level of service for customer orders that should be satisfied
Z_α	Standard normal deviate such that $P(z \leqslant z_\alpha) = \alpha$
β	Weight factor associated with transportation cost
θ	Weight factor associated with the inventory cost

Weight factors β and θ are proportional weights to adjust the relative proportion of location, transportation and inventory costs in the objective function.

$$M_{kvt} - M_{lvt} + (B \times R_{klvt}) \leqslant B - 1 \quad \forall t \in T \forall k, l \in k, \forall v \in V \tag{6}$$

$$\sum_{l \in M} R_{klvt} - \sum_{l \in M} R_{lkvt} = 0 \quad \forall t \in T \forall k \in M, \forall v \in V \tag{7}$$

$$\sum_{j \in J} \sum_{k \in k} R_{jkvt} \leqslant 1 \quad \forall t \in T, \forall v \in V \tag{8}$$

$$\sum_{l \in M} R_{klvt} + \sum_{l \in M} R_{jlvt} - Y_{jk} \leqslant 1 \quad \forall t \in T, \forall j \in J, \forall k \in K, \forall v \in V \tag{9}$$

$$\sum_{n \in N_j} U_j^n \leqslant 1 \quad \forall j \in j \tag{10}$$

$$\sum_{p \in P} \sum_{k \in k} \mu_{kpt} Y_{jk} \leqslant \sum_{n \in N_j} b_j^n U_j^n \quad \forall t \in T, \forall j \in J \tag{11}$$

$$R_{jj'} = 0 \quad \forall t \in T, \forall v \in V, \forall j, j' \in J \tag{12}$$

$$\sum_{j \in J} Y_{jk} = 1 \quad \forall k \in K \tag{13}$$

$$Y_{jk} \in \{0, 1\} \quad \forall j \in J, \forall k \in k \tag{14}$$

$$U_j^n \in \{0, 1\} \quad \forall j \in J, \forall n \in Nj \tag{14}$$

$$R_{klvt} \in \{0, 1\} \quad \forall t \in T, \forall k, l \in M, \forall v \in V \tag{14}$$

$$M_{kvt} \text{free variable} \quad \forall k \in k, \forall v \in v, \forall t \in T \tag{15}$$

$$T_{\max} > 0 \tag{15}$$

Decision variables

$$R_{klvt} = \begin{cases} 1 & \text{if } k \text{precedes } l \text{immediately in a route of vehicle } v \text{in period} t \\ 0 & \text{otherwise} \end{cases} \quad (\forall k, l \in M, v \in V, t \in T)$$

$$Y_{jk} = \begin{cases} 1 & \text{if customer } k \text{is assigned to distribution center } j \\ 0 & \text{otherwise} \end{cases} \quad (\forall j \in j, \forall k \in K)$$

$$U_j^n = \begin{cases} 1 & \text{if distribution center} j \text{is opened with capacity level } n \\ 0 & \text{otherwise} \end{cases} \quad (\forall j \in J, \forall n \in N_j)$$

T_{max} Maximum time for delivering to customers
M_{kvt} Sub-tour elimination variable for customer k in route of vehicle v in period t $(\forall k \in K, \forall v \in V, \forall t \in T)$
Q^*_{jpt} Optimal order size at distribution center j for product p in period t $(\forall j \in J, \forall p \in P, \forall t \in T)$

2.3. Mathematical model

$$\min \sum_{j \in J} \sum_{n \in N_j} f_j^n U_j^n + \beta q \left(\sum_{v \in V} \sum_{k \in M} \sum_{l \in M} \sum_{t \in T} d_{kl} R_{klvt} + \sum_{j \in J} \sum_{\in M} \sum_{v \in V} \sum_{t \in T} \Pr_v R_{jlvt} \right)$$
$$+ \sum_{j \in J} \left[\sqrt{2\theta \sum_{k \in K} \sum_{p \in P} \sum_{t \in T} h_{jpt} (\theta P_j + \beta g_j) \mu_{kpt} Y_{jk}^2} \right.$$
$$\left. + \beta \sum_{p \in P} \sum_{t \in T} a_{jp} \sum_{k \in K} \mu_{kpt} Y_{jk}^2 + \theta \sum_{p \in P} \sum_{t \in T} h_{jpt} z_\alpha \sqrt{lt_{jt} \sum_{k \in k} \delta_{kpt}^2 Y_{jk}^2} \right] \tag{1}$$

$$\min \quad T_{\max} \tag{2}$$

s.t.

$$\sum_{k \in M} \sum_{l \in K} E(t_{kl}) R_{klvt} \leqslant T_{\max} \quad \forall v \in V, \forall t \in T \tag{3}$$

$$\sum_{v \in v} \sum_{l \in M} R_{klvt} = 1 \quad \forall t \in T \forall k \in k \tag{4}$$

$$\sum_{p \in P} \sum_{l \in k}^{n} \mu_{lpt} \sum_{k \in M}^{n} R_{klvt} \leqslant Vc_v \quad \forall t \in T, \forall v \in V \tag{5}$$

2.4. Objectives

The first objective function includes the following terms:

(Term 1) Location allocation costs: The fixed cost for opening DC_j with capacity level n is given by:

$$\sum_{j \in J} \sum_{n \in N_j} f_j^n U_j^n$$

(Term 2) Transportation costs: The variable routing cost and vehicle usage fix cost are given by:

$$\beta q \left(\sum_{t \in T} \sum_{v \in V} \sum_{k \in M} \sum_{l \in M} d_{kl} R_{klvt} + \sum_{j \in J} \sum_{\in M} \sum_{v \in V} \sum_{t \in T} \Pr_v R_{jlvt} \right)$$

(Term 3) Inventory costs: It consists of inventory holding costs, ordering costs and safety stock costs as given below:

$$\sum_{j \in J} \left[\sqrt{2\theta \sum_{k \in K} \sum_{p \in P} \sum_{t \in T} h_{jpt} (\theta P_j + \beta g_j) \mu_{kpt} Y_{jk}^2} + \beta \sum_{p \in P} \sum_{t \in T} a_{jp} \sum_{k \in K} \mu_{kpt} Y_{jk}^2 \right.$$
$$\left. + \theta \sum_{p \in P} \sum_{t \in T} h_{jpt} z_\alpha \sqrt{lt_{jt} \sum_{k \in K} \delta_{kpt}^2 Y_{jk}^2} \right]$$

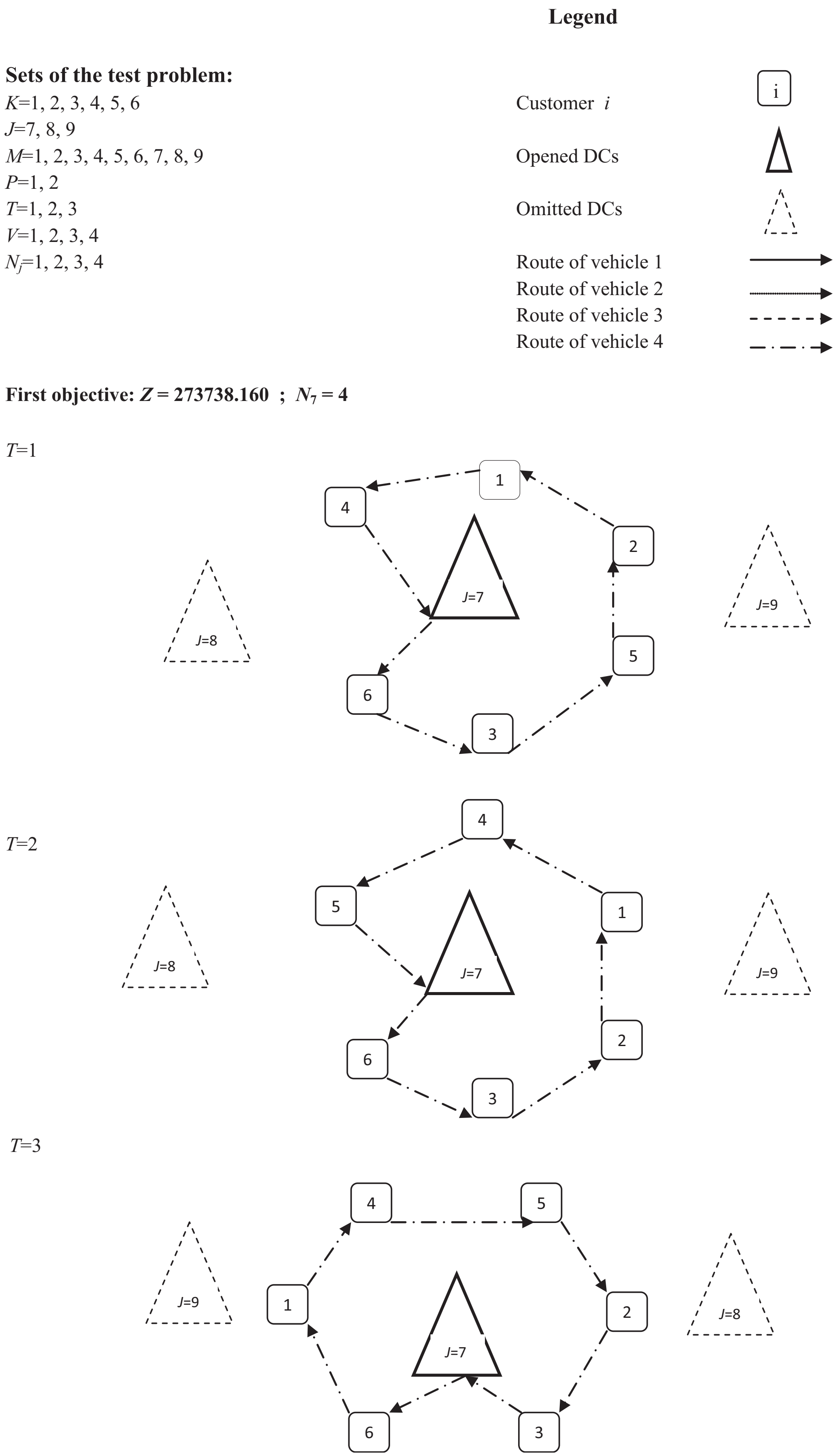

Fig. 1. Solutions for the first objective in different periods.

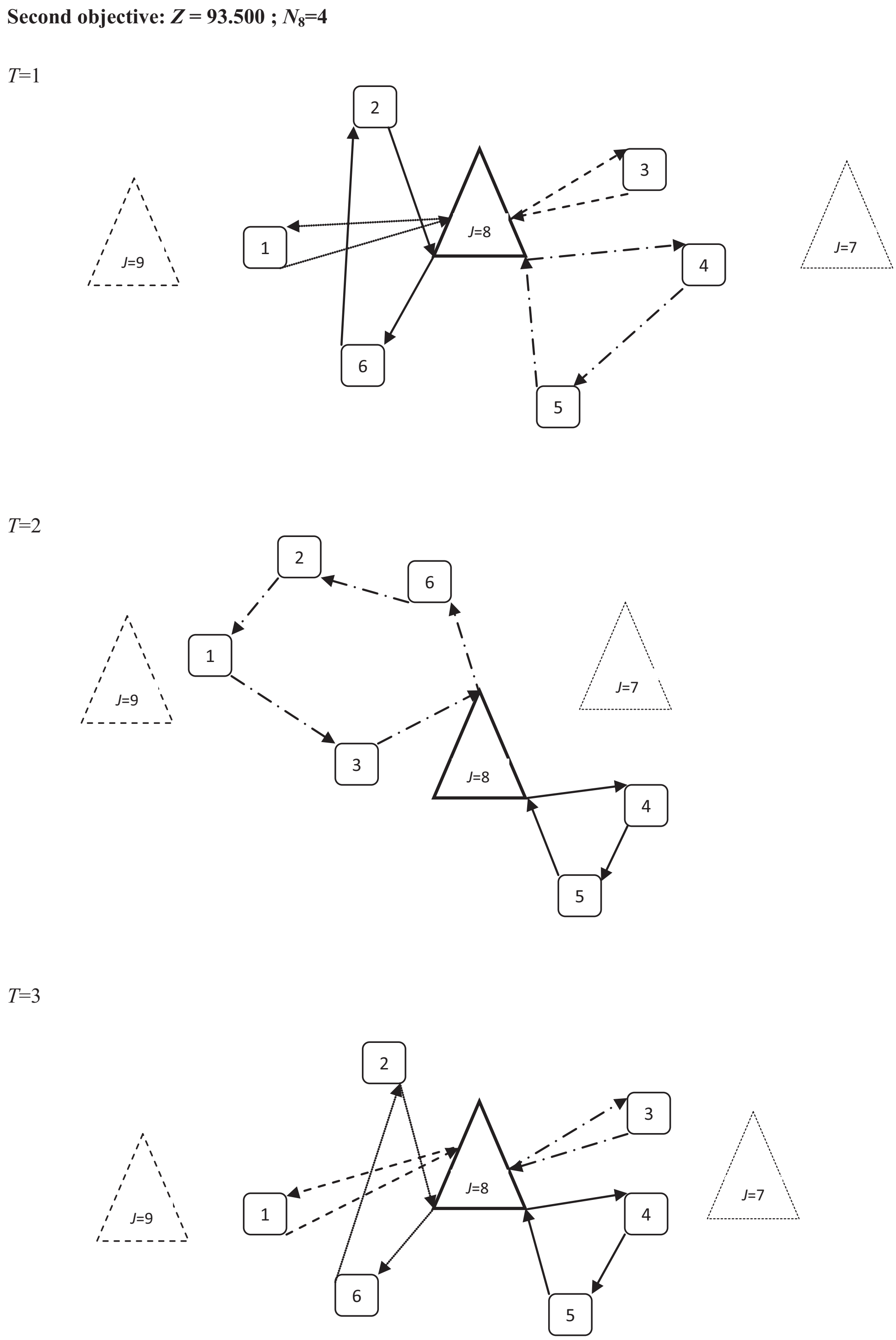

Fig. 2. Solutions for the second objective in different periods.

The second objective function minimizes the maximum mean time for delivering to customers, given as Eq. (16):

$$\text{Min Max}\left\{E\left[\sum_{k\in M}\sum_{l\in K} t_{kl}R_{klvt}\right]\right\} \quad (\forall v \in V, t \in T) \tag{16}$$

It is known that customer delivery time increases along with the tour (from the first customer to the last customer in specific tour). Thus, to decrease the maximum mean customer delivery time, it is better to minimize a delivery time of the last customer in a specific tour. The nonlinear equation can be transformed into the linear one. As a result, the deterministic equations are presented in Eqs. (3) and (4) respectively. The optimal value of the order size at distribution center j for product p in period t is shown by:

$$Q^*_{jpt} = \sqrt{\frac{2(\theta P_j + \beta g_j)\sum_k \mu_{kpt} Y_{jk}}{\theta h_{jpt}}} \tag{17}$$

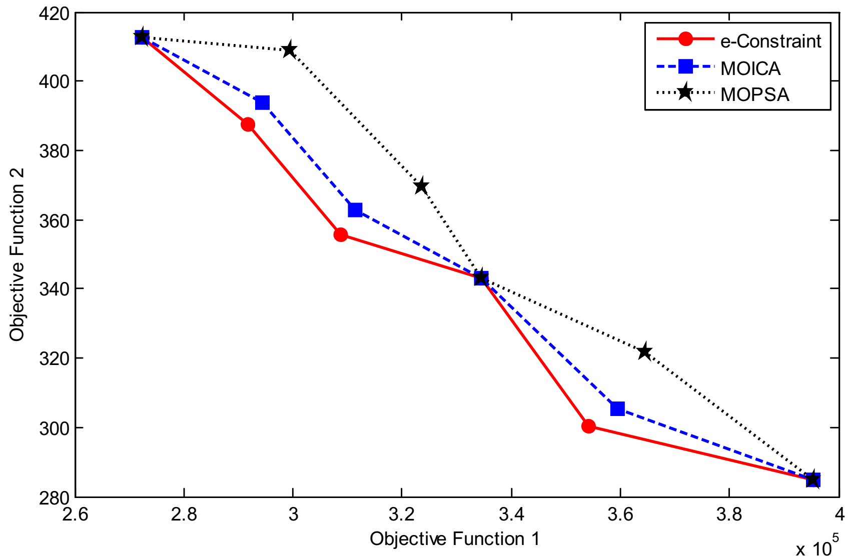

Fig. 3. Comparison between Pareto results from the proposed MOICA and MOPSA with the ϵ-constraint method.

2.5. Constraints

Constraint (4) ensures that each customer stand on just one vehicle route in each period. Constraint (5) is the vehicle capacity constraint. This constraint ensures that the total delivery to each customer in each period is less than or equal to vehicle capacity. Constraint (6) is sub-tour elimination constraint and ensures that no route is apart from the distribution center node. Constraint (7) guarantees that if a vehicle enters a node in a time period, it should leave that node in that period. Constraint (8) ensures that a route contains a distribution center node in a time period. Constraint (9) states that a customer can be allocated to a distribution center in each time period only if there is a route passed by that customer and originated from that distribution center. Constraint (10) states that each distribution center could only use from one of its capacity levels if it was opened. Constraint (11) ensures that total delivery to all customer of vehicle v is less than or equal to vehicle capacity. Constraint (12) states that we do not have any route between distribution centers. Constraint (13) states that each customer can be assigned to only one distribution center through a planning horizon. Constraint (14) ensures the integrality of the binary variables, and Constraint (15) enforces the non-negativity restrictions on the other variables.

2.6. Validation of the model

To explore the properties of the problem, we solve the presented bi-objective model for the given small test problem. We consider this model as two single-objective problems separately and solved them by the GAMS software using Baron Solver for a numerical example. Afterward, optimum solutions are obtained for both crisp single-objective models. The results show there is a conflict between two objectives as shown in Figs. 1 and 2.

Furthermore, we can assert that there exists a contradiction between two objective functions. It means that the units of our two objectives are cost and time respectively. Logically, if we incur more cost we have a less time and vice versa. In this way, we interpret the contradiction existing between these two objectives.

To validate the proposed algorithms, we compare the Pareto solutions of each algorithm with the non-dominated solutions achieved by the -constraint method. It can be seen in Fig. 3 that the solutions obtained by the proposed MOICA are clearly closer to the achieved by the MOPSA. So, the proposed algorithms are reliable sufficiently to be used for larger-scale problems.

0.2	0.8	0.1	0.9	0.3	0.6	0.1	0.8

Generating a number for each cell

6	3	7	1	5	4	8	2

Priority of each cell

Fig. 4. Customer sequence.

0.2	0.3	0.5	0.1	0.0

Fig. 5. Number of the used vehicles.

3. Proposed algorithm

Obviously, solving the NP-hard problems in a reasonable time is very difficult. As the location-routing-inventory problem is known as an NP-hard problem Ahmadi Javid and Azad (2010), we proposed and used four multi-objective handling meta-heuristic algorithms for solving the given problem. In this paper, a multi-objective imperialist competitive algorithm (MOICA) is proposed to solve our multi-objective mathematical model. Additionally,

0.2	0.4	0.3	0.1	0.5

Generating a number for each cell

4	2	3	5	1

Priority of each cell

Fig. 6. Used vehicle.

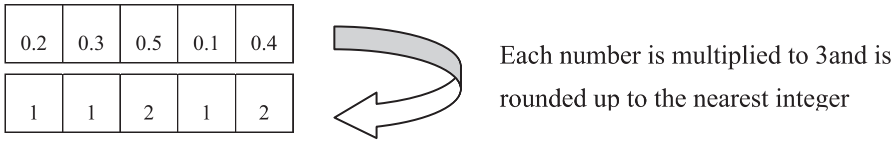

Fig. 7. Distribution centers indices.

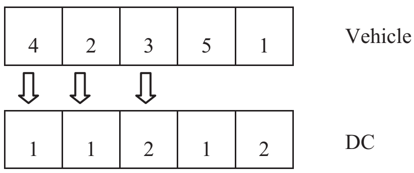

Fig. 8. Assignment of vehicles from distribution centers (DCs).

Section 4 presents three well-known multi-objective evolutionary algorithms, such as NSGA-II (non-dominated sorting genetic algorithm II), MOPSA (multi-objective parallel simulated annealing) and PAES (Pareto archived evolution strategy) to compare the numerical results generated by the proposed MOICA.

3.1. Solution representation

3.1.1. Customer sequence

To represent the solution in the algorithm, the arrays called chromosome in the genetic algorithm are used. Each array consists of a number of genes representing customers. The first array is the $1 \times K$ array, in which K random numbers (say $K = 8$) related to the customer number in the range of (0, 1) are generated, which the larger corresponds to the higher priority. This is shown in Fig. 4.

3.1.2. Number of vehicles used

A $1 \times V$ array is created, in which each cell is corresponded to a random number falling in the (0, 1) interval. Therefore, the place of the largest number indicates the number of vehicles used in the problem. The sample array with five genes ($V = 5$) is depicted in Fig. 5. It shows that three vehicles are needed to be used because the maximum number is placed in the third position.

3.1.3. Used vehicles

A $1 \times V$ array is created such that each cell is corresponded to a random number falling in the (0, 1) interval. Thus, the larger number equals to the higher priority specifies that vehicles should be used. For instance, three selected vehicles in the sample are the first on the left side of the chromosome and are shown in the Fig. 6. This means that vehicles 2, 3 and 4 should be used.

3.1.4. Distribution centers indices

Like above, a $1 \times V$ array is created such that each cell is corresponded to a random number falling in the (0, 1) interval. Each number is multiplied to the number of potential distribution center (J) while it is rounded up to the nearest integer. E.g. regarded to the Fig. 7, it is assumed that three distribution centers exist and the numbers are multiplied by 3 ($J = 3$). Consequently, the selected vehicles of 2, 3 and 4 are suggested to be allocated to the distribution centers 1, 2 and 1, respectively (see Figs. 4, 5 and 8).

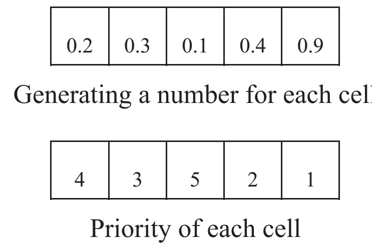

Fig. 10. Customer indices.

3.1.5. Distribution centers capacity levels

Likewise, an array of $1 \times J$ is created where each cell is corresponded to a random number falling in the (0, 1) interval. The values of gens (or cells) are multiplied by the number of capacity levels available for distribution center (N_j) while it is rounded up to the nearest integer. In the discussed example, three distribution centers' number are multiplied by 4 ($N_j = 4$) and are rounded up to the represented capacity level. Therefore, the first and forth capacity levels are attributed to the distribution centers 1 and 2, respectively (Fig. 9).

3.1.6. Customers tour formation

Finally, an array of $1 \times V$ is created where each cell is corresponded to a random number falling in the (0, 1) interval that the larger number represents higher priority. This array forms the customer index chromosome. As it is assumed in the discussed example, the chromosome of five vehicles has led into a layout of three uses, because a number of used vehicles are three. Consequently, we choose the first three numbers of the array. Accordingly, the first three numbers of the customers' priority array shown in Fig. 10 are chosen for further evaluation.

Then, the maximum number (i.e., 5) is replaced to the total number of customers (i.e., 8) followed by an ascending order. As a result, three cuts of customer sequence chromosome are obtained showing three tours illustrated in Figs. 11 and 12 respectively. Finally, in the customer sequence chromosome, three tours from positions 3 and 4 are formed (see Fig. 11). Therefore, this test problem is depicted in Fig. 12.

To develop the current model to a multi-period network, it is necessary to define different customer sequences independently. For each period, we have different customer sequences, in which

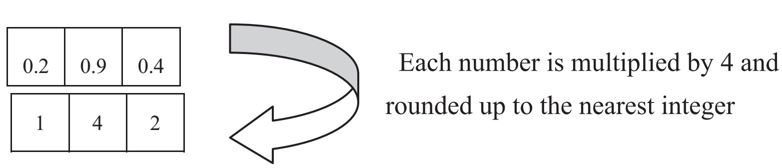

Fig. 9. Capacity levels of distribution centers.

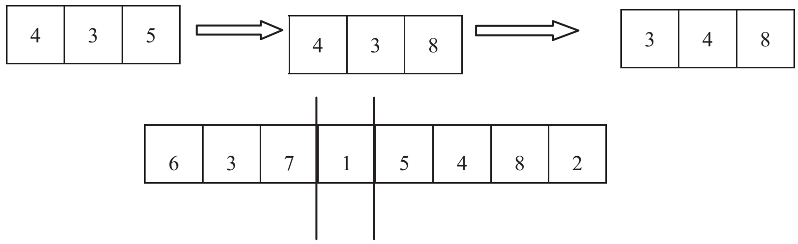

Fig. 11. Customer tours.

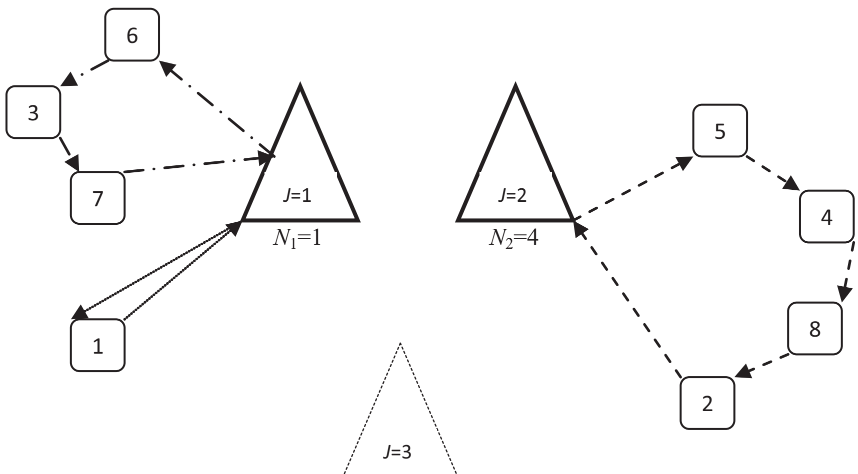

Fig. 12. Solution for the test problem.

location decisions and number of customers, number of potential distribution centers, number of vehicles and number of distribution levels are fixed as well as routing decisions, which are modified in different periods. Because, the presented model is a multi-period model, thus for this model we should generate chromosomes equal to a number of periods.

3.2. Multi-objective imperialist competitive algorithm

A multi-objective imperialist competitive algorithm (MOICA) is considered as a multi-objective meta-heuristic evolutionary algorithm. Evolutionary optimization algorithms mostly inspire natural processes model and other aspects of species evolution in their method. However, the imperialist competitive algorithm (ICA) uses human social evolution as a source of inspiration for developing a strong optimization strategy (Atashpaz-Gargari & Lucas, 2007).

3.2.1. Initializing empires

The first step in the MOICA is creating the initial solutions. Each form of solution in the imperialist competitive algorithm (ICA) is like a matrix called "country". Each matrix consists of variable values (cells) to be optimized. In a N-dimensional optimization problem, a country is a $1 \times N$ matrix that have N cells, such as country = $[p_1, p_2, p_3, \ldots, p_N]$. Each p_i in a country states a socio-political trait of a country, such as culture, language and economical policy (Fig. 13). Therefore, the algorithm searches for the best country which is the country with the best culture, language and economical policy (Atashpaz-Gargari & Lucas, 2007).

After generating countries, a non-dominance technique and a crowding distance metric are used to rank and select the population fronts and the members of front one that are saved in an archive. Then, the best solutions in terms of the non-dominance and crowding distance are selected from the population's front one as the imperialists. The remaining countries are considered as colonies. For calculating the cost value of each imperialist, the value of each objective function is obtained for each imperialist. Then, the cost value of each objective function is calculated by:

$$Cost_{i,n} = \frac{|f^p_{i,n} - f^{p,best}_i|}{f^{p,\max}_{i,total} - f^{p,\min}_{i,total}} \tag{18}$$

where $Cost_{i,n}$ is the normalized value of the objective function i for imperialist n, and $f^p_{i,n}$ is the value of the objective function i for imperialist n. $f^{p,best}_i$ are the best values for the objective function i in each iteration. $f^{p,\max}_{i,total}$ and $f^{p,\min}_{i,total}$ are the maximum and minimum values for the objective function i in each iteration, respectively. Finally, the normalized cost value of each imperialist ($Cost_n$(is calculated by summation of the normalized value for all objective functions in this way:

$$Cost_n = \sum_{i=1}^{r} Cost_{i,n} \tag{19}$$

where r is the number of objective functions.

The power of each imperialist is calculated after acquiring normalized value of objective functions, so that the colonies are arranged between the imperialists according to the power of each imperialist country.

$$P_n = \left| \frac{Cost_n}{\sum_{i=1}^{N_{imp}} Cost_i} \right| \tag{20}$$

Then, the initial number of colonies of an empire is calculated by Eq. (21):

$$NC_n = round\{P_n \times N_{col}\} \tag{21}$$

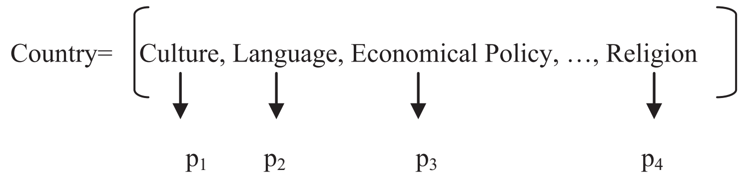

Fig. 13. Candidate solutions of the problem, called country.

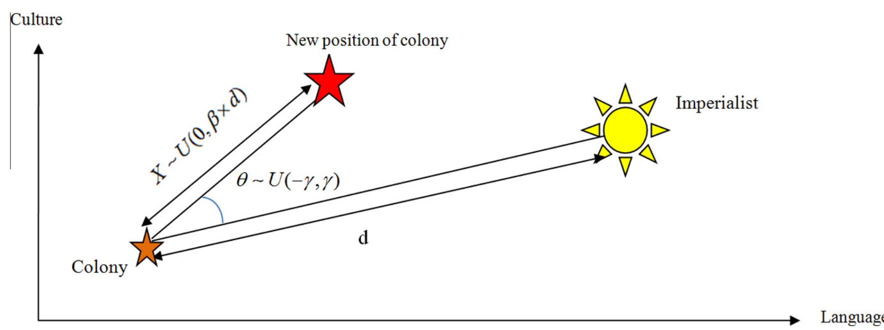

Fig. 14. Moving colonies toward the imperialists with a random angle θ.

where NC_n is the initial number of colonies in the n-th imperialist, and N_{col} is the number of all colonies. We select NC_n of colonies and assign them to the imperialists, respectively. Finally, the more powerful imperialist takes a greater number of colonies while imperialist with weaker power has less number.

3.2.2. Total power of an empire

The total power of an empire is mostly influenced by the power of the imperialist country while, the power of the colonies of an empire has an insignificant effectiveness on the total power of that empire. As a result, the equation of the total power of an empire is represented by Eq. (22).

$$TP\ of\ Emp_n = Total\ Cost(\mathrm{Imperialist}_n) + \xi mean\{Total\ Cost\ (Colonies\ of\ Empire_n)\} \qquad (22)$$

where $TPEmp_n$ is the total power of the n-th empire and ζ (zeta) is a positive number, which is considered to be less than 1 plus the total cost of imperialists and colonies calculated by Eqs. (18) and (19). The efficacy of colonies on an empire's cost is related to the value of ζ. For greater value of ζ, colonies are more effective for the total cost of an empire and vice versa.

3.2.3. Assimilation: movement of colonies toward the imperialist

First, colonies are divided between imperialists, and then the imperialists made their colonies to move toward themselves along different socio-political axis, such as culture, language and religion. This absorption is shown in Fig. 14, in which the colony is absorbed by the imperialist in the culture and language axes. In this figure, d is the distance between colony and imperialist. x is the amount of movement of colony toward the imperialist that follows a uniform distribution between 0 and $\beta \times d$, and β is a number greater than 1. Angle is defined for moving colony toward imperialist in other directions and follows a uniform distribution between $-\gamma$ and γ.

It has been estimated that $\pi/4$ is the most proper value of γ.

The best solution in each population group is chosen according to the non-dominance strategy and crowding distance metric.

3.2.4. Crossover between colonies

Pursuing the crossover method, colonies share their information between themselves by crossover. Colonies with a better cost are more fortunate for being crossover under a binary tournament selection. The population percent sharing information is shown by p-Crossover.

3.2.5. Revolution

In each decade, in each imperialism, the number of colonies will be revolted; in other words, socio-political characteristics of a country change suddenly. The revolution policy consists of creating a new solution. It is similar to the mutation operator in the GA. The revolution rate in this paper is shown by $P_{\mathrm{Revolution}}$.

3.2.6. Archive adaption

Pareto solutions of the MOICA are saved in an archive as ranking and sorting done by the non-dominated sorting and crowding distance colonies of a merged population, the members of front one are saved and, the others are deleted after ranking and sorting solutions in the archive. Furthermore, the size of archive is equal to n archives.

3.2.7. Exchanging positions of the imperialist and a colony

The total cost of each imperialism is up-to-dated. Then each imperialist with the best colony in term of the crowding distance in front one of the related colonies are compared together. If the cost of a colony is better than its related imperialism, that colony will be selected as an empire for their related imperialism.

3.2.8. Imperialistic competition

In the imperialistic competition, more precisely, the power of the weaker emprise will reduce and the power of the stronger ones will increase. Therefore, the strongest imperialists try to pick other colonies of other less powerful imperialists. The imperialistic competition is done by grabbing some of the weakest colonies of the weakest empires by the more powerful empires during a competition relaying on their possession probability by Eq. (23). This means that there are not any insurance that powerful empires definitely possess these colonies; however, this is more probable that these empires possess them.

$$NTP\ Emp_n = \max\{TP\ Emp_i\} - TP\ Emp_n \qquad (23)$$

where $NTP\ Emp_n$ is the normalized total power of the n-th empire, and $TP\ Emp_n$ is the total power of the n-th empire. Next, the

```
NFC ← 0
set the parameters of HMOICA (n-Pop, N-imp,ξ,β, P-Assimilation, P-Crossover, P-Revolution, n-Archive)
Generate the initial countries (Randomly) ←n-Pop
Evaluate fitness of each country
update the NFC
Form the initial empires:
    a)  Choose most powerful countries as the imperialists←N-imp
    b)  Assign other countries to imperialists based on imperialist power (pop1)
terminate ←false
while (terminate = false) do at each Imperialist
        Move the colonies of an empire toward the imperialist (pop2) ←P-Assimilation (P_A), β
        Crossover some colonies with empire (pop3) ←P-Crossover (P_C)
        Revolution among colonies (pop4) ←P-Revolution (P_R)
        Evaluate fitness of each country
        Update the NFC
        Merge all created population
        Update colonies
        Update archive ←n-Archive
        if (Cost of colony is lower than its own Empire) then
            Exchange the positions of the imperialist and a colony
        end
            Calculate the total power of the empires ←ξ
            Perform imperialistic competition
            Eliminate the powerless empires (the imperialist with no colony)
        if (NFC = predefined value) then
            terminate = true
        end if
end while
```

Fig. 15. Pseudo code of the proposed MOICA.

possession probability of each empire is calculated by the following equation:

$$P_{pn} = \left| \frac{NTP\ Emp_n}{\sum_{i=1}^{N_{imp}} NTP\ Emp_i} \right| \tag{24}$$

Then, the roulette wheel method is used for assigning the mentioned colony to one of the empires.

3.2.9. Eliminating the powerless empires

Powerless empires that cannot develop their colonies will be eliminated and grabbed with stronger imperialism. They will be collapsed and their colonies will be distributed among other empires in the imperialistic competition. This process will be continued until it remains only one emperor.

3.2.10. Stopping criteria

The last step of imperialistic competition is considered when all powerless empires distributed between powerful empires and there is only one empire between all of the countries. The Pseudo code of the proposed MOICA is presented in Fig. 15.

4. Computational results

In this section, we try to evaluate the efficiency of the suggested MOICA with three famous multi-objective evolutionary algorithms (EAs), namely NSGA-II (non-dominated sorting genetic algorithm II) and PAES (Pareto archived evolution strategy) and MOPSA (multi-objective parallel simulated annealing). All algorithms studied in this paper are coded using MATLAB software and run on a personal computer with 2.66 GHZ CPU and 4 GB main memory (Fig. 16)

4.1. Comparative algorithms

This section presents the comparative algorithms (i.e., NSGA-II, MOPSA and PAES) in order to compare the numerical results generated by the proposed MOICA.

4.1.1. Non-dominated sorting genetic algorithm II

The non-dominated sorting genetic algorithm II (NSGA-II) proposed by Deb, Agrawal, Pratap, and Meyarivan (2000), Deb, Pratap, Agarwal, and Meyarivan (2002) was one of the famous multi-objective evolutionary algorithms. A non-dominance technique and a crowding distance are performed for ranking and selecting the population fronts.

The current population and new generated solutions are combined together. Then the best solutions in terms of non-dominance and crowding distance are selected from the combined population. In this proposed NSGA-II, crossover and mutation operators to generate new solutions are used.

Non-dominance technique: A multi-objective model has n objective functions, solution x_1 and x_2 are placed in same front when do not dominate each other, in which x_1 dominate x_2 if:

- For all the objective functions, solution x_1 is not worse than another solution x_2.
- For at least one of the n objective functions, x_1 is exactly better than x_2.

Front one consists of all solutions that are not dominated by any other solutions. The second front is made by all solutions that only dominated by solutions in front number one.

Crowding distance: It proposes an estimate of the density of solutions surrounding a particular solution. The solutions with a lower value of crowding distances are preferred over solutions with a higher value of the crowding distance. The crowding distance used in the NSGA-II is computed by:

$$CD_i = \sum_{k=1}^{r} \frac{f^p_{k,i+1} - f^p_{k,i-1}}{f^{p,\max}_{k,total} - f^{p,\min}_{k,total}} \tag{25}$$

where r is the number of objective functions, $f^p_{k,i+1}$ is the k-th objective function of the $(i+1)$-th solution and $f^p_{k,i-1}$ is the k-th objective function of the $(i-1)$-th solution after sorting the population according to crowding distance of the k-th objective function. In

```
NFC ← 0
set the parameters of PSA (nPop, nMutate, pCrossover, α, β, T0)
create initial solution ←nPop
terminate ←false
while (terminate = false) do
mutate each initial solution ←nMutate
find the best solution (imperialist) ← non-dominated sorting & crowding distance
update NFC
assimilate mutated solutions (colonies) toward imperialist ←β
merge whole new created solution
apply crossover ←pCrossover
update NFC
find better new solutions ←nPop
update NFC
if (new solution dominates old solution) then
accept the new solution
else if (no one dominate the other one) then
calculate crowding distance- CD of each solution
if (CD of new solution > CD of old solution) then
accept the new solution
end if
else
apply probable acceptance function ←P = e^(−Δf/T) ←Δ& T
create random value r
if (r<P) then
accept new solution
else
accept old solution
end if
end if
update NFC
T←α × T
if (NFC = predefined value) then
terminate = true
end if
end while
```

Fig. 16. Pseudo code of the proposed MOPSA.

addition, $f^{p,\max}_{k,total}$ and $f^{p,\min}_{k,total}$ are the maximum and minimum values of objective function k, respectively.

Tournament selection operator: Selecting two solutions of the population size is known as the first step of the tournament selection mechanism. This procedure includes both the crossover and mutation operators. If two populations are from different fronts, the lowest front number is selected and if they are belong to the same front, the solution with the highest crowding distance is selected.

4.1.2. Pareto archived evolution strategy (PAES)

The Pareto archived evolution strategy (PAES) is a meta-heuristic algorithm for solving multi-objective models (Corne, Knowles, & Oates 2000; Knowles & Corne 1999). PAES in a simple method uses a simple (1 + 1) local search evolution strategy in order to find varied solutions in the Pareto optimal set. In the first step, this algorithm begins with the initialization of a single solution, which is appraised using the multi-objective cost function. In each iteration, this algorithm uses a mutation operator and produce a new solution. Then, the new and current solutions are compared together based on the method proposed by Knowles and Corne (1999). Subsequently, the new solution is updated and archived. This process is resumed to finish the iteration numbers of the algorithm.

4.1.3. Multi-objective parallel simulated annealing

The simulated annealing (SA) algorithm starts with an initial feasible solution and through search over its neighbourhood by means of three methods (e.g., mutation, assimilation and crossover) to find the best objective function. Thus, the well-suited objective function is dependent on the selected initial solution that limits its efficiency. Alternatively, the multi-objectives parallel simulated annealing (MOPSA) algorithm is originated by several initial solutions simultaneously. This reduces the number of iterations and running time. Fig. 16 shows the Pseudo code of the proposed MOPSA.

Probability of acceptance new solution: In the SA process, if the new result does not meet the better solution, a probability will be considered for its acceptance. This is defined as follows; otherwise, it will be selected as a new solution.

$$P = e^{\frac{-\Delta f}{T}} \tag{26}$$

where T is the temperature as a controllable parameter, and Δf is derived from the following formula:

$$\Delta f = f(x') - f(x) \tag{27}$$

where x and x' are the old and new solution, respectively.This equation is inserted in the proposed MOPSA as follows:

$$\Delta = \left| \frac{(f_1(x) - f_1(x'))}{f_1(x)} + \frac{(f_2(x) - f_2(x'))}{f_2(x)} \right| \tag{28}$$

Annealing program: The annealing program consists of key parameters, such as starting temperature (T_0), final temperature (T_f) and temperature reduction factor (α). Choosing these factors properly facilitates the convergence of the algorithm.

As iteration continues, the previous temperature should be decreased considering Eq. (29) to reduce the probability of accepting incompatible solutions and reach the optimum result.

$$T_{n+1} = \alpha . T_n \quad (29)$$

Pareto solutions of the proposed algorithm: In order to find Pareto solutions, each solution of iteration should be compared in pairs in order to calculate the non-dominance ranking. The solutions that are not dominated with any others are introduced as Pareto solutions.

Stopping criteria: In the proposed MOPSA, the number of function calls (NFCs) is considered as the stopping criteria.

4.2. Parameter values

To evaluate the model, we solve 30 test problems. Additionally, the following parameters in these test problems have different values, but in a defined tolerance.

$d_{kl} \sim$ Uniform(0, 300) $\quad \sigma^2_{lpt} \sim Uniform(10, 30)$
$h_{jpt} \sim$ Uniform(5, 10) $\quad g_j \sim$ Uniform(10, 15)
$p_j \sim$ Uniform(10, 15) $\quad a_{jp} \sim$ Uniform(5, 10)
$lt_{jt} \sim$ Uniform(6.365, 10, 365)
$\mu_{lpt} \sim$ Uniform(400, 1500) $\quad Vc_v \sim Uniform([\frac{D}{V}], 2[\frac{D}{V}])$
$D = \sum_k \sum_p \sum_t \mu_{kpt}$

where D represents the total average of the customers' demands for all kinds of product over time horizon, and $|V|$ is the number of vehicles.

Each vehicle meets 100 customers in each time period. The fill rate is 97.5%, so Z_α is equal to 1.96. Other parameters are defined as follows:

$$E(t_{kl}) = d_{kl}/2$$

$$Pr_v = 1500 + 0.1 \times Vc_v$$

$$\alpha = 0.003$$

$$\beta = 0.7$$

We assume four capacity levels for each distribution center, as shown below.

$$b_j^1 = [cap(j)], b_j^2 = [1.5cap(j)], b_j^3 = [2cap(j)], b_j^4 = [2.5cap(j)]$$

where $cap(j) = c_j \times \frac{D}{j}$ and $c_j \sim Uniform(0.8, 1.2)$ and $|j|$ is the number of distribution centers.

Therefore, the following four cost levels are considered for using each distribution center.

$$f_j^1 = [0.65 \times k'_j], f_j^2 = [0.9 \times k'_j], f_j^3 = [1.1 \times k'_j], f_j^4 = [1.35 \times k'_j]$$

where $k'_j \sim Uniform(300, 450)$

4.3. Parameter tuning

To evaluate the performance of the algorithm, we set the assigned value of the algorithm parameters. Parameter values have a very significant effect on the efficiency of the algorithm. If these values are not set truly, getting the proper result will become difficult. To tune the parameters of algorithms, we consider two different sets of problems, namely small sizes (i.e., problem numbers 1–15) and large sizes (i.e., problem number 16–30). To determine the values of parameters, we use the response surface methodology (RSM) (Zandieh, Dorri, & Khamseh, 2009).

In this section, each algorithm is run for different combinations of parameters and their levels. In the RSM methodology, we need an index in order to compare different combinations of parameters. Because the model presented in this paper is a bi-objective, we should use a unique index (e.g., quality index). In this method, all the Pareto solutions getting from different parameter combinations are considered together, and non-dominance operation is done for all of them simultaneously.

4.3.1. MOPSA

In this paper, the NFC is set to 30,000 and 100,000 for small and large-sized problems, respectively. Table 1 shows the parameters and their levels for small and large sizes in the MOPSA algorithm. In addition, Table 2 shows the tuned value parameters of the proposed MOPSA.

4.3.2. NSGA-II

- The initial population sizes for small and large-sized problems are set 200 and 300, respectively.
- The mutation and crossover rates are set 0.2 and 0.8, respectively.
- The stopping criterion is the NFC and is set with the value of 30,000 and 100,000 for small and large-sized instance, respectively.

4.3.3. PAES

- The archive size is assumed to be 150.
- The stopping criterion is the NFC considering 30,000 and 100,000 for small and large-sized instance, respectively.

Table 1
Parameters and their levels in the MOPSA algorithm for small (S) and large (L) sizes.

Factor	Problem size											
	S	L	S	L	S	L	S	L	S	L	S	L
	T_0		α		*n-Move*		*n-Pop*		*p-Crossover*		β	
Lower limit	5	5	0.75	0.75	5	8	3	5	0.4	0.4	1	1
Upper limit	15	15	0. 95	0.95	15	20	8	12	0.8	0.8	3	3

Table 2
Tuned value parameters of the proposed MOPSA.

Factor	Problem size											
	S	L	S	L	S	L	S	L	S	L	S	L
	T_0		α		*n-Move*		*n-Pop*		*p-Crossover*		β	
Tuned Value	10	13	0. 84	0.91	10	16	5	6	0.5	0.7	1.8	2

Table 3
Parameters and their levels in the MOICA algorithm for small (S) and large (L) sizes.

Factors	Coded level					
	−1		0		+1	
	S	L	S	L	S	L
n-Pop	100	150	150	225	200	300
N-imp	4	8	6	10	8	12
P_A	0.3	0.4	0.5	0.6	0.7	0.8
P_C	0.2	0.3	0.4	0.5	0.6	0.7
P_R	0.1	0.2	0.2	0.3	0.3	0.4
	0.1	0.1	0.15	0.15	0.2	0.2
β	1	1	2	2	3	3

Table 4
Tuned value parameters of the proposed MOICA.

Factors	Optimal coded value		Optimal real value	
	S	L	S	L
n-Pop	0.85	1	193	300
N-imp	−0.2	−1	5	8
P_A	0.18	0.2	0.54	0.64
P_C	1	0.5	0.6	0.6
P_R	−0.8	0.19	0.12	0.32
ξ	0.9	−0.5	0.195	0.125
β	−0.2	0.15	1.8	2.15

Table 5
Default value for different parameters in each test problem.

Parameters	Default values
Number of products	2, 3, 5
Number of periods	3, 5, 7
Number of customers	10, 15, 30, 50, 70, 100
Number of distribution centers	5, 10, 15
Number of vehicles	3, 4

4.3.4. MOICA

Table 3 shows the parameters and their levels for small and large sizes in the MOICA algorithm. In addition, Table 4 shows the tuned value parameters of the proposed MOICA.

4.4. Comparison metric

To validate the proposed MOICA, four comparison metrics are taken into account.

4.4.1. Quality metric (QM)

This method puts the non-dominated solutions acquired by the algorithms together and calculates the percentage of the Pareto solution belonging to each algorithm. Having a higher value is known as a favourable scale for the solutions of this metric (Moradi, Zandieh, & Mahdavi, 2011).

4.4.2. Mean ideal distance (MID)

The distance between the best solutions and Pareto solutions is calculated by the MID, as given below.

$$MID = \frac{\sum_{i=1}^{n}\sqrt{\left(\frac{f_{1i}-f_1^{best}}{f_{1,total}^{\max}-f_{1,total}^{\min}}\right)^2+\left(\frac{f_{2i}-f_2^{best}}{f_{2,total}^{\max}-f_{2,total}^{\min}}\right)^2}}{n} \tag{30}$$

where n is the number of non-dominated solutions, $f_{i,total}^{\max}$ is the maximum value of i-th fitness function among all non-dominated solutions obtained by the algorithms,$f_{i,total}^{\min}$ is the minimum value of i-th fitness functions among all non-dominated solutions obtained by the algorithms and f_i^{best} is the ideal solution of i-th fitness function. In contrast to the QM, the algorithm with a lower value of the MID has a better performance.

Table 6
Different test problems with their data sets.

Problem no.	Product no.	Period no.	Customer no.	DC no.	Vehicle no.
1	2	3	10	5	3
2	3	3	15	5	3
3	5	3	30	5	3
4	2	5	50	5	3
5	3	5	70	5	3
6	5	5	100	5	3
7	2	7	10	5	3
8	3	7	15	5	3
9	5	7	30	5	3
10	2	3	50	5	3
11	3	3	70	10	3
12	5	3	100	10	3
13	2	5	10	5	3
14	3	5	15	10	3
15	5	5	30	10	3
16	2	7	50	10	4
17	3	7	70	10	4
18	5	7	100	10	4
19	2	3	15	5	4
20	3	3	15	10	4
21	5	3	30	15	4
22	2	5	50	15	4
23	3	5	70	15	4
24	5	5	100	15	4
25	2	7	10	15	4
26	3	7	15	15	4
27	5	7	30	15	4
28	2	3	50	15	4
29	3	5	70	15	4
30	5	7	100	15	4

4.4.3. Diversification metric (DM)

The spread of a Pareto solution set is calculated by the DM. A higher value in this metric represents a better performance of the algorithm.

$$DM = \sqrt{\left(\frac{\max f_{1i}-\min f_{1i}}{f_{1,total}^{\max}-f_{1,total}^{\min}}\right)^2+\left(\frac{\max f_{2i}-\min f_{2i}}{f_{2,total}^{\max}-f_{2,total}^{\min}}\right)^2} \tag{31}$$

4.4.4. Spacing metric (SM)

The uniformity of the spread of the non-dominated set solutions is calculated by the SM (Tavakkoli-Moghaddam, Azarkish, & Sadeghnejad, 2011)

$$SM = \frac{\sum_{i=1}^{n-1}|\bar{d}-d_i|}{(n-1)\bar{d}} \tag{32}$$

In Eq. (32), d_i is the Euclidean distance between consecutive solutions in the obtained non-dominated set of solutions and $\bar{d}$ is the average of these distances. As the value of this metric reduces, the algorithm shows a better performance.

4.5. Entering data and problem classes

Table 5 shows the default value for different parameters in each test problem. According to Table 5, different test problems are carried out in Table 6.

Table 7
Comparison results between the PAES, NSGA-II and MOICA algorithms according to the QM and SM.

Problem no.	Quality metric (QM)			$\bar{d}_1$	$\bar{d}_2$	Spacing metric (SM)			$\bar{d}_3$	$\bar{d}_4$
	PAES	NSGA-II	MOICA			NSGA-II	PAES	MOICA		
1	0	0.08	0.92	0.84	0.92	0.558	0.879	0.444	0.435	0.114
2	0.09	0.18	0.73	0.55	0.64	0.521	0.937	0.315	0.622	0.206
3	0	0.2	0.8	0.6	0.8	0.409	0.81	0.385	0.425	0.024
4	0	0	1	1	1	0.395	0.762	0.373	0.389	0.022
5	0	0.1	0.9	0.8	0.9	0.608	0.726	0.359	0.367	0.249
6	0	0	1	1	1	0.359	0.932	0.282	0.65	0.077
7	0	0.15	0.85	0.7	0.85	0.608	0.761	0.274	0.487	0.334
8	0	0	1	1	1	0.595	0.816	0.335	0.481	0.26
9	0	0.25	0.75	0.5	0.75	0.87	0.866	0.303	0.563	0.567
10	0	0.1	0.9	0.8	0.9	0.672	0.768	0.243	0.525	0.429
11	0	0	1	1	1	0.574	0.893	0.586	0.307	−0.012
12	0	0	1	1	1	0.862	0.845	0.303	0.542	0.559
13	0	0	1	1	1	0.674	0.745	0.394	0.351	0.28
14	0.08	0.17	0.75	0.58	0.67	0.624	0.935	0.476	0.459	0.148
15	0.06	0.2	0.74	0.54	0.68	0.87	0.73	0.314	0.416	0.556
16	0	0	1	1	1	0.517	0.788	0.521	0.267	−0.004
17	0	0	1	1	1	0.657	0.771	0.442	0.329	0.215
18	0	0.12	0.88	0.76	0.88	0.601	0.859	0.276	0.583	0.325
19	0	0	1	1	1	0.552	0.727	0.486	0.241	0.066
20	0	0	1	1	1	0.562	0.895	0.61	0.285	−0.048
21	0	0	1	1	1	0.772	0.819	0.636	0.183	0.136
22	0	0	1	1	1	0.703	0.812	0.481	0.331	0.222
23	0	0.04	0.96	0.92	0.96	0.641	0.835	0.423	0.412	0.218
24	0	0.1	0.9	0.8	0.9	0.774	0.686	0.359	0.327	0.415
25	0	0	1	1	1	0.531	0.808	0.224	0.584	0.307
26	0	0	1	1	1	0.724	0.767	0.512	0.255	0.212
27	0	0.05	0.95	0.9	0.95	0.721	0.866	0.335	0.531	0.386
28	0	0	1	1	1	0.669	0.794	0.572	0.222	0.097
29	0	0	1	1	1	0.555	0.527	0.303	0.224	0.252
30	0	0	1	1	1	0.621	0.862	0.394	0.468	0.227
Average	0.007667	0.058	0.934333	0.876333	0.926666	0.626633	0.807367	0.398667	0.4087	0.227966

Table 8
Comparison results between the PAES, NSGA-II and MOICA algorithms according to the DM and MID.

Problem no.	Diversity metric (DM)			$\bar{d}_1$	$\bar{d}_2$	Mean ideal distance (MID)			$\bar{d}_3$	$\bar{d}_4$
	PAES	NSGA-II	MOICA			PAES	NSGA-II	MOICA		
1	0.929	0.872	1.379	0.507	0.45	0.862	0.569	0.255	0.314	0.607
2	0.606	1.065	1.143	0.078	0.537	0.698	0.673	0.258	0.415	0.44
3	0.617	0.772	1.3	0.528	0.683	0.795	0.743	0.302	0.441	0.493
4	0.667	0.805	0.971	0.166	0.304	0.893	0.621	0.48	0.141	0.413
5	0.86	0.758	1.111	0.353	0.251	0.622	0.797	0.317	0.48	0.305
6	0.893	0.754	1.358	0.604	0.465	0.776	0.527	0.282	0.245	0.494
7	0.859	1.048	1.296	0.248	0.437	0.724	0.596	0.245	0.351	0.479
8	0.78	0.932	1.38	0.448	0.6	0.793	0.653	0.519	0.134	0.274
9	0.819	0.92	1.228	0.308	0.409	0.679	0.518	0.312	0.206	0.367
10	0.718	0.758	0.917	0.159	0.199	0.828	0.718	0.239	0.479	0.589
11	0.898	1.042	1.325	0.283	0.427	0.899	0.667	0.43	0.237	0.469
12	0.675	0.949	1.367	0.418	0.692	0.656	0.659	0.227	0.432	0.429
13	0.875	0.84	1.24	0.4	0.365	0.835	0.749	0.385	0.364	0.45
14	0.673	0.905	1.279	0.374	0.606	0.658	0.758	0.203	0.555	0.455
15	0.771	1.033	1.212	0.179	0.441	0.898	0.737	0.372	0.365	0.526
16	0.793	1.022	1.24	0.218	0.447	0.673	0.606	0.447	0.159	0.226
17	0.648	0.724	1.098	0.374	0.45	0.895	0.655	0.282	0.373	0.613
18	0.836	0.86	1.084	0.224	0.248	0.638	0.636	0.245	0.391	0.393
19	0.69	0.911	1.394	0.483	0.704	0.669	0.481	0.319	0.162	0.35
20	0.754	0.867	0.918	0.051	0.164	0.607	0.468	0.312	0.156	0.295
21	0.833	0.963	1.343	0.38	0.51	0.782	0.496	0.239	0.257	0.543
22	0.7	0.951	1.357	0.406	0.657	0.633	0.649	0.43	0.219	0.203
23	0.716	0.817	1.298	0.481	0.582	0.722	0.647	0.227	0.42	0.495
24	0.847	0.873	0.949	0.076	0.102	0.866	0.571	0.385	0.186	0.481
25	0.706	0.706	1.031	0.325	0.325	0.764	0.572	0.203	0.369	0.561
26	0.93	1.094	1.068	−0.026	0.138	0.711	0.486	0.372	0.114	0.339
27	0.994	0.767	1.24	0.473	0.246	0.662	0.61	0.437	0.173	0.225
28	0.892	0.742	0.968	0.226	0.076	0.732	0.639	0.27	0.369	0.462
29	0.737	0.849	1.261	0.412	0.524	0.887	0.532	0.334	0.198	0.553
30	0.834	0.779	0.953	0.174	0.119	0.637	0.534	0.471	0.063	0.166
Average	0.785	0.879267	1.190267	0.311	0.405267	0.7498	0.6189	0.326633	0.292267	0.423167

Table 9
Comparison results between the PAES, NSGA-II and MOPSA algorithms according to the QM and SM.

Problem no.	Quality metric (QM)			$\bar{d}_1$	$\bar{d}_2$	Spacing metric (SM)			$\bar{d}_3$	$\bar{d}_4$
	PAES	NSGA-II	MOPSA			PAES	NSGA-II	MOPSA		
1	0	0.24	0.76	0.52	0.76	1.06	0.74	0.36	0.38	0.7
2	0	0.17	0.83	0.66	0.83	0.53	0.35	0.41	−0.06	0.12
3	0	0.07	0.93	0.86	0.93	0.67	0.4	0.5	−0.1	0.17
4	0.1	0.15	0.65	0.5	0.55	1.05	0.79	0.26	0.53	0.79
5	0	0	1	1	1	0.88	0.64	0.26	0.38	0.62
6	0	0	1	1	1	1.09	0.84	0.59	0.25	0.5
7	0	0.12	0.88	0.76	0.88	0.32	0.62	0.53	0.09	−0.21
8	0	0.21	0.79	0.58	0.79	0.86	0.85	0.23	0.62	0.63
9	0	0.14	0.86	0.72	0.86	0.69	0.82	0.24	0.58	0.45
10	0	0	1	1	1	1.07	0.51	0.45	0.06	0.62
11	0	0	1	1	1	0.8	0.52	0.23	0.29	0.57
12	0	0	1	1	1	0.78	0.47	0.68	−0.21	0.1
13	0	0.22	0.78	0.56	0.78	0.43	0.749	0.78	−0.031	−0.35
14	0	0.34	0.66	0.32	0.66	0.547	0.735	0.579	0.156	−0.032
15	0	0	1	1	1	0.456	0.583	0.512	0.071	−0.056
16	0	0.04	0.96	0.92	0.96	0.43	0.749	0.78	−0.031	−0.35
17	0	0.1	0.9	0.8	0.9	0.688	0.7	0.273	0.427	0.415
18	0	0	1	1	1	0.592	0.692	0.347	0.345	0.245
19	0	0	1	1	1	0.838	0.862	0.454	0.408	0.384
20	0	0	1	1	1	0.578	0.744	0.512	0.232	0.066
21	0	0	1	1	1	0.59	0.747	0.232	0.515	0.358
22	0	0	1	1	1	0.568	0.844	0.572	0.272	−0.004
23	0	0	1	1	1	0.591	0.823	0.511	0.312	0.08
24	0	0.04	0.96	0.92	0.96	0.674	0.731	0.395	0.336	0.279
25	0	0	1	1	1	0.624	0.573	0.374	0.199	0.25
26	0	0	1	1	1	0.87	0.596	0.379	0.217	0.491
27	0	0	1	1	1	0.672	0.855	0.322	0.533	0.35
28	0	0.1	0.9	0.8	0.9	0.574	0.511	0.243	0.268	0.331
29	0	0	1	1	1	0.862	0.696	0.312	0.384	0.55
30	0	0.05	0.95	0.9	0.95	0.892	0.567	0.452	0.115	0.44
Average	0.0033333	0.0663333	0.927	0.8606667	0.923667	0.7092	0.6769	0.4256333	0.2512667	0.2835667

Table 10
Comparison results between the PAES, NSGA-II and MOPSA algorithms according to the DM and MID.

Problem no.	Diversity metric (DM)			$\bar{d}_1$	$\bar{d}_2$	Mean ideal distance (MID)			$\bar{d}_3$	$\bar{d}_4$
	PAES	NSGA-II	MOPSA			PAES	NSGA-II	MOPSA		
1	0.782	1.117	1.393	0.276	0.611	0.879	0.609	0.674	−0.065	0.205
2	0.821	1.123	1.194	0.071	0.373	0.937	0.534	0.241	0.293	0.696
3	0.816	1.142	0.956	−0.186	0.14	0.81	0.47	0.252	0.218	0.558
4	0.849	0.829	1.293	0.464	0.444	0.762	0.483	0.271	0.212	0.491
5	0.768	1.061	1.255	0.194	0.487	0.726	0.762	0.283	0.479	0.443
6	0.856	1.104	1.014	−0.09	0.158	0.932	0.67	0.511	0.159	0.421
7	0.754	0.969	1.074	0.105	0.32	0.761	0.587	0.487	0.1	0.274
8	1.098	1.098	1.182	0.084	0.084	0.816	0.765	0.226	0.539	0.59
9	0.742	1.125	1.259	0.134	0.517	0.866	0.441	0.666	−0.225	0.2
10	0.522	1.112	1.143	0.031	0.621	0.768	0.698	0.565	0.133	0.203
11	0.802	1.076	0.852	−0.224	0.05	0.893	0.695	0.569	0.126	0.324
12	0.764	0.88	1.106	0.226	0.342	0.845	0.625	0.231	0.394	0.614
13	0.944	1.03	1.19	0.16	0.246	0.745	0.473	0.631	−0.158	0.114
14	0.883	0.856	1.012	0.156	0.129	0.935	0.639	0.668	−0.029	0.267
15	0.948	0.886	0.845	−0.041	−0.103	0.73	0.52	0.693	−0.173	0.037
16	1.02	0.827	0.933	0.106	−0.087	0.788	0.453	0.63	−0.177	0.158
17	1.038	0.973	0.876	−0.097	−0.162	0.771	0.485	0.593	−0.108	0.178
18	0.686	1.005	0.94	−0.065	0.254	0.859	0.758	0.457	0.301	0.402
19	0.662	0.964	1.02	0.056	0.358	0.855	0.663	0.379	0.284	0.476
20	0.914	0.928	1.064	0.136	0.15	0.799	0.545	0.32	0.225	0.479
21	0.642	0.993	1.029	0.036	0.387	0.829	0.677	0.431	0.246	0.398
22	0.836	0.994	1.238	0.244	0.402	0.848	0.62	0.352	0.268	0.496
23	0.839	1.004	1.059	0.055	0.22	0.721	0.66	0.307	0.353	0.414
24	0.988	0.991	1.272	0.281	0.284	0.967	0.834	0.587	0.247	0.38
25	0.818	1.084	1.119	0.035	0.301	0.719	0.661	0.429	0.232	0.29
26	0.777	0.862	1.279	0.417	0.502	0.831	0.656	0.589	0.067	0.242
27	0.902	1.013	0.92	−0.093	0.018	0.866	1.271	0.585	0.686	0.281
28	0.934	0.871	1.138	0.267	0.204	0.564	0.972	0.303	0.669	0.261
29	0.834	0.835	0.944	0.109	0.11	0.711	0.486	0.372	0.114	0.339
30	0.898	1.042	1.325	0.283	0.427	0.862	1.21	0.637	0.573	0.225
Average	0.8379	0.9931333	1.09746	0.104	0.2595	0.81316	0.6640667	0.46463	0.1994	0.34853

4.6. Comparison of meta-heuristic solution methods

The comparison results of the four suggested algorithms according to four proposed comparison metric indices are illustrated in Tables 7–11. In Tables 7 and 8, 30 test problems are solved and the performance of the MOICA is compared with the NSGA-II and PAES. According to the QM and the MID, the proposed MOICA is superior in all the 30 test problems. In addition, it is superior in most test problems according to the DM and the SM. Based on the comparison metrics, the proposed MOICA is superior to the NSGA-II and PAES.

In Tables 9 and 10, 30 test problems are solved and the performance of the proposed MOPSA is compared with the NSGA-II and PAES. According to the QM, the proposed MOPSA is precisely gets better results in all the 30 test problems. In addition, it performs considerably well in most test problems according to the DM, SM and MID. It can be seen that the supremacy of the proposed MOPSA is obviously demonstrated in comparison with the NSGA-II and PEAS.

In Table 11, 30 test problems are solved by the proposed MOPSA and MOICA and the performance of the proposed MOICA is compared with the proposed MOPSA. Based on the obtained results, the proposed MOICA is recommended for the proposed bi-objective location-routing-inventory (LRI) problem.

4.7. Statistical hypothesis tests on comparison results

A statistical hypothesis test for each two methods according to each comparison metric is designed until the better performance of the proposed methods is measured. In Tables 7 and 8, according to the QM and MID, the proposed MOICA is superior rather than the NSGA-II and PAES in all the 30 test problems. Furthermore, in Tables 7 and 8 according to the DM and SM, the proposed MOICA is superior rather than the PAES in all the 30 test problems, consequently the statistical hypothesis test is not needed for each two of them.

In Tables 9 and 10, according to the QM, the proposed MOPSA is precisely gets better results rather than the NSGA-II and the PAES in all the 30 test problems. Furthermore, in Tables 9 and 10 according to the MID, the proposed MOPSA is precisely gets better results rather than the PAES in all the 30 test problems; therefore, the statistical hypothesis test is not needed for each two of them.

H_0 represents no significant difference between every two methods (or $\bar{d} = d'_{benchmark} - d_{proposedmethod} = 0$ for the SM and MID that the lower value of them shows the better performance and $\bar{d} = d_{proposedmethod} - d'_{benchmark} = 0$ for the QM and DM that the higher value of them shows the better performance). H_1 indicates that the proposed algorithm has the significant better performance rather than the benchmarked algorithms (or $\bar{d} = d'_{benchmark} - d_{proposedmethod} > 0$ for the SM and MID that the lower value of them shows the better performance and $\bar{d} = d_{proposedmethod} - d'_{benchmark} > 0$ for the QM and DM that the lower value of them shows the better performance).

The above statistical hypothesis tests have been executed under the 0.975 confidence level. To solve these tests, the following steps are performed.

Step 1: $\begin{cases} H_0: & \bar{d} = 0 \\ H_1: & \bar{d} > 0 \end{cases}$ where the test statistic is $t = \frac{\bar{d}-\mu_0}{s/\sqrt{n}}$ and $\alpha = 0.025$ since it is assumed that the data follows normal distribution with a unknown standard deviation of the statistical population. The data follows a normal distribution because 30 test problems are produced. Because of the Central Limit Theorem, we can assume that the data follow a normal distribution if the number of the data is greater than 27.

Step 2: The null hypothesis is rejected if $t \geqslant t_{\alpha,n-1}$ or $t \geqslant 2.045$ where t is computed based on the above formulation in Step 1.

Table 11
Comparison results between the MOICA and MOPSA algorithms according to QM, SM, DM and MID.

Problem no.	Quality metric (QM)		$\bar{d}_1$	Spacing metric (SM)		$\bar{d}_2$	Diversity metric (DM)		$\bar{d}_3$	Mean ideal distance (MID)		$\bar{d}_4$
	MOICA	MOPSA		MOICA	MOPSA		MOICA	MOPSA		MOICA	MOPSA	
1	0.92	0.76	0.16	0.444	0.36	−0.084	1.379	1.393	−0.014	0.255	0.674	0.419
2	0.73	0.83	−0.1	0.315	0.41	0.095	1.143	1.194	−0.051	0.258	0.241	−0.017
3	0.8	0.93	−0.13	0.385	0.5	0.115	1.3	0.956	0.344	0.302	0.252	−0.05
4	1	0.65	0.35	0.373	0.26	−0.113	0.971	1.293	−0.322	0.48	0.271	−0.209
5	0.9	1	−0.1	0.359	0.26	−0.099	1.111	1.255	−0.144	0.317	0.283	−0.034
6	1	1	0	0.282	0.59	0.308	1.358	1.014	0.344	0.282	0.511	0.229
7	0.85	0.88	−0.03	0.274	0.53	0.256	1.296	1.074	0.222	0.245	0.487	0.242
8	1	0.79	0.21	0.335	0.23	−0.105	1.38	1.182	0.198	0.519	0.226	−0.293
9	0.75	0.86	−0.11	0.303	0.24	−0.063	1.228	1.259	−0.031	0.312	0.666	0.354
10	0.9	1	−0.1	0.243	0.45	0.207	0.917	1.143	−0.226	0.239	0.565	0.326
11	1	1	0	0.586	0.23	−0.356	1.325	0.852	0.473	0.43	0.569	0.139
12	1	1	0	0.303	0.68	0.377	1.367	1.106	0.261	0.227	0.231	0.004
13	1	0.78	0.22	0.394	0.78	0.386	1.24	1.19	0.05	0.385	0.631	0.246
14	0.75	0.66	0.09	0.476	0.579	0.103	1.279	1.012	0.267	0.203	0.668	0.465
15	0.74	1	−0.26	0.314	0.512	0.198	1.212	0.845	0.367	0.372	0.693	0.321
16	1	0.96	0.04	0.521	0.78	0.259	1.24	0.933	0.307	0.447	0.63	0.183
17	1	0.9	0.1	0.442	0.273	−0.169	1.098	0.876	0.222	0.282	0.593	0.311
18	0.88	1	−0.12	0.276	0.347	0.071	1.084	0.94	0.144	0.245	0.457	0.212
19	1	1	0	0.486	0.454	−0.032	1.394	1.02	0.374	0.319	0.379	0.06
20	1	1	0	0.61	0.512	−0.098	0.918	1.064	−0.146	0.312	0.32	0.008
21	1	1	0	0.636	0.232	−0.404	1.343	1.029	0.314	0.239	0.431	0.192
22	1	1	0	0.481	0.572	0.091	1.357	1.238	0.119	0.43	0.352	−0.078
23	0.96	1	−0.04	0.423	0.511	0.088	1.298	1.059	0.239	0.227	0.307	0.08
24	0.9	0.96	−0.06	0.359	0.395	0.036	0.949	1.272	−0.323	0.385	0.587	0.202
25	1	1	0	0.224	0.374	0.15	1.031	1.119	−0.088	0.203	0.429	0.226
26	1	1	0	0.512	0.379	−0.133	1.068	1.279	−0.211	0.372	0.589	0.217
27	0.95	1	−0.05	0.335	0.322	−0.013	1.24	0.92	0.32	0.437	0.585	0.148
28	1	0.9	0.1	0.572	0.243	−0.329	0.968	1.138	−0.17	0.27	0.303	0.033
29	1	1	0	0.303	0.312	0.009	1.261	0.944	0.317	0.334	0.372	0.038
30	1	0.95	0.05	0.394	0.452	0.058	0.953	1.325	−0.372	0.471	0.637	0.166
Average	0.934333	0.927	0.0073	0.3986	0.42563	0.026966	1.19026	1.09746	0.0928	0.32663	0.46463	0.138

Step 3: At first, the mean and standard deviation of the data reported in column $\overline{d_4}$ in Table 7 are computed. In this way, $\overline{d}$ = 0.228 (i.e., average of difference) and S = 0.169 (i.e., standard deviation). Also, by replacing all parameters and $\mu_0 = 0$, $n = 30$ into the t formulation, it is conducted that $t = \frac{0.228-0}{0.169/\sqrt{30}} = 7.377$.
Step 4: Because t = 7.377 and it is greater than $t_{0.25,\ 30\text{-}1}$ = 2.045, thus the null hypothesis is rejected and the alternative hypothesis is accepted. In other words, under the 97.5% confidence level, the hypothesis $\overline{d} = d'_{benchmark} - d_{proposedmethod} > 0$ is accepted. Finally, the results of the proposed algorithm are less than the benchmark algorithms. Thus, our proposed algorithm is more effective to solve the mathematical formulation.

The other entire statistical hypothesis tests for each two algorithms are done same as the above-mentioned steps and given below.

In Table 8, column $\overline{d_1}$ comparison between the MOICA and NSGA-II according to the DM:

Because $t = 10.75$ is greater than $t_{0.25,30-1} = 2.045$, thus the proposed algorithm has significantly better performance rather than the benchmark algorithm.

In Table 9, column $\overline{d_3}$comparison between the NSGA-II and MOPSA according to the SM:

Because $t = 6.45$ is greater than $t_{0.25,30-1} = 2.045$, thus the proposed algorithm has significantly better performance rather than benchmark algorithm.

In Table 9, column $\overline{d_4}$comparison between the PAES and MOPSA according to the SM:

Because $t = 5.16$ is greater than $t_{0.25,30-1} = 2.045$, thus the proposed algorithm has significantly better performance rather than the benchmark algorithm.

In Table 10, column $\overline{d_1}$ comparison between the NSGA-II and MOPSA according to the DM:

Because $t = 3.49$ is greater than $t_{0.25,30-1} = 2.045$, thus the proposed algorithm has significantly better performance rather than the benchmark algorithm.

In Table 10, column $\overline{d_2}$ comparisons between the PAES and MOPSA according to the DM:

Because $t = 6.96$ is greater than $t_{0.25,30-1} = 2.045$, thus the proposed algorithm has significantly better performance rather than the benchmark method.

In Table 10, column $\overline{d_3}$ comparisons between the NSGA-II and MOPSA according to the MID: Because $t = 4.45$ is greater than $t_{0.25,30-1} = 2.045$, thus the proposed algorithm has significantly better performance rather than the benchmark algorithm.

In Table 11, column $\overline{d_1}$ comparison between MOICA and MOPSA according to the QM:

Because $t = 0.34$ is less than $t_{0.25,30-1} = 2.045$, thus there is no significant difference between every two algorithms.

In Table 11, column $\overline{d_2}$ comparison between the MOICA and MOPSA according to the SM:

Because $t = 0.74$ is less than $t_{0.25,30-1} = 2.045$, thus there is no significant difference between every two methods.

In Table 11, column $\overline{d_3}$ comparison between the MOICA and MOPSA according to the DM:

Because $t = 2.059$ is greater than $t_{0.25,30-1} = 2.045$, thus the proposed algorithm has significantly better performance rather than the benchmark algorithm.

In Table 11, column $\overline{d_4}$ comparison between the MOICA and MOPSA according to the MID:

Because $t = 4.29$ is greater than $t_{0.25,30-1} = 2.045$, thus the proposed algorithm has significantly better performance rather than the benchmark algorithm.

Based on all the statistical hypothesis tests, we find that our proposed MOICA has better performance in comparison with other meta-heuristics except rather than MOPSA according to the QM and SM, in which there is no significant difference between them. Therefore the proposed MOICA is recommended for the given problem.

5. Conclusion

This paper has presented a multi-objective imperialist competitive algorithm (MOICA) for solving a bi-objective location-routing-inventory (LRI) problem, in which decisions have been made in a multi-period and multi-product distribution system. The model has considered three key LRI decisions in one problem. The main contribution of this paper has been to consider the integrated problem in a multi-product and multi-period supply chain system. Additionally, the model has considered the travelling time among customers probabilistic. It has determined the location of a set of distribution centers to serve a set of customers, inventory optimal policy, and routes of vehicles to satisfy the customers' demand in each period of planning horizon for each kind of products such that the total cost and the maximum mean time for delivering to customers have been minimized. To validate the proposed MOICA, a number of test problems have been generated to evaluate the performance of the proposed algorithm in comparison with three multi-objective evolutionary algorithms (i.e., non-dominated sorting genetic algorithm II (NSGA-II), Pareto archived evolution strategy (PAES) and multi-objective parallel simulated annealing (MOPSA)). To do this, four comparison metrics (i.e., quality, spacing, diversity and mean ideal distance) have been applied. The associated results have indicated that the proposed MOICA has outperformed the foregoing algorithms. Developing other meta-heuristics for the given problem can be an interesting further research direction. Beside, analyzing the model under uncertainty on each parameter and time window can be a valuable subject. Finally, it is interesting to also consider the LRI problem in a closed loop supply chain system and a reverse logistics network.

Acknowledgements

The authors would like to thank the Editor-in-Chief of C&IE and anonymous reviewers for their helpful comments and suggestions, which greatly improved the presentation of this paper. Additionally, the second author would like to acknowledge the partially financial support of the University of Tehran for this research under Grant No. 8106043/1/24.

References

Abdelmaguid, T. F., Dessouky, M. M., & Ordóñez, F. (2009). Heuristic approaches for the inventory-routing problem with backlogging. *Computers and Industrial Engineering, 56*, 1519–1534.

Ahmadi Javid, A., & Azad, N. (2010). Incorporating location, routing and inventory decisions in supply chain network design. *Transportation Research – Part E, 46*, 582–597.

Ahmadi Javid, A., & Seddighi, A. (2012). A location-routing-inventory model for designing multi-source distribution networks. *Engineering Optimization, 44*(6), 637–656.

Al Dhaheri, N. A., & Diabat, A. (2010). A multi-product capacitated inventory-location model with risk pooling. In *Proceedings of the IEEE international conference on industrial engineering and engineering management* (pp. 1159–1164). 7–10 December 2010, Macao, China.

Ambrosino, D., & Scutella, M. G. (2005). Distribution network design: New problems and related models. *European Journal of Operational Research, 165*, 610–624.

Atashpaz-Gargari, E., & Lucas, C. (2007). Imperialist competitive algorithm: An algorithm for optimization inspired by imperialist competitive. In *IEEE congress on evolutionary computation*, 25–28 September 2007, Singapore.

Bard, J. F., & Nananukul, N. (2009). Heuristics for a multi-period inventory routing problem with production decisions. *Computers and Industrial Engineering, 57*, 713–723.

Chanchan, W., Zujun, M., & Huajun, L. (2008). Stochastic dynamic location-routing-inventory problem in closed-loop logistics system for reusing end-of-use products. In *Proceedings of the IEEE international conference on intelligent computation technology and automation* (pp. 691–695).

Corne, D. W., Knowles, J.D., & Oates, M. J. (2000). The Pareto envelope-based selection algorithm for multi-objective optimization. In M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J. J. Merelo, H.-P. Schwefel, (Eds.), *Parallel Problem Solving from Nature – PPSN VI* (pp. 839–848). Paris, France, 18–20 September 2000.

Daskin, M., Coullard, C., & Shen, Z. J. (2002). An inventory–location model: Formulation, solution algorithm and computational results. *Annals of Operations Research, 110*, 83–106.

Deb, K., Agrawal, S., Pratap, A., & Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J.J. Merelo, H.-P. Schwefel, (Eds.), *Parallel Problem Solving from Nature – PPSN VI* (pp. 849–858). Paris, France, 18–20 September 2000.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transaction on Evolutionary Computation, 6*(2), 182–197.

Gaur, V. L., & Fisher, M. (2004). A periodic inventory routing problem at a supermarket chain. *Operations Research, 52*(6), 813–822.

Hanczar, P. (2012). A fuel distribution problem – Application of new multi-item inventory routing formulation. *Procedia – Social and Behavioral Sciences, 54*, 726–735.

Hiassat, A. H., & Diabat, A. (2011). A location routing inventory problem with perishable product. In *The proceedings of the 41st international conference on computers and industrial engineering*. Los Angeles, CA, 23–25 October 2011.

Kang, J.-H., & Kim, Y.-D. (2010). Inventory replenishment and delivery planning in a two-level supply chain with compound Poisson demands. *International Journal of Advanced Manufacturing Technology, 49*, 1107–1118.

Knowles, J. D., & Corne, D. W. (1999). The Pareto archived evolution strategy: A new base line algorithm for Pareto multi-objective optimization. *Congress on Evolutionary Computation (CEC99)* (Vol. 1, pp. 98–105). NJ, Piscataway: IEEE Press.

Liao, S., Hsieh, C., & Lai, P. (2011). An evolutionary approach for multi-objective optimization of the integrated location–inventory distribution network problem in vendor-managed inventory. *Expert Systems with Applications, 38*, 6768–6776.

Liu, S. C., & Lee, S. B. (2003). A two-phase heuristic method for the multi-depot location routing problem taking inventory control decisions in to consideration. *International Journal of Advanced Manufacturing Technology, 22*, 941–950.

Liu, S. C., & Lin, C. C. (2005). A heuristic method for the combined location routing and inventory problem. *International Journal of Manufacturing Technology, 26*, 327–381.

Moin, N. H., Salhi, S., & Aziz, N. A. B. (2011). An efficient hybrid genetic algorithm for the multi-product multi-period inventory routing problem. *International Journal of Production Economics, 133*, 334–343.

Moradi, H., Zandieh, M., & Mahdavi, I. (2011). Non-dominated ranked genetic algorithm for a multi-objective mixed model assembly line sequencing problem. *International Journal of Production Research, 49*(12), 3479–3499.

Shen, Z., & Qi, L. (2007). Incorporating inventory and routing cost in strategic location models. *European Journal of Operational Research, 179*, 372–389.

Tavakkoli-Moghaddam, R., Azarkish, M., & Sadeghnejad, A. (2011). A new hybrid multi-objective Pareto archive PSO algorithm for a bi-objective job shop scheduling problem. *Expert Systems with Applications, 38*(9), 10812–10821.

Zandieh, M., Dorri, B., & Khamseh, A. R. (2009). Robust meta-heuristics for group scheduling with sequence-dependent setup times in hybrid flexible flow shops. *International Journal of Advanced Manufacturing Technology, 43*, 767–777.