Expert Systems With Applications 202 (2022) 117364

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Modified metaheuristic algorithms to design a closed-loop supply chain network considering quantity discount and fixed-charge transportation

Golara Chaharmahali [a], Davoud Ghandalipour [a], Milad Jasemi [b,*], Saber Molla-Alizadeh-Zavardehi [a]

[a] *Department of Industrial Engineering, Azad University, Masjed Soleyman Branch, Masjed Soleyman, Iran*
[b] *Department of Computer Informatics, Stephens College of Business, University of Montevallo, AL, United States*

ARTICLE INFO

Keywords:
Closed-loop supply chain
Modified metaheuristic algorithm
Quantity discount
Fixed-charge transportation
Taguchi method

ABSTRACT

Discount is an efficient way to reduce the closed-loop supply chain costs, and applying it would make the model closer to real ones. In this paper, the quantity discount is firstly applied along with fixed and variable transportation costs. The application of well-known, efficient algorithms, alongside developing modified versions to address the developed model is another contribution of this study. To calibrate the proposed algorithms' parameters and operators, the Taguchi method is used. In this regard, several test problems in different sizes are generated considering the concerns of real-world cases, and the algorithms' efficiencies are investigated by the relative percentage deviation method. The results show the superior performance of the hybrid algorithm of modified differential evolution and restart mechanism (MDE_Restart) and the algorithm of modified differential evolution (MDE) compared to the other employed algorithms.

1. Introduction and literature review

Closed-loop supply chain (CLSC) is one of the important topics in supply chain (SC) areas that includes both forward and reverse supply chains and has attracted both academia and industrial practitioners, especially during the last decade (Salehi-Amiri, Zahedi, Gholian-Jouybari, Calvo, & Hajiaghaei-Keshteli, 2022). Generally, the goods are transferred from the suppliers to customers, namely forward logistics, and the used, damaged, or unsold goods are transferred backward from the customers to the suppliers, namely reverse supply chain or logistics (Kannan, Sasikumar, & Devika, 2010; Ali, Paksoy, Torğul, & Kaur, 2020). Both forward and reverse configurations effect on performances of each other, considerably. Therefore, their network design should be considered in an integrated fashion to avoid any side optimizations due to the separate design (Hosseini, Paydar, & Hajiaghaei-Keshteli, 2021).

As one of the main important and primary works in this area, Fleischmann, Beullens, Bloemhof-Ruwaard, and Van Wassenhove (2001) developed an integrated system of forward and reverse logistics. The reverse stream of their model had separation and manufacturing centers for inspection and reproduction of the returned goods and landfilling centers for useless goods. Likewise, Salema, Barbosa-Povoa, and Novais (2007) investigated on similar network considering reverse logistics, capacity constraints, multiple goods, and uncertainty in demand and returned goods. Wang and Hsu (2010) developed a CLSC with distribution and recycling centers. In their model, the recyclable goods are sent to factories as raw materials, and the unrecyclable goods are landfilled. El-Sayed, Afia, and El-Kharbotly (2010) made a multi-period multi-echelon forward-reverse logistic network with risk. They used stochastic programming to address the problem of uncertainty. Also, Soleimani, Seyyed-Esfahani, and Akbarpour Shirazi (2013) developed a multi-period multi-echelon multi-product model based on mixed-integer linear programming. In their model, the customers could get the goods through the manufacturers, distributors, or warehouses. They applied CPLEX as a strong solution for small-scale problems and a developed genetic algorithm (GA) for large-scale ones. Ramezani, Bashiri, and Tavakkoli-Moghaddam (2013) considered the same network and developed a model to maximize the CLSC with return rate and uncertain demand. They applied a scenario relaxation algorithm for their proposed model. Zeballos, Mendez, Barbosa-Povoa, and Novais (2014) worked on uncertain supply and demand both to minimize the total cost and maximize recycled goods income. In addition, Özceylan,

* Corresponding author.
E-mail addresses: golarachaharmahalii@gmail.com (G. Chaharmahali), davoudghandalipour@gmail.com (D. Ghandalipour), mjasemiz@montevallo.edu (M. Jasemi), saber.alizadeh@gmail.com (S. Molla-Alizadeh-Zavardehi).

https://doi.org/10.1016/j.eswa.2022.117364
Received 11 August 2021; Received in revised form 6 April 2022; Accepted 25 April 2022
Available online 28 April 2022
0957-4174/

Paksoy, and Bektas (2014) developed and integrated CLSC to optimize strategic and tactical decisions. The former refers to the goods transferred in both forward and backward SC, and the latter is about disassembled lines in the reverse chains. The model was formulated to minimize the costs of transportation, purchasing, refurbishing, and disassembly. Also, Devika, Jafarian, and Nourbakhsh (2014) considered a multiple-criteria decision-making (MCDM) model with suppliers, manufacturers, distributors, retailers, customers, collection/inspection, and recycling centers. In their model, along with the regular cost criterion, the social aspect of sustainability is considered. They solved it by six hybrid metaheuristic methods. Similarly, Govindan, Darbari, Agarwal, and Jha (2017), using the MCDM model along with minimizing the cost and maximizing sustainability, tried to maximize the suppliers' performance. Soleimani and Kannan (2015) worked on solving a multi-period multi-echelon multi-product CLSC with mathematical programming tools. For large-scale models, they also developed an efficient hybrid particle swarm genetic algorithm. Also, Soleimani, Govindan, Saghafi, and Jafari (2017) worked on a green fuzzy MCDM CLSC and solved it with GA. In this study, the wasted working days due to occupational accidents are minimized. Another similar paper is, Safaei, Roozbeh, and Paydar (2017) used a robust optimization approach to address the uncertainty in their case-based CLSC model. Mohamadpour Tosarkani and Hassanzadeh Amin (2018) applied a fully fuzzy MCDM model on a CLSC case and developed it. Fathollahi Fard and Hajaghaei-Keshteli (2018) considered a tri-level model to formulate the same network, utilizing several metaheuristics. Gholizadeh and Fazlollahtabar (2020) applied robust optimization and metaheuristics for a green CLSC under uncertainty with different grades and emphasis on profitability alongside a case study in the melting industry. Nayeri, Paydar, Asadi-Gangraj, and Emami (2020) formulated a sustainable SCLSC network considering a water tank and addressed the model by goal programming and robust fuzzy optimization. Also, Lotfi, Zare Mehrjerdi, Pishvaee, Sadeghieh, and Weber (2021) proposed a CLSC to minimize the costs, CO2 emissions, and energy, along with maximizing employment, taking into account sustainability, resilience, robustness, and risk aversion. Salehi-Amiri, Zahedi, Akbapour, and Hajiaghaei-Keshteli (2021) proposed a CLSC network in the walnut industry to decide on the number of facilities and optimize the intra-levels forward ad reverse stream cost. To solve the model, both exact and *meta*-heuristic methods are applied and finally, the best solutions are achieved by the Taguchi method. More recently, Lotfi, Sheikhi, Amra, AliBakhshi, and Weber (2021) designed to maintain robustness against changes in demand, minimize expenses, environmental pollution, and energy consumption and maximize the employment rate a resilient, and sustainable CLSC model. They considered the risk criterion in their model.

In the real world, in addition to the transportation cost, there is usually a fixed penalty at all stages of the supply chain, independent of the transported goods quantity. The fixed penalty can be due to different reasons like setup expenses, permit charges, toll fees, and others (Hajiaghaei-Keshteli, Molla-Alizadeh-Zavardehi, & Tavakkoli-Moghaddam, 2010; Bertazzi & Maggioni, 2018; Midya, Roy, & Weber, 2021). In the literature of CLSC, there are rarely studies with fixed-cost transportation. Soleimani (2018) developed a robust CLSC with two opposite objective functions in an Iranian case study. In the model, the transportation costs are considered with the fixed cost part. Then, to reach the Pareto front and non-dominant solutions, the repetitive epsilon-constraint approach was developed. Likewise, Gholamian, Mahdavi, Mahdavi-Amiri, and Tavakkoli-Moghaddam (2021) developed a model for a large-scale sustainable CLSC. To solve the model, they proposed a new interactive fuzzy approach.

In most of the supply chain models, it is assumed that the cost of the raw materials per unit is fixed, which is not always true in the real world. The sellers, either suppliers or manufacturers, usually suggest discounts to motivate the customers to order bigger. Discounts on raw materials purchases could reduce the total supply chain cost considerably. Due to bigger orders, the sellers would reduce the price per unit, and therefore the buyers according to the purchase discount could choose the best supplier (Tsai, 2007). There are few studies in the field of supplier selection based on quantity discounts for the design of CLSC. In this regard, Shafiei Kisomi, Solimanpur, and Doniavi (2016) developed a CLSC under uncertainty with the supplier selection model considering price discount and solved it with robust optimization. In a study, Sadeghi Rad and Nahavandi (2018) developed an integrated multi-period multi-echelon multi-product MCDM of a green CLSC offering discounts. They

Table 1
Review of CLSC network design literature.

Year	Reference	Fixed Cost for Opening	Facility Capacity	Facilities Location	Transportation Cost	Fixed-Charge Transportation	Discount	Heuristic / Metaheuristic
2001	Fleischmann et al.	*		*	*			
2007	Salema et al.	*	*	*	*			
2010	Wang and Hsu	*	*	*	*			*
2010	El-Sayed et al.	*	*	*	*			
2013	Soleimani et al.	*	*	*	*			*
2013	Ramezani et al.	*	*	*	*			
2014	Zeballos et al.	*	*		*			
2014	Devika et al.	*	*	*	*			*
2014	Özceylan et al.		*		*			
2015	Soleimani and Kannan	*	*	*	*			*
2016	Shafiei Kisomi et al.	*	*	*	*		*	
2017	Govindan et al.		*	*	*			
2017	Soleimani et al.	*	*	*	*			*
2017	Safaei et al.		*	*	*			
2018	Sadeghi Rad and Nahavandi	*	*	*	*		*	
2018	Mohamadpour Tosarkani and Hassanzadeh Amin	*	*	*	*			
2018	Fathollahi-Fard and Hajaghaei-Keshteli	*	*	*	*			*
2018	Soleimani	*	*	*	*	*		
2019	Ghahremani-Nahr et al.	*	*	*	*		*	*
2020	Gholizadeh and Fazlollahtabar		*	*	*			*
2020	Nayeri et al.	*	*	*	*			
2021	Khalili Nasr et al.	*	*	*	*		*	
2021	Salehi-Amiri et al.	*	*	*	*			*
2021	Gholamian et al.	*	*	*	*	*		
	This study	*	*	*	*	*	*	*

did some numerical examples with CPLEX. Also, Ghahremani-Nahr, Kian, and Sabet (2019) developed a multi-echelon, multi-product, multi-period CLSC network with discounted raw materials to analyze some uncertain parameters like customer demand, transportation costs, a fraction of returned products, raw materials cost, and shortage costs by robust fuzzy programming. They used a novel whale optimization algorithm to minimize the total network cost by using a modified priority-based encoding method. In the same vein, Khalili Nasr, Tavana, Alavi, and Mina (2021) developed a multi-objective CLSC and used the Fuzzy method to solve the model. Then by the fuzzy best-worst method, they selected the best supplier based on economic, environmental, social, and circular measures.

The development of appropriate and strong solution methods in CLSC problems is very important to make better decisions. The literature contains some exact and metaheuristic methods to solve the mathematical models. According to the above literature and Table 1, there is no CLSC study with a quantity discount and transportation fixed-charge. This study develops such a model for the first time and solves it with basic and modified metaheuristics algorithms. Based on the literature review and discussions of this part, there have been no studies so far that consider the transportation fixed cost and quantity discount concepts simultaneously in an integrated CLSC model. It is exactly the gap this study aims to fill.

According to Fig. 1, in the forward flow after purchasing the discounted raw materials from the suppliers and production by the manufacturers, the goods are distributed among the customers by the hybrid centers of distribution and collection (HCDC). In fact, an HCDC center is considered first as a distributor and then as a collector on a round trip. In a CLSC, an HCDC center can act as both distributor and collector (Wang & Hsu, 2010). In the reverse flow, the used goods are given to HCDC before being sent to the recycling centers. The restorable returned goods are sent to the manufacturers in the recycling centers, and others are landfilled. In other words, the core concept of CLSC is recycling and reusing the delivered products to customers. Therefore, it has huge importance due to protecting the environment and decreasing costs.

The CLSC model developed, discussed, and solved in this study includes the transportation fixed cost and quantity discounts to motivate the buyers for bigger orders. In this study, the developed model is solved with eight basic and modified metaheuristic algorithms. Table 1 shows the contribution this study could have to the literature on CLSCs.

2. The model formulation

Here a six-stage CLSC model with a quantity discount and transportation fixed-charge is developed. According to Fig. 1, at stage 1, the suppliers deliver the product to the manufacturers with a general discount and after manufacturing centers, the products are delivered to the customers through the distribution centers. After meeting the demand, the materials and products that are usable or recyclable are sent to the collection centers to be used as materials. The stages include suppliers (I), manufacturers (J), the HCDC (K), customers (L), recycling centers (M), and discount levels (N). The most important suppositions are as follows:

- The best suppliers are selected according to the offered discounts.
- The location, number, and capacity of each facility are specified.
- The stream of goods is allowed only between adjacent levels.
- The number and capacity of the facilities to open are limited.
- The forward and reverse capacities are specified.
- Each customer's demand must be satisfied.
- Dump and recycle rates are fixed.
- Unrecyclable materials are got out of the chain.
- Fixed transportation cost is considered for each stage of CLSC.

2.1. The mathematical model

The new model's variables are as follows:

x_{ij}	Amount of goods transported from supplier i to manufacturer j
y_{jk}	Amount of goods transported from manufacturer j to HCDC k
z_{kl}	Amount of goods transported from HCDC k to customer l
rz_{lk}	Amount of goods transported from customer l to HCDC k
o_{km}	Amount of goods transported from HCDC k to recycling center m
rd_{mj}	Amount of goods transported from recycling center m to manufacturer j
fyx_{ij}	1 if a good is transported from supplier i to manufacturer j; 0 otherwise
fcy_{jk}	1 if a good is transported from manufacturer j to HCDC k; 0 otherwise
fyz_{kl}	1 if a good is transported from HCDC k to customer l; 0 otherwise
$fyrz_{lk}$	1 if a good is transported from customer l to HCDC k; 0 otherwise
fyo_{km}	1 if a good is transported from HCDC k to recycling center m; 0 otherwise
$fyrd_{mj}$	1 if a good is transported from recycling center m to manufacturer j; 0 otherwise
α_j	If manufacturer j does any production it equals 1; otherwise 0
β_k	If HCDC k is founded it equals 1; otherwise 0
δ_m	If recycling center m is founded it equals 1; otherwise 0
q_i	Number of goods that are bought from supplier i at a discounted price
s_{in}	If the goods are bought from supplier i at the discount level of n, it equals 1; otherwise 0

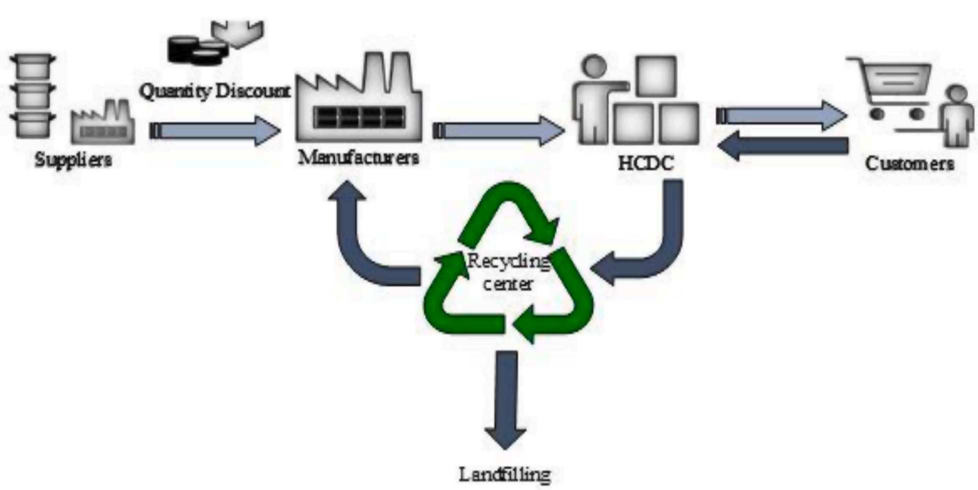

Fig. 1. The CLSC with quantity discount and transportation fixed cost.

The new model parameters are as follows:

b_j	Capacity of manufacturer j
sc_k	Capacity of HCDC k in both forward and reverse procurement
pd_k	Percentage of the total capacity for goods return in HCDC k
pc_l	Percentage of recycled goods from customer l
pl_m	Landfilling rate of center m
d_l	Demand of the customer l
e_m	Capacity of recycling center m
sx_{ij}	Transportation cost per unit from supplier i to manufacturer j
t_{jk}	Transportation cost per unit from manufacturer j to HCDC k
u_{kl}	Transportation cost per unit from HCDC k to customer l
ru_{lk}	Transportation cost per unit from customer l to HCDC k
v_{km}	Transportation cost per unit from HCDC k to recycling center m
w_{mj}	Transportation cost per unit from recycling center m to manufacturer j
f_j	Fixed cost of opening manufacturer j
g_k	Fixed cost of opening HCDC k
h_m	Fixed cost of opening recycling center m
â ± ·	Fixed cost for landfilling per unit
fx_{ij}	Fixed cost of transportation per unit from supplier i to manufacturer j
fy_{jk}	Fixed cost of transportation per unit from manufacturer j to HCDC k
fz_{kl}	Fixed cost of transportation per unit from HCDC k to customer l
frz_{lk}	Fixed cost of transportation per unit from customer l to HCDC k
fo_{km}	Fixed cost of transportation per unit from HCDC k to recycling center m
frd_{mj}	Fixed cost of transportation per unit from recycling center m to manufacturer j
pq_i	Price of ordered goods from supplier i that could have discount
$q2_{in}$	The lower bound of discount range n that is specified by supplier i
p_{in}	Price of a product unit by supplier i discount plan when the order quantity is $q2_{in}$
r_{in}	Discount curve slope in the discount level n by supplier i when the order quantity is between $q2_{in}$ and $q2_{i(n+1)}$
LN	Large number

The new model objective function to minimize the total cost is as follows:

$$\begin{aligned}\text{minimize TC} = & \sum_i\sum_j(sx_{ij}x_{ij}+fx_{ij}fyx_{ij})+\sum_j\sum_k(t_{jk}y_{jk}+fy_{jk}fcy_{jk}) \\ & +\sum_k\sum_l(u_{kl}z_{kl}+fz_{kl}fyz_{kl})+\sum_l\sum_k(ru_{lk}rz_{lk}+frz_{lk}fyrz_{lk}) \\ & +\sum_k\sum_m(v_{km}o_{km}+fo_{km}fyo_{km})\sum_m\sum_j(w_{mj}rd_{mj}+frd_{mj}fyrd_{mj}) \\ & +\sum_j f_j\alpha_j+\sum_k g_k\beta_k+\sum_m h_m\delta_m+\varphi\sum_m(pl_m\sum_k o_{km})+\sum_i pq_iq_i\end{aligned} \tag{1}$$

subject to.

$\sum_j x_{ij} = q_i$	$\forall i$	(2)
$\sum_k y_{jk} \le b_j\alpha_j$	$\forall j$	(3)
$\sum_i x_{ij} + \sum_m rd_{mj} = \sum_k y_{jk}$	$\forall j$	(4)
$\sum_l z_{kl} + \sum_m o_{km} \le sc_k\beta_k$	$\forall k$	(5)
$\sum_j y_{jk} = \sum_l z_{kl}$	$\forall k$	(6)
$\sum_m o_{km} \le pd_k sc_k\beta_k$	$\forall k$	(7)
$\sum_l rz_{lk} = \sum_m o_{km}$	$\forall k$	(8)
$\sum_k rz_{lk} \ge pc_l\sum_k z_{kl}$	$\forall l$	(9)
$\sum_k z_{kl} = d_l$	$\forall l$	(10)
$\sum_j rd_{mj} + pl_m\sum_k o_{km} \le e_m\delta_m$	$\forall m$	(11)
$\sum_k o_{km} = \sum_j rd_{mj} + pl_m\sum_k o_{km}$	$\forall m$	(12)
$pq_i = \sum_n s_{in}(p_{in} + r_{in}(q_i - q2_{in}))$	$\forall i$	(13)
$\sum_n s_{in}q2_{in} \le q_i \le \sum_n s_{in}q2_{i(n+1)}$	$\forall i$	(14)
$\sum_n s_{in} = 1$	$\forall i$	(15)
$x_{ij} \le LNfyx_{ij}$	$\forall i,j$	(16)
$y_{jk} \le LNfcy_{jk}$	$\forall j,k$	(17)
$z_{kl} \le LNfyz_{kl}$	$\forall k,l$	(18)
$rz_{lk} \le LNfyrz_{lk}$	$\forall l,k$	(19)
$o_{km} \le LNfyo_{km}$	$\forall k,m$	(20)
$rd_{mj} \le LNfyrd_{mj}$	$\forall m,j$	(21)

Constraint (2) shows that all the goods transferred from the supplier to the manufacturer are discounted. Constraints (3), (5), (7), and (11), guarantee no facilities' capacities are violated. Constraints (4), (6), (8), and (12), guarantee facilities' input and output flows are equal. Constraint (9) shows the relation of customers' returned goods with the recycling rate. Constraint (10) makes sure the customers' demands are met. Constraint (13) shows the price of bought goods with a discount. Constraint (14) guarantees the purchased quantity from a supplier at a special discounted price is within the eligible range. Constraint (15) guarantees that a purchase from a supplier is only at one discount level and not more. Constraints (16)-(23), all are logical and obvious regarding the associated variables.

2.2. Numerical example

To validate the model, here is a small problem in which $I = 3$, $J = 5$, $K = 2$, $L = 2$, $M = 2$, and $N = 2$ is solved with DICOPT solver in GAMS (24.8.5). The output approves the proposed model validity. Taheri-Bavil-Oliaei, Zegordi, and Tavakkoli-Moghaddam (2021) and Babaee Tirkolaee, Goli, Bakhsi, and Mahdavi (2017) also used GAMS to validate their proposed model. The parameters are presented in Table 2.

The GAMS outputs, that all validate (Table 3) the proposed mathematical model, are as follows:

3. Solution methodology

Since the type of model developed in this paper is NP-hard (Wang & Hsu, 2010), to solve the new model, eight basic and modified metaheuristic algorithms are used. We propose a particular decoding plan used for all the employed algorithms. In designing and utilizing metaheuristic algorithms, considering feasibility, intensification, and diversification phases are important (Chouhan, Khan, & Hajiaghaei-Keshteli, 2021). To improve the metaheuristic algorithms and be sure about the feasibility of the solution during the running of the algorithms, several ideas are generated and put into practice as modified metaheuristics.

3.1. Encoding scheme

The base of each metaheuristic approach is the way the answer is displayed. It depends on the problem nature, i.e., the number and dimension of both decision variables and constraints. Here, the Random-Key (RK) method displays the feasible solution (Rajabi, Najafi, Hajiaghaei-Keshteli, & Molla-Alizadeh-Zavardehi, 2013). The applied display method in the proposed algorithms leads to the generation of feasible answer vectors. The length of an answer vector for our problem is $(I + J) + (J + K) + (K + L) + (L + K) + (K + M) + (M + J) + I$. Based on this method, a random vector, including numbers between 0 and 1 for each supplier, manufacturer, HCDC, customer, and recycling center, is generated. Then each vector is ordered ascending, while the smaller numbers are of higher priority. Then, in each stage, the origin and destination with the highest priorities are selected, and an amount equal to the minimum of the facilities' capacities is carried from the origin to the destination. For better illustration of this sequence by RK, the small example mentioned before is being used again. Fig. 2 shows the decoding method of the first stage.

The decoding procedure based on the example in Fig. 2, is as follows:

Specifying the maximum amounts of purchase from each supplier:

$$q_i = rand\cdot(q2_{iN} - q2_{i1}) + q2_{i1} \tag{24}$$

Table 2
The parameters needed to solve the numerical example.

sx	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$
$i=1$	8	3	10	4	7
$i=2$	10	8	1	6	5
$i=3$	5	5	4	6	3

fx	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$
$i=1$	8	5	10	4	3
$i=2$	10	7	9	5	7
$i=3$	3	10	4	9	3

t	$k=1$	$k=2$
$j=1$	8	8
$j=2$	10	2
$j=3$	4	10
$j=4$	5	6
$j=5$	1	6

fy	$k=1$	$k=2$
$j=1$	8	8
$j=2$	10	2
$j=3$	8	10
$j=4$	5	2
$j=5$	6	6

u	$l=1$	$l=2$
$k=1$	7	2
$k=2$	5	1

fz	$l=1$	$l=2$
$k=1$	5	1
$k=2$	5	8

ru	$k=1$	$k=2$
$l=1$	9	9
$l=2$	4	3

frz	$k=1$	$k=2$
$l=1$	5	7
$l=2$	10	9

v	$m=1$	$m=2$
$k=1$	5	3
$k=2$	7	6

fo	$m=1$	$m=2$
$k=1$	5	5
$k=2$	1	9

w	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$
$m=1$	2	5	10	8	5
$m=2$	10	2	9	5	7

frd	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$
$m=1$	5	7	3	4	5
$m=2$	10	8	1	5	2

$q2$	$n=1$	$n=2$
$i=1$	100	200
$i=2$	100	200
$i=3$	100	200

p	$n=1$	$n=2$
$i=1$	20	19
$i=2$	22	21
$i=3$	18	17

r	$n=1$	$n=2$
$i=1$	-0.01	-0.01
$i=2$	-0.01	-0.01
$i=3$	-0.01	-0.01

	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$
f	90	90	90	90	90

	$k=1$	$k=2$
g	90	90

	$m=1$	$m=2$
h	90	90

Table 3
The numerical example output.

Objective function value			25019
x_{ij}	$x_{12}=140$	$x_{23}=200$	$x_{33}=200$
y_{jk}	$y_{22}=500$	$y_{31}=400$	
z_{kl}	$z_{11}=300$	$z_{12}=100$	$z_{21}=500$
rz_{lk}	$rz_{11}=80$	$rz_{21}=20$	$rz_{22}=300$
o_{km}	$o_{12}=100$	$o_{22}=300$	
rd_{mj}	$rd_{22}=360$		
q_i	$q_1=140$	$q_2=200$	$q_3=200$
pq_i	$pq_1=19.6$	$pq_2=21$	$pq_3=17$
s_{in}	$s_{11}=1$	$s_{21}=1$	$s_{31}=1$
α_j	$\alpha_2=1$	$\alpha_3=1$	
β_k	$\beta_1=1$	$\beta_2=1$	
δ_m	$\delta_2=1$		

Step 1: Name the first left element of the vector S_1, as the origin, i^*; and name the first left element of the vector S_2, as the destination, j^*.

$$x_{i^*j^*} = \min\left(q_i, b_j, total\,demand\right) \tag{25}$$

$$q_{i^*} = q_{i^*} - x_{i^*j^*} \tag{26}$$

Fig. 2. The coding plan for the first stage.

$$b_{j^*} = b_{j^*} - x_{i^*j^*} \tag{27}$$

$$total\ demand = total\ demand - x_{i^*j^*} \tag{28}$$

Step 2: If $q_{i^*} = 0$, omit the first argument of S_1; If $b_{j^*} = 0$, omit the first argument of S_2.
Step 3: Repeat the above steps until the total demand is zero.
Step 4: After the first stage, the algorithm is applied and implemented for the next stages.

3.2. Genetic algorithm (GA)

The basic GA includes four steps of initial population generation, selection, crossover, and mutation (Holland, 1975). In this study, for selection, the roulette wheel sampling mechanism is used. Also, four crossover operators of one-point (Holland, 1975), two-point (De Jong, 1975), arithmetic (Michalewicz & Schoenauer, 1996), and uniform (Syswerda, 1989) are used as well as nine mutation operators of swap (Larrañaga et al., 1999), big swap (Larrañaga et al., 1999), displacement (Michalewicz, 1996), inversion (Fogel, 1990), modified boundary, scramble (Syswerda, 1991), insertion (Fogel, 1988), random minor, random part.

3.3. Modified backtracking search algorithm (MBSA)

The backtracking search algorithm (BSA) was developed for the first time by Civicioglu (2013). This algorithm includes five stages of initialization, selection-1, mutation, crossover, and selection-2. In BSA, there is no guarantee that the mutation answer is feasible. Therefore, since the crossover operator may get some of the vectors from the mutation stage, it may affect the feasibility of some of the final answer vectors. To solve the problem, in this study MBSA is developed for the first time. In the proposed algorithm, the movement phase of the Electromagnetism-like mechanism (EM) algorithm is used to modify the mutation step. Also, in MBSA the algorithm frontier mechanism of BSA is not used. The stages of the proposed algorithm of MBSA are as follows, while its pseudo-code is presented in Appendix A.

3.3.1. Initialization

At this stage, an initial population P_{ij} is generated based on Uniform distribution by line 5′s equation where i and j denote the number of people and variables, respectively.

3.3.2. Selection-1

At this stage, MBSA generates a historical population of *oldP* according to line 6′s equation, which is used as the search direction. With each iteration, MBSA has a choice to redefine *oldP* through comparing the two generated random numbers of a and b by the "*if −then*" rule according to line 11's equation. After specifying the historical population, MBSA changes the people's order in *oldP* randomly by line 12's equation.

3.3.3. Mutation

At this stage, *Mutant* is generated by line 22′s equation, which is inspired by the "movement" phase of the EM algorithm, as the initial form of the under-trial population. Since there are some precautions taken to control the frontier in this step, the frontier control mechanism of the BSA algorithm is not used.

3.3.4. Crossover

At this stage, *Mutant* changes to the final under a trial population of T. First, T is equal to *Mutant*, then a binary integer-valued matrix named *map* is generated according to line 28 to select the people. If $map_{i,j} = 1$, then person $T_{i,j}$ is updated with $P_{i,j}$ that means$T_{i,j} = P_{i,j}$. The number of members that can participate in the crossover process is controlled by the controlling parameter of *mixrate*.

3.3.5. Selection-2

This stage is to update and record the better solution. At this step, the people of population T that have better goodness of fit than their peers of population P, are relocated to update P. The thorough minimum of all the people is also updated according to the goodness of fit for T and P.

3.4. Modified differential evolution algorithm (MDE)

Differential evolution (DE) is a metaheuristic algorithm inspired by nature by Storn and Price (1997). DE is an unconstrained algorithm and does not apply to constrained problems. This study addresses the issue of making the powerful DE algorithm applicable to constrained problems. Therefore, the MDE algorithm is developed for the first time. The main difference between DE and MDE lies in the kind and trend of their mutation. With the MDE algorithm, the mutation is inspired by the movement phase of EM. The stages of the proposed algorithm of MDE are as follows, while its pseudo-code is presented in Appendix B.

3.4.1. Initialization

At this stage, first, an initial population P_{ij} is generated based on Uniform distribution according to line 3 where i and j denote the number of people and variables respectively while all the answers are in the feasible region. As the next step, four members of the population are selected randomly according to line 10. Out of these four members, one member is selected as the target vector, and the others are selected as vectors one, two, and three randomly.

3.4.2. Mutation

At this stage, the movement phase of the EM algorithm is applied to modify the mutation of the DE algorithm to get to line 17 as the mutation operator of MDE. The parameter F is constant and between 0 and 1.

3.4.3. Crossover

At this stage, the mutated and target vectors are combined to generate the trial vector. The combination is performed based on the crossover probability coefficient, Cr, between 0 and 1. Line 25 shows this operator.

3.4.4. Selection

Line 31, chooses between the trial and target vectors. At this stage, the trial and target vectors are given values according to the objective function. Then the vector with a higher value becomes a member of the next generation.

3.5. Hybrid algorithms with restart mechanism

This technique is combined with GA, BSA, MBSA, and MDEA to improve their performance. The mentioned algorithms could find high-performance areas in the solution space in an acceptable time, but sometimes they are stuck in the local optimum. To get out of the local optimum with the algorithms, the restart mechanism is used. This method shocks the location in which the algorithm is in; and would cause more dispersions. In this study, based on the described mechanism to solve the model, the hybrid algorithm of genetic and restart mechanism (GA_Restart), the hybrid algorithm of backtracking search and restart mechanism (BSA_Restart), the hybrid algorithm of modified backtracking search and restart mechanism (MBSA_Restart), and the

Table 4
The generated test problems' size specifications.

Test Problems No.	I	J	K	L	M
TP-1	3	4	3	20	2
TP-2	5	7	6	30	4
TP-3	12	8	9	40	6
TP-4	13	10	11	50	7
TP-5	18	12	14	60	9
TP-6	20	14	16	70	10
TP-7	25	16	18	80	13
TP-8	30	17	20	90	14
TP-9	35	19	22	100	16
TP-10	40	22	24	110	18

hybrid algorithm of modified differential evolution and restart mechanism (MDE_Restart) are presented and used.

4. Experimental design

In this section, first, a set of random test problems with different sizes are generated. The Taguchi method selects the proposed algorithms' best parameters (Sadeghi-Moghaddam, Hajiaghaei-Keshteli, & Mahmoodjanloo, 2019). Then some experiments on the test problems are performed, and the results are compared.

4.1. Experimental problems

In this section, to generate test problems, we created ten problems of different sizes randomly. To do so, the necessary data are generated based on the uniform distribution. Here I, J, K, L, and M denote the number of suppliers, manufacturers, HCDC, customers, and recycling centers, respectively. The search time for all the experiments or combination of operators and parameters' levels is the same as $((I+J)+(J+K)+(K+L)+(L+K)+(K+M)+(M+J)+I)\times 100$. The search time changes according to the problem size. Table 4 shows the size of the test problems.

For each size, four different problems with different fixed costs ranges and the same variable cost range are considered. Therefore, there are a total of 40 problems that are solved with the eight proposed algorithms. Table 5 shows the fixed and variable costs. Table 6 shows the facilities' capacities, discount breakdown points, customers' demands, returned goods percentage, and unrecyclable goods rate.

Table 6
Parameters of the proposed model.

Parameters	Value
a_i	Uniform ~ [10000, 40000]
b_j	Uniform ~ [18000, 54000]
sc_k	Uniform ~ [18000, 72000]
e_m	Uniform ~ [6000, 24000]
$q2_{in}$	Uniform ~ [10000, 40000]
$q2_{i(n+1)}$	Uniform ~ [15000, 60000]
d_l	Uniform ~ [6000, 24000]
pc_l	Uniform ~ [0.01, 0.15]
pl_m	0.1
φ	10

4.2. The parameters and operators adjustment with Taguchi method

The correct selection of parameters and operators of metaheuristic algorithms impacted its performance considerably. In fact, if the algorithm's parameters are not set well, the algorithm would be inefficient (Babaee Tirkolaee, Goli, Faridnia, Soltani, & Weber, 2020).

In this study, Taguchi methods with the evolutionary and metaheuristic algorithms have proved an acceptable performance regarding the test problems (Chou, Chen, & Li, 2000; Tsai, Liu, & Chou, 2004). Table 7 shows the levels of all the parameters and operators.

After implementing the experiments, S/N (Eq. (29)) is applied to specify the parameters with the best solutions, while the greater S/N, the better the answer. As a sample, Fig. 3 shows the GA algorithm's averages of S/N for different factors' levels.

$$S/N = -10\log_{10}(objectivefunction) \tag{29}$$

Based on the calculations, the best level parameters of the algorithms are shown in Table 8.

4.3. Final experiments

Each generated test problem is applied five times independently. For better results, the parameters' best levels, according to Table 8 are applied. For all the algorithms, the calculation time of $(coefficient)\times$

Table 5
The fixed and variable costs of test problems.

Variable cost parameters	Lower bound	Upper bound	Fixed cost parameters	A Lower bound	A Upper bound	B Lower bound	B Upper bound	C Lower bound	C Upper bound	D Lower bound	D Upper bound
sx_{ij}	3	8	fx_{ij}	50	300	100	400	200	800	300	1200
t_{jk}	3	8	fy_{jk}	50	200	100	400	150	600	300	1200
u_{kl}	3	8	fz_{kl}	50	200	100	400	200	800	800	1600
ru_{lk}	3	8	frz_{lk}	50	200	100	400	200	800	800	1600
v_{km}	3	8	fo_{km}	50	200	100	400	200	800	800	1600
w_{mj}	3	8	frd_{mj}	50	200	100	400	200	800	800	1600
			f_j	2000	8000	4000	16,000	10,000	40,000	20,000	80,000
			g_k	3000	12,000	5000	20,000	20,000	80,000	40,000	160,000
			h_m	3000	15,000	5000	20,000	20,000	80,000	40,000	160,000

Table 7
The candidate levels for the proposed metaheuristic algorithms.

Algorithm	Factor	Levels
GA	A: Size of population (popsize)	50, 60, 70
	B: Probability of crossover (P_c)	0.1, 0.2, 0.3
	C: Probability of mutation (P_m)	0.05, 0.1, 0.15
	D: Type of crossover (T_c)	One-point, Two-point, Uniform, Arithmetic
	E: Type of mutation (T_m)	Swap, Big swap, Displacement, Inversion, Modified boundary, Scramble, Insertion, Random minor, Random part
GA_Restart	A: Size of population (popsize)	60, 70, 80
	B: Probability of crossover (P_c)	0.2, 0.25, 0.3
	C: Probability of mutation (P_m)	0.1, 0.15, 0.2
	D: Probability of restart ($P_{restart}$)	0.7, 0.8, 0.9
	E: Maximum lack of improvement (MaxLI)	1000, 1500, 2000
	F: Type of Crossover (T_c)	One-point, Two-point, Uniform, Arithmetic
	G: Type of mutation (T_m)	Swap, Big swap, Displacement, Inversion, Modified boundary, Scramble, Insertion, Random minor, Random part
BSA	A: Size of population (popsize)	30, 40, 50
	B: Amplitude of the search-direction matrix (FV)	2, 3, 4, 5, 6
	C: Probability of crossover (P_c)	0.2, 0.4, 0.6, 0.8, 1
BSA_Restart	A: Size of population (popsize)	60, 65, 70
	B: Amplitude of the search-direction matrix (FV)	2, 3, 4, 5, 6
	C: Probability of crossover (P_c)	0.2, 0.4, 0.6, 0.8, 1
	D: Probability of restart ($P_{restart}$)	0.7, 0.8, 0.9
	E: Maximum lack of improvement (MaxLI)	600, 700, 800
MBSA	A: Size of population (popsize)	30, 40, 50
	B: Amplitude of the search-direction matrix (FV)	2, 3, 4, 5, 6
	C: Probability of crossover (P_c)	0.2, 0.4, 0.6, 0.8, 1
MBSA_Restart	A: Size of population (popsize)	40, 50, 60
	B: Amplitude of the search-direction matrix (FV)	2, 3, 4, 5, 6
	C: Probability of crossover (P_c)	0.2, 0.4, 0.6, 0.8, 1
	D: Probability of restart ($P_{restart}$)	0.7, 0.8, 0.9
	E: Maximum lack of improvement (MaxLI)	1000, 2000, 3000
MDE	A: Size of population (popsize)	50, 60, 70
	B: Probability of crossover (P_c)	0.3, 0.5, 0.9
	C: Control factor (F)	0.5, 0.8, 1
MDE_Restart	A: Size of population (popsize)	30, 40, 50
	B: Probability of crossover (P_c)	0.2, 0.4, 0.9
	C: Control factor (F)	0.3, 0.4, 0.8
	D: Probability of restart ($P_{restart}$)	0.7, 0.8, 0.9
	E: Maximum lack of improvement (MaxLI)	350, 400, 450

$((I+J)+(J+K)+(K+L)+(L+K)+(K+M)+(M+J)+I)$ is considered while the coefficient is 3. Therefore, the bigger size of the problem, the greater time of calculation. Table 9, shows the five runs' averages while MDE_Restart is the superior algorithm with the best performance over 21 out of 40 problems. Finally, with problems 12, 6, and 1, MDE, BSA, and MBSA perform the best, respectively.

To compare the algorithms, the objective function value is the criterion. However, since the objective functions scales are different and therefore could not be compared directly, relative percentage deviation (RPD), Eq.(30), is applied for each problem.

After specifying the best level parameters and operators with the Taguchi method, the algorithms are compared based on the 40 experimental problems. Since the methods are random, each problem is implemented five times by each algorithm. After implementing the algorithms, the average RPD across the 10 problem sizes is calculated as is presented in Table 10 and Fig. 4. The table shows a meaningful difference between the algorithms' performances while the four algorithms of MDE_Restart, MDE, BSA, and BSA_Restart are the best. Fig. 5 shows the four superior algorithms.

Based on the results, four out of the best four algorithms are among the algorithms proposed in this study for the first time. The Restart mechanism combination with MDE and GA has improved the solutions considerably.

$$RPD = \frac{Alg_{sol} - Min_{sol}}{Min_{sol}} \times 100 \tag{30}$$

where Alg_{sol} and Min_{sol} are the objective function and the best solution, respectively.

4.4. Convergence chart

The convergence charts for the proposed metaheuristic algorithms are drawn in this way that the best-found objective is analyzed. The algorithms must have sufficient time for convergence to reach their best solution. After deriving the convergence charts for all the proposed metaheuristic algorithms, it was revealed that all the algorithms are convergent in a sensible amount of time. Fig. 6 shows the convergence chart.

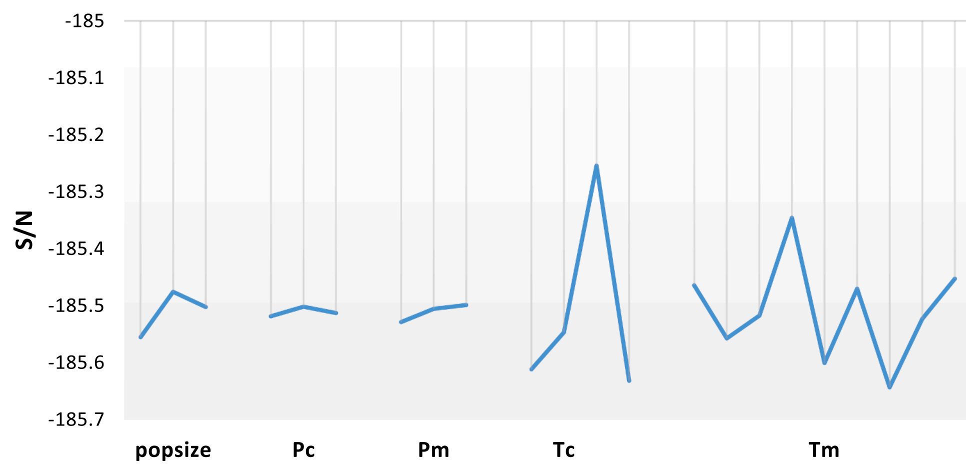

Fig. 3. The S/N averages for each factor's level in GA.

Table 8
The algorithms' best level parameters' values.

Algorithm	Parameters
GA	popsize = 60, $P_c = 0.2$, $P_m = 0.15$, T_c = Uniform, T_m = Inversion
GA_Restart	popsize = 60, $P_c = 0.25$, $P_m = 0.1$, T_c = Uniform, T_m = Scramble, $P_{restart} = 0.8$, MaxLI = 1000
BSA	popsize = 40, FV = 3, $P_c = 0.2$
BSA_Restart	popsize = 65, FV = 2, $P_c = 0.2$, $P_{restart} = 0.9$, MaxLI = 600
MBSA	popsize = 50, FV = 6, $P_c = 1$
MBSA_Restart	popsize = 40, FV = 6, $P_c = 1$, $P_{restart} = 0.7$, MaxLI = 2000
MDE	popsize = 70,$P_c = 0.3$, F = 0.5
MDE_Restart	popsize = 50,$P_c = 0.2$, F = 0.8, $P_{restart} = 0.9$, MaxLI = 350

4.5. Comparing the algorithms with ANOVA

After running the *meta*-heuristic algorithms and getting the RPD averages for the problems, to make a reliable statistical analysis to specify the significant differences between the algorithms' performances, a one-way analysis of variance (ANOVA) is applied with MINITAB software (Fasihi, Tavakkoli-Moghaddam, Najafi, & Hajiaghaei-Keshteli, 2021; Hamdi-Asl, Amoozad-Khalili, Tavakkoli-Moghaddam, & Hajiaghaei-Keshteli, 2021; Chouhan, Khan, & Hajiaghaei-Keshteli, 2022). H_0 checks if the RPD averages of the 8 algorithms are equal. H_1 checks if at least one algorithm has a different RPD average from the rest of the algorithms. The results are shown in Fig. 7. For a more accurate comparison of the results, the plot of the mean and the least significant difference (LSD) intervals at the 95% confidence level for the eight algorithms in Fig. 8 are presented.

According to Fig. 7, the resulting *P-Value* is zero which means H_0 did not receive any approval from the sample which led to its rejection. Therefore, at the significance level of 95%, H_0 is rejected and H_1 is accepted that means there is a significant difference between the algorithms in terms of the objective function. Since in this study the goal is to decrease the costs, according to Fig. 8, the best algorithms are the ones with smaller RPD averages.

4.6. Managerial implications

Managers should always seek the profitability of their organization. In this study, by focusing on the cost reduction of a CLSC, the final achievement would be the stronger profitability of the organization. The suppliers are offering quantitative discounts to encourage the manufacturers for bigger purchases, resulting in higher profitability. Also, the proposed model contributes to cost reduction through the hybrid centers of distribution and collection by avoiding extra building centers. Moreover, using recycled materials is also reduced the costs considerably.

Considering that the proposed CLSC focuses on both forward and reverse activities, it is quite compatible with different industries and sectors, including auto, battery, plastic, food, and chemicals manufacturing.

5. Conclusion and further study

This paper developed a multi-stage CLSC model in which the quantity discount and transportation fixed-charge are considered for the first time. The model minimizes the total cost, including fixed and variable costs between the facilities, facilities opening cost, landfilling cost, and the discounted purchase cost of the raw materials from the suppliers. Since the model is NP-Hard, the study tried to contribute to the literature by finding good solutions through eight basic and modified meta-heuristic algorithms. Forty test problems in different sizes were generated. Taguchi method was used to calibrate the proposed algorithms' parameters and operators before the final experiments of the test problems. The results showed that out of the eight proposed algorithms, MDE_Restart and MDE algorithms developed for the first time in this study had the best performances, respectively.

As the future directions, the following ideas can be considered and investigated. In this study, certainty in the parameters of the proposed model is considered. But the complexity of issues, the rapid pace of change, and today's turbulent environment have increased volatility and further uncertainties in data from financial markets and economies. The high level of supply chain uncertainty has hampered the ability of

Table 9
The final results of test problems with the proposed algorithms.

Proposed *meta*-heuristic Test Problem	GA	GA_Restart	BSA	BSA_Restart	MBSA	MBSA_Restart	MDE	MDE_Restart
TP-1-A	165083022.2	164961963.3	163658038.6	163658045.7	163658172.5	163658261.7	163658053.3	**163658038.5**
TP-1-B	280797307.9	280885183.1	280745743.4	280746322.3	280746839.6	280748508.7	280747334.6	**280743878.4**
TP-1-C	809886494.3	809781767.8	809660424.5	809,660,944	**809658020.8**	809666364.5	809663161.3	809,659,523
TP-1-D	1,889,037,889	1,893,373,745	1,888,491,669	1,888,495,597	1,888,493,784	1,889,585,770	1,888,493,408	**1,888,489,349**
TP-2-A	274089829.2	273686387.7	272670014.8	272673672.2	272674793.4	272681288.9	272669099.6	**272669065.8**
TP-2-B	490,911,617	489,559,196	488999660.7	489007560.3	489135592.2	489110220.4	488988847.4	**488986774.1**
TP-2-C	2,040,222,509	2,040,746,159	**2,039,333,622**	2,039,374,859	2,039,410,376	2,039,478,081	2,039,346,542	2,039,349,441
TP-2-D	3,684,668,137	3,684,736,242	3,682,256,169	3,682,268,469	3,682,418,637	3,682,450,283	**3,682,232,011**	3,682,233,759
TP-3-A	361,536,603	361626429.3	360954580.4	360976847.1	361106877.6	361202254.7	**360928268.1**	360933631.5
TP-3-B	831082710.9	830,803,697	830,370,487	830456421.6	830656212.1	830777217.1	**830356190.6**	830433867.7
TP-3-C	2,212,156,026	2,203,741,965	2,203,052,556	2,203,003,376	2,203,184,541	2,203,131,811	**2,202,961,383**	2,202,980,867
TP-3-D	5,551,977,338	5,551,961,912	5,550,766,269	5,550,847,942	5,550,852,613	5,550,994,906	5,550,643,390	**5,550,625,474**
TP-4-A	461559539.1	460838477.5	459964557.1	460001821.6	460417412.8	460730369.9	459906303.6	**459,895,782**
TP-4-B	817762308.4	818039182.7	**816801218.2**	816909267.3	817110668.4	817147957.7	816862973.5	816,812,719
TP-4-C	3,030,269,213	3,024,638,032	3,022,196,881	3,022,225,607	3,022,395,140	3,022,862,821	3,022,119,081	**3,022,091,901**
TP-4-D	5,271,785,620	5,263,472,771	**5,259,627,649**	5,259,744,091	5,259,987,769	5,260,654,793	5,259,745,838	5,259,760,468
TP-5-A	562,424,477	561123880.8	558170142.4	558244048.2	558583844.3	559184854.5	**557980414.2**	558,050,840
TP-5-B	1,138,607,023	1,136,265,644	1,134,273,984	1,134,311,298	1,135,313,154	1,137,238,546	1,134,151,538	**1,134,142,668**
TP-5-C	3,848,154,527	3,844,846,089	3,841,456,272	3,841,620,014	3,842,386,824	3,842,125,526	3,841,342,319	**3,841,316,884**
TP-5-D	5,324,919,815	5,321,045,794	5,310,940,279	5,311,119,571	5,311,482,574	5,311,258,677	5,310,929,536	**5,310,857,826**
TP-6-A	706502908.6	704568601.9	703330794.1	703387094.1	704423248.2	704824204.6	703077215.4	**703068953.4**
TP-6-B	1,371,968,270	1,368,400,757	1,364,632,344	1,364,820,358	1,365,746,539	1,366,875,294	**1,364,510,293**	1,364,554,138
TP-6-C	3,455,114,564	3,438,436,557	3,436,620,770	3,436,645,620	3,437,317,528	3,437,948,377	3,436,633,543	**3,436,535,418**
TP-6-D	7,399,308,563	7,381,233,084	7,361,058,515	7,361,436,938	7,366,790,573	7,376,842,212	7,361,033,898	**7,360,866,905**
TP-7-A	794921044.4	794586599.3	790607457.4	791158078.8	792168342.8	798179073.3	791035827.8	**790584238.2**
TP-7-B	1,261,985,022	1,259,685,491	1,254,242,223	1,254,619,314	1,258,137,952	1,262,917,500	**1,254,069,082**	1,254,159,744
TP-7-C	3,944,493,766	3,927,657,158	3,925,361,877	3,925,575,506	3,927,723,796	3,927,011,456	3,925,410,700	**3,925,253,704**
TP-7-D	9,671,894,230	9,687,872,394	9,638,856,034	9,639,189,360	9,643,613,759	9,647,242,893	9,640,082,901	**9,638,796,489**
TP-8-A	960463618.6	960,749,946	956591882.7	956948418.6	965277565.2	976114175.2	**956049609.9**	956114578.5
TP-8-B	1,291,400,277	1,292,030,974	1,285,286,805	1,285,455,826	1,288,465,412	1,308,998,598	**1,284,705,126**	1,284,861,255
TP-8-C	5,090,314,383	5,089,465,376	**5,060,837,027**	5,061,422,224	5,063,459,693	5,092,839,802	5,061,158,437	5,061,078,377
TP-8-D	11,327,717,020	11,277,545,941	**11,238,193,802**	11,238,812,845	11,272,022,795	11,314,293,981	11,238,610,457	11,238,443,551
TP-9-A	1,100,251,503	1,097,280,678	1,091,639,590	1,091,803,559	1,099,002,891	1,102,613,476	**1,091,376,059**	1,091,434,105
TP-9-B	1,580,832,165	1,578,342,394	1,572,039,825	1,572,291,796	1,587,213,529	1,602,641,277	**1,571,570,189**	1,571,671,650
TP-9-C	6,754,883,221	6,743,857,374	**6,731,921,349**	6,732,652,672	6,750,522,160	6,760,630,927	6,732,288,482	6,732,018,162
TP-9-D	10,406,301,405	10,401,358,168	10,365,738,811	10,366,146,639	10,418,846,949	10,496,609,411	10,366,021,949	**10,365,575,687**
TP-10-A	1,229,682,603	1,226,102,787	1,223,731,237	1,224,122,144	1,235,761,354	1,253,977,398	**1,223,051,858**	1,223,074,943
TP-10-B	1,779,999,459	1,767,124,474	1,762,001,937	1,762,554,674	1,784,926,326	1,837,721,143	1,762,579,842	**1,761,813,469**
TP-10-C	6,560,879,225	6,515,950,506	6,502,283,350	6,502,755,792	6,515,399,187	6,619,804,905	6,502,831,851	**6,502,139,244**
TP-10-D	12,021,837,244	12,041,778,690	11,982,639,660	11,982,860,289	12,019,767,360	12,340,044,424	11,982,947,074	**11,982,165,939**

Table 10
The algorithms' RPD average of the objective function for the best level parameters.

Problem Size	The problem's dimensions ($I \times J \times K \times L \times M \times I$)	The algorithms							
		GA	GA_Restart	BSA	BSA_Restart	MBSA	MBSA_Restart	MDE	MDE_Restart
1	3×4×3×20×2×3	0.237	0.281	0.001	0.001	0.001	0.016	0.001	0
2	5×7×6×30×4×5	0.257	0.158	0.002	0.003	0.011	0.012	0.001	0.001
3	12×8×9×40×6×12	0.182	0.084	0.011	0.015	0.032	0.043	0.008	0.010
4	13×10×11×50×7×13	0.252	0.135	0.011	0.017	0.049	0.074	0.010	0.008
5	18×12×14×60×9×18	0.414	0.264	0.018	0.024	0.068	0.135	0.006	0.009
6	20×14×16×70×10×20	0.531	0.214	0.019	0.026	0.103	0.177	0.008	0.007
7	25×16×18×80×13×25	0.510	0.388	0.012	0.039	0.166	0.456	0.025	0.008
8	30×17×20×90×14×30	0.597	0.501	0.032	0.049	0.409	1.332	0.009	0.013
9	35×19×22×100×16×35	0.540	0.380	0.020	0.032	0.627	1.181	0.009	0.009
10	40×22×24×110×18×40	0.707	0.320	0.023	0.041	0.722	2.914	0.020	0.006
	The RPD average	0.4227	0.2725	0.0149	0.0247	0.2188	0.634	0.0097	0.0071

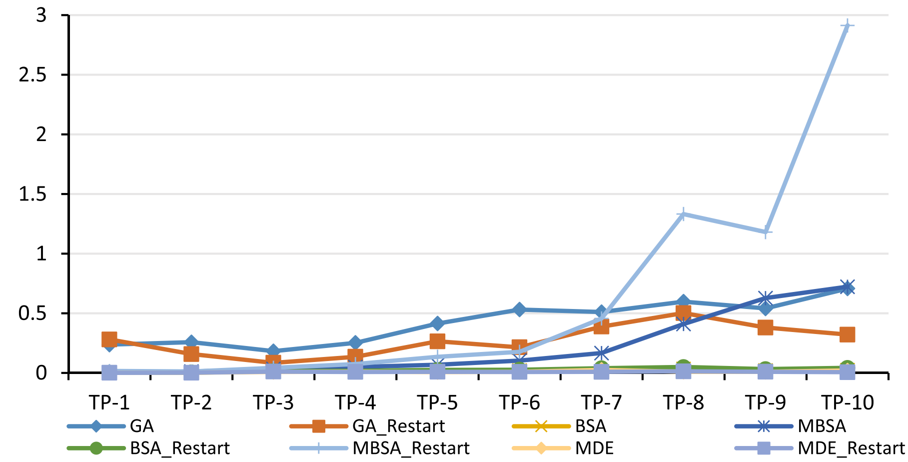

Fig. 4. The average RPD values obtained for each size of the test problems.

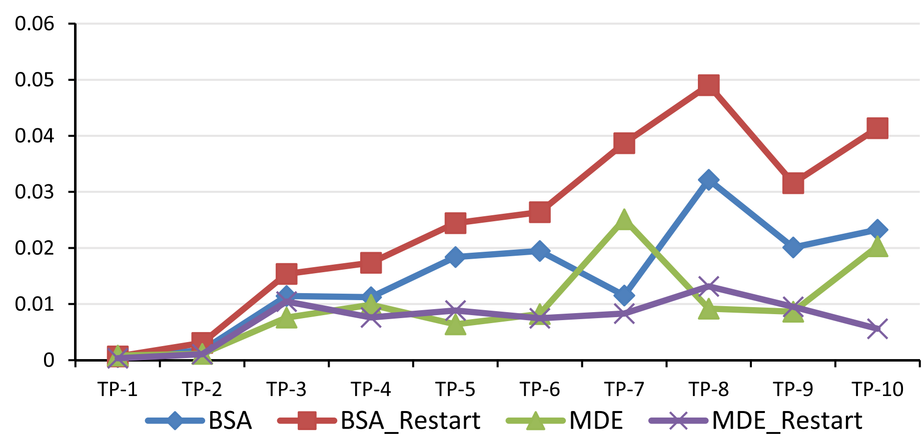

Fig. 5. The RPD averages for the top four algorithms.

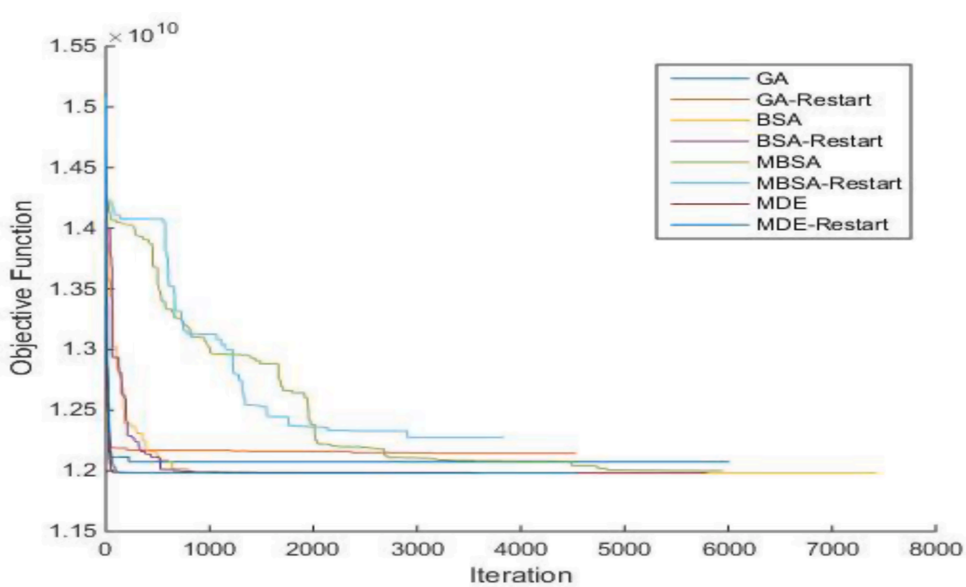

Fig. 6. Convergence graph of the proposed algorithms.

One-way ANOVA: GA, GA_Restart, BSA, BSA_Restart, MBSA, MBSA_Restart, MDE, MDE_Restart

Method

Null hypothesis	All means are equal
Alternative hypothesis	Not all means are equal
Significance level	α = 0.05

Equal variances were assumed for the analysis.

Factor Information

Factor	Levels	Values
Algorithms	8	BSA, BSA_Restart, GA, GA_Restart, MBSA, MBSA_Restart, MDE, MDE_Restart

Analysis of Variance

Source	DF	Adj SS	Adj MS	F-Value	P-Value
Algorithms	7	3.820	0.5457	4.36	0.000
Error	72	9.003	0.1250		
Total	79	12.823			

Means

Algorithms	N	Mean	StDev	95% CI
BSA	10	0.01490	0.00948	(-0.20801, 0.23781)
BSA_Restart	10	0.02470	0.01599	(-0.19821, 0.24761)
GA	10	0.4227	0.1806	(0.1998, 0.6456)
GA_Restart	10	0.2725	0.1293	(0.0496, 0.4954)
MBSA	10	0.2188	0.2686	(-0.0041, 0.4417)
MBSA_Restart	10	0.634	0.937	(0.411, 0.857)
MDE	10	0.00970	0.00754	(-0.21321, 0.23261)
MDE_Restart	10	0.00710	0.00396	(-0.21581, 0.23001)

Pooled StDev = 0.353612

Fig. 7. The outputs of ANOVA.

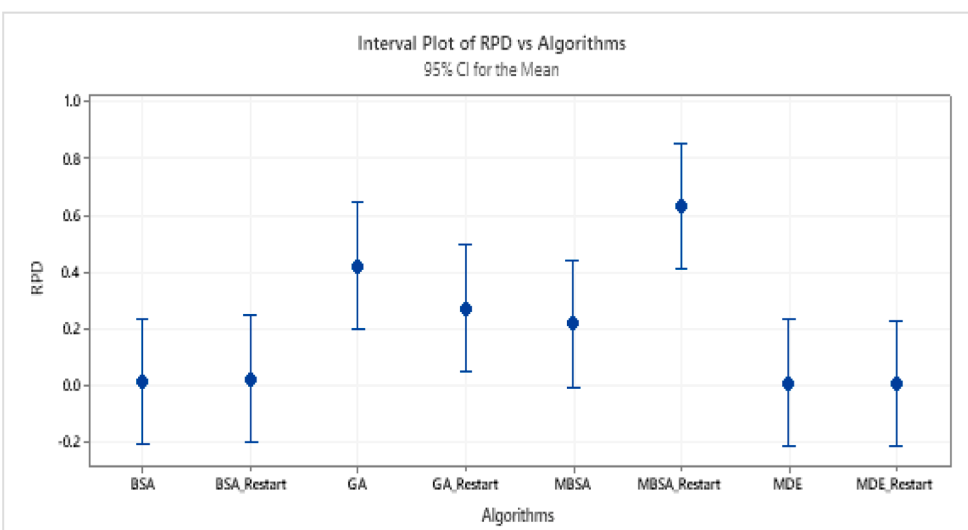

Fig. 8. Means plot and LSD intervals for the algorithms.

organizations to predict and decide for the future. Therefore, for better and more correct management and planning, attention to the issue of uncertainty in parameters such as demand, product prices, manufacturing costs, transportation costs, the amount of returned products, etc. in SC networks has become more important. In this regard, one can analyze our model in different uncertainties, such as what Özmen and Weber (2014), Roy, Maity, and Weber (2017), Kropat and Weber (2018), Khalilpourazari, Mirzazadeh, Weber, and Pasandideh (2019), Baltas et al. (2021) did in their research. In addition, the waste management (WM) system is an important and necessary problem in SC, especially in the case of COVID-19. The importance of WM causes many researchers, including Babaee Tirkolaee, Mahdavi, Esfahani, and Weber (2020), Babaee Tirkolaee, Abbasian, and Weber (2021), Akbarpour, Salehi-Amiri, Hajiaghaei-Keshteli, and Oliva (2021) offer approaches in this field. One can use these approaches to develop our proposed model. Another key issue in SC management is inventory management (IM), which aims to minimize investment in inventory while balancing supply and demand. Researchers such as Pervin, Roy, and Weber (2020), Savku and Weber (2020), Das, Pervin, Roy, and Weber (2021), Paul, Pervin, Roy, Maculan, and Weber (2021) have researched in this field that can be used to develop our model. Also, the developed model can be extended by considering environmental aspects and their impacts on costs to reach a green SC. The last but not least suggestion is to consider a three-dimension SC, like multi-product, multi-period, or multi-vehicle SC.

CRediT authorship contribution statement

Golara Chaharmahali: Conceptualization, Methodology, Software, Writing – original draft, Visualization. **Davoud Ghandalipour:** Conceptualization, Methodology, Software, Writing – original draft. **Milad Jasemi:** Writing – review & editing, Supervision. **Saber Molla-Alizadeh-Zavardehi:** Conceptualization, Validation, Supervision.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A

function MBSA (*ObjFun, N, D, maxcycle, low, up*)
Input: *ObjFun, N, D, maxcycle, mixrate*, $low_{1:D}$, $up_{1:D}$
Output: *globalminimum, globalminimizer*
// INITIALIZATION
$globalminimum = inf$
for i **from** 1 **to** N **do**
for j **from** 1 **to** D **do**
$P_{i,j} = rnd\cdot(up_i - low_j) + low_j$// Initialization of population, P
$oldP_{i,j} = rnd\cdot(up_i - low_j) + low_j$// Initialization of $oldP$
end
$fitnessP_i = ObjFun(P_i)$// Initial-fitness values of P
end
for *iteration* **from** 1 **to** *maxcycle* **do**
//SELECTION-I
if $(a < b|a.b \;\; U(0.1))$ **then** $oldP := P$ **end**
$oldP := permuting(oldP)$
Generation of Trial-Population
//MUTATION
for i **from** 1 **to** N **do**
$F = 3\cdot rndn$
$move = F\cdot(oldP_i - P_i)$
if $norm(move) > 0$
$move = move/norm(move)$
end
for j **from** 1 **to** D **do**
if $move_j > 0$
$mutant_{(i.j)} = P_{(i.j)} + move_j\cdot(1 - P_{(i.j)})$
else
$mutant_{(i.j)} = P_{(i.j)} + move_j\cdot(P_{(i.j)})$
end
end
end‘
// CROSSOVER
$Map_{1:N.l:D} = 1$ // Initial-map is an N-by-D matrix of ones
if $(c < d|c.d \;\; U(0.1))$ **then**
for i **from** 1 **to** N **do**
$map_{i.u_{(1:\lceil mixrate\cdot rnd\cdot D\rceil)}} = 0|u = permuting(< 1.2.3.\cdots.D >)$
end
else
for i **from** 1 **to** N **do**, $map_{i.randi(D)} = 0$, **end**
end
// Generation of Trial Population, T
$T := mutant$
for i **from** 1 **to** N **do**
for j **from** i **to** D **do**
if $map_{i,j} = 1$ **then** $T_{i,j} := P_{i,j}$
end
end

(*continued on next page*)

(*continued*)

```
    end
// SELECTION-II
fitnessT = ObjFn(T)
for i from 1 to N do
    if fitnessT_i < fitnessP_i then
        fitnessP_i := fitnessT_i
        P_i = T_i
    end
end
fitnessP_best = min(fitnessP) | best ∈ {1. 2. 3. ···. N}
if fitnessP_best < globalminimum then
    globalminimum := fitnessP_best
    globalminimizer := P_best
//Export globalminimum and globalminimizer
end
end
```

Appendix B

```
function MDE (ObjFun, N, D, maxcycle, low, up)
for i from 1 to N do
    for j from 1 to D do
        P_i,j = rnd·(up_j − low_j) + low_j // Initialization of population, P
    end
    fitnessP_i = ObjFun(P_i) // Initial-fitness values of P
end
for iteration from 1 to maxcycle do
    for i = 1 to N do
        select randomly r1 ≠ r2 ≠ r3 ≠ i
        move = F·(P_(r2,:) − P_(r3,:))
        if norm(move) > 0
            move = move/norm(move)
        end
    for j from 1 to D do
        if move_(j) > 0
        T_(i,j) = P_(r1,j) + move_(j)·(1 − P_(r1,j))
    else
        T_(i,j) = P_(r1,j) + move_(j)·(P_(r1,j))
        end
        end
    end
    for j from 1 to D do
        j_rand = rand(1.D)
        if (rand(0.1) < cr) or (j == jrand) then
            T_(i,j) = P_(i,j)
        end
    end
    fitnessT = ObjFn(T)
    for i = 1 to N do
        if fitnessT_i < fitnessP_i then
            newP_i = T_i
        else
            newP_i = P_i
        end
    end
    end
```

References

Akbarpour, N., Salehi-Amiri, A., Hajiaghaei-Keshteli, M., & Oliva, D. (2021). An innovative waste management system in a smart city under stochastic optimization using vehicle routing problem. *Soft Computing, 25*, 6707–6727. https://doi.org/10.1007/s00500-021-05669-6

Ali, S. S., Paksoy, T., Torğul, B., & Kaur, R. (2020). Reverse logistics optimization of an industrial air conditioner manufacturing company for designing sustainable supply chain: A fuzzy hybrid multi-criteria decision-making approach. *Wireless Networks, 26*, 5759–5782. https://doi.org/10.1007/s11276-019-02246-6

Babaee Tirkolaee, E., Abbasian, P., & Weber, G.-W. (2021). Sustainable fuzzy multi-trip location-routing problem for medical waste management during the COVID-19 outbreak. *Science of The Total Environment, 756*, Article 143607. https://doi.org/10.1016/j.scitotenv.2020.143607

Babaee Tirkolaee, E., Goli, A., Bakhsi, M., & Mahdavi, I. (2017). A robust multi-trip vehicle routing problem of perishable products with intermediate depots and time windows. *Numerical Algebra, Control & Optimization, 7*(4), 417–433. https://doi.org/10.3934/naco.2017026

Babaee Tirkolaee, E., Goli, A., Faridnia, A., Soltani, M., & Weber, G.-W. (2020). Multi-objective optimization for the reliable pollution-routing problem with cross-dock selection using Pareto-based algorithms. *Journal of Cleaner Production, 276*, Article 122927. https://doi.org/10.1016/j.jclepro.2020.122927

Babaee Tirkolaee, E., Mahdavi, I., Esfahani, M. M. S., & Weber, G.-W. (2020). A robust green location-allocation-inventory problem to design an urban waste management system under uncertainty. *Waste Management, 102*, 340–350. https://doi.org/10.1016/j.wasman.2019.10.038

Baltas, I., Dopierala, L., Kolodziejczyk, K., Szczepański, M., Weber, G.-W., & Yannacopoulos, A. N. (2021). Optimal management of defined contribution pension funds under the effect of inflation, mortality and uncertainty. *European Journal of Operational Research, 298*(3), 1162–1174. https://doi.org/10.1016/j.ejor.2021.08.038

Bertazzi, L., & Maggioni, F. (2018). A stochastic multi-stage fixed charge transportation problem: Worst-case analysis of the rolling horizon approach. *European Journal of Operation Research, 267*(2), 555–569. https://doi.org/10.1016/j.ejor.2017.12.004

Chouhan, V. K., Khan, S. H., & Hajiaghaei-Keshteli, M. (2021). Metaheuristic approaches to design and address multi-echelon sugarcane closed-loop supply chain network. *Soft Computing, 25*(16), 11377–11404. https://doi.org/10.1007/s00500-021-05943-7

Chouhan, V. K., Khan, S. H., & Hajiaghaei-Keshteli, M. (2022). Sustainable planning and decision-making model for sugarcane mills considering environmental issues. *Journal of Environmental Management, 303*, Article 114252. https://doi.org/10.1016/j.jenvman.2021.114252

Chou, J. H., Chen, S. H., & Li, J. J. (2000). Application of the Taguchi-genetic method to design an optimal grey-fuzzy controller of a constant turning force system. *Journal of Materials Processing Technology, 105*(3), 333–343. https://doi.org/10.1016/S0924-0136(00)00651-8

Civicioglu, P. (2013). Backtracking search optimization algorithm for numerical optimization problems. *Applied Mathematics & Computation, 219*(15), 8121–8144. https://doi.org/10.1016/j.amc.2013.02.017

Das, S. K., Pervin, M., Roy, S. K., & Weber, G. W. (2021). Multi-objective solid transportation-location problem with variable carbon emission in inventory management: A hybrid approach. *Annals of Operations Research, 1–27*. https://doi.org/10.1007/s10479-020-03809-z

De Jong, K. A. (1975). *An analysis of the behavior of a class of genetic adaptive systems*. University of Michigan. Doctoral dissertation.

Devika, K., Jafarian, A., & Nourbakhsh, V. (2014). Designing a sustainable closed-loop supply chain network based on triple bottom line approach: A comparison of metaheuristics hybridization techniques. *European Journal of Operation Research, 235* (3), 594–615. https://doi.org/10.1016/j.ejor.2013.12.032

El-Sayed, M., Afia, N., & El-Kharbotly, A. (2010). A stochastic model for forward-reverse logistics network design under risk. *Computers & Industrial Engineering, 58*(3), 423–431. https://doi.org/10.1016/j.cie.2008.09.040

Fasihi, M., Tavakkoli-Moghaddam, R., Najafi, S. E., & Hajiaghaei-Keshteli, M. (2021). Optimizing a bi-objective multi-period fish closed-loop supply chain network design by three multi-objective meta-heuristic algorithms. *Scientia Iranica*. https://doi.org/10.24200/SCI.2021.57930.5477

Fathollahi Fard, A. M., & Hajaghaei-Keshteli, M. (2018). A tri-level location-allocation model for forward/reverse supply chain. *Applied Soft Computing, 62*, 328–346. https://doi.org/10.1016/j.asoc.2017.11.004

Fleischmann, M., Beullens, P., Bloemhof-Ruwaard, J. M., & Van Wassenhove, L. N. (2001). The impact of product recovery on logistics network design. *Production & Operations Management, 10*(2), 156–173. https://doi.org/10.1111/j.1937-5956.2001.tb00076.x

Fogel, D. B. (1988). An evolutionary approach to the traveling salesman problem. *Biological Cybernetics, 60*, 139–144. https://doi.org/10.1007/BF00202901

Fogel, D. B. (1990). A parallel processing approach to a multiple traveling salesman problem using evolutionary programming. *In Proceedings of the fourth annual symposium on parallel processing* (pp. 318-326). Fullerton, CA.

Ghahremani-Nahr, J., Kian, R., & Sabet, E. (2019). A robust fuzzy mathematical programming model for the closed-loop supply chain network design and a whale optimization solution algorithm. *Expert Systems with Applications, 116*, 454–471. https://doi.org/10.1016/j.eswa.2018.09.027

Gholamian, N., Mahdavi, I., Mahdavi-Amiri, N., & Tavakkoli-Moghaddam, R. (2021). Hybridization of an interactive fuzzy methodology with a lexicographic min-max approach for optimizing a multi-period multi-product multi-echelon sustainable closed-loop supply chain network. *Computers & Industrial Engineering, 158*, Article 107282. https://doi.org/10.1016/j.cie.2021.107282

Gholizadeh, H., & Fazlollahtabar, H. (2020). Robust optimization and modified genetic algorithm for a closed loop green supply chain under uncertainty: Case study in Melting Industry. *Computers & Industrial Engineering, 147*, Article 106653. https://doi.org/10.1016/j.cie.2020.106653

Govindan, K., Darbari, J. D., Agarwal, V., & Jha, P. C. (2017). Fuzzy multi-objective approach for optimal selection of suppliers and transportation decisions in an eco-efficient closed loop supply chain network. *Journal of Cleaner Production, 165*, 1598–1619. https://doi.org/10.1016/j.jclepro.2017.06.180

Hajiaghaei-Keshteli, M., Molla-Alizadeh-Zavardehi, S., & Tavakkoli-Moghaddam, R. (2010). Addressing a nonlinear fixed-charge transportation problem using a spanning tree-based genetic algorithm. *Computers & Industrial Engineering, 59*(2), 259–271. https://doi.org/10.1016/j.cie.2010.04.007

Hamdi-Asl, A., Amoozad-Khalili, H., Tavakkoli-Moghaddam, R., & Hajiaghaei-Keshteli, M. (2021). Toward sustainability in designing agricultural supply chain network: A case study on palm date. *Scientia Iranica*. https://doi.org/10.24200/SCI.2021.58302.5659

Holland, J. (1975). Adaptation in natural and artificial systems: An introductory analysis with applications to biology. Control and artificial intelligence. University of Michigan Press.

Hosseini, S. M., Paydar, M. M., & Hajiaghaei-Keshteli, M. (2021). Recovery solutions for ecotourism centers during the Covid-19 pandemic: Utilizing fuzzy DEMATEL and fuzzy VIKOR methods. *Expert Systems with Applications, 185*, Article 115594. https://doi.org/10.1016/j.eswa.2021.115594

Kannan, G., Sasikumar, P., & Devika, K. (2010). A genetic algorithm approach for solving a closed loop supply chain model: A case of battery recycling. *Applied Mathematical Modelling, 34*(3), 655–670. https://doi.org/10.1016/j.apm.2009.06.021

Khalili Nasr, A., Tavana, M., Alavi, B., & Mina, H. (2021). A novel fuzzy multi-objective circular supplier selection and order allocation model for sustainable closed-loop supply chains. *Journal of Cleaner Production, 287*, Article 124994. https://doi.org/10.1016/j.jclepro.2020.124994

Khalilpourazari, S., Mirzazadeh, A., Weber, G.-W., & Pasandideh, S. H. R. (2019). A robust fuzzy approach for constrained multi-product economic production quantity with imperfect items and rework process. *Optimization, 69*(1), 63–90. https://doi.org/10.1080/02331934.2019.1630625

Kropat, E., & Weber, G.-W. (2018). Fuzzy target-environment networks and fuzzy-regression approaches. *Numerical Algebra, Control & Optimization, 8*(2), 135–155. https://doi.org/10.3934/naco.2018008

Larrañaga, P., Kuijpers, C. M. H., Murga, R. H., Inza, I., & Dizdarevic, S. (1999). Genetic algorithms for the travelling salesman problem: A review of representations and operators. *Artificial Intelligence Review, 13*, 129–170. https://doi.org/10.1023/A:1006529012972

Lotfi, R., Sheikhi, Z., Amra, M., AliBakhshi, M., & Weber, G.-W. (2021). Robust optimization of risk-aware, resilient and sustainable closed-loop supply chain network design with Lagrange relaxation and fix-and-optimize. *International Journal of Logistics Research and Applications*. https://doi.org/10.1080/13675567.2021.2017418

Lotfi, R., Zare Mehrjerdi, Y., Pishvaee, M. S., Sadeghieh, A., & Weber, G. W. (2021). A robust optimization model for sustainable and resilient closed-loop supply chain network design considering conditional value at risk. *Numerical Algebra, Control & Optimization, 11*(2), 221–253. https://doi.org/10.3934/naco.2020023

Michalewicz, Z. (1996). *Genetic algorithms + Data structures = Evolution programs*. New York: Springer.

Michalewicz, Z., & Schoenauer, M. (1996). Evolutionary algorithms for constrained parameter optimization problems. *Evolutionary Computation, 4*(1), 1–32. https://doi.org/10.1162/evco.1996.4.1.1

Midya, S., Roy, S. K., & Weber, G. W. (2021). Fuzzy multiple objective fractional optimization in rough approximation and its aptness to the fixed-charge transportation problem. *RAIRO Operations Research, 55*(3), 1715–1741. https://doi.org/10.1051/ro/2021078

Mohamadpour Tosarkani, B., & Hassanzadeh Amin, S. (2018). A possibilistic solution to configure a battery closed-loop supply chain: Multi-objective approach. *Expert Systems With Applications, 92*, 12–26. https://doi.org/10.1016/j.eswa.2017.09.039

Nayeri, S., Paydar, M. M., Asadi-Gangraj, E., & Emami, S. (2020). Multi-objective fuzzy robust optimization approach to sustainable closed-loop supply chain network design. *Computers & Industrial Engineering, 148*, Article 106716. https://doi.org/10.1016/j.cie.2020.106716

Özceylan, E., Paksoy, T., & Bektas, T. (2014). Modeling and optimizing the integrated problem of closed-loop supply chain network design and disassembly line balancing. *Transportation Research Part E: Logistics and Transportation Review, 61*, 142–164. https://doi.org/10.1016/j.tre.2013.11.001

Özmen, A., & Weber, G. W. (2014). RMARS: Robustification of multivariate adaptive regression spline under polyhedral uncertainty. *Journal of Computational and Applied Mathematics, 259*, 914–924. https://doi.org/10.1016/j.cam.2013.09.055

Paul, A., Pervin, M., Roy, S. K., Maculan, N., & Weber, G. W. (2021). A green inventory model with the effect of carbon taxation. *Annals of Operations Research, 28*, 1–16. https://doi.org/10.1007/s10479-021-04143-8

Rajabi, F., Najafi, S. E., Hajiaghaei-Keshteli, M., & Molla-Alizadeh-Zavardehi, S. (2013). Solving fuzzy step fixed charge transportation problems via metaheuristics. *International Journal of Research in Industrial Engineering, 2*(3), 24–34.

Ramezani, M., Bashiri, M., & Tavakkoli-Moghaddam, R. (2013). A robust design for a closed-loop supply chain network under an uncertain environment. *The International Journal of Advanced Manufacturing Technology, 66*, 825–843. https://doi.org/10.1007/s00170-012-4369-8

Pervin, M., Roy, S. K., & Weber, G. W. (2020). Deteriorating inventory with preservation technology under price and stock sensitive demand. *Journal of Industrial and Management Optimization, 16*(4), 1585–1612. https://doi.org/10.3934/jimo.2019019

Roy, S. K., Maity, G., & Weber, G.-W. (2017). Multi-objective two-stage grey transportation problem using utility function with goals. *Central European Journal of Operations Research, 25*(2), 417–439. https://doi.org/10.1007/s10100-016-0464-5

Sadeghi-Moghaddam, S., Hajiaghaei-Keshteli, M., & Mahmoodjanloo, M. (2019). New approaches in metaheuristics to solve the fixed charge transportation problem in a fuzzy environment. *Neural Computing & Applications, 31*, 477–497. https://doi.org/10.1007/s00521-017-3027-3

Sadeghi Rad, R., & Nahavandi, N. (2018). A novel multi-objective optimization model for integrated problem of green closed loop supply chain network design and quantity discount. *Journal of Cleaner Production, 196*, 1549–1565. https://doi.org/10.1016/j.jclepro.2018.06.034

Safaei, A. S., Roozbeh, A., & Paydar, M. M. (2017). A robust optimization model for the design of a cardboard closed-loop supply chain. *Journal of Cleaner Production, 166*, 1154–1168. https://doi.org/10.1016/j.jclepro.2017.08.085

Salehi-Amiri, A., Zahedi, A., Akbapour, N., & Hajiaghaei-Keshteli, M. (2021). Designing a sustainable closed-loop supply chain network for walnut industry. *Renewable & Sustainable Energy Reviews, 141*, Article 110821. https://doi.org/10.1016/j.rser.2021.110821

Salehi-Amiri, A., Zahedi, A., Gholian-Jouybari, F., Calvo, E. Z. R., & Hajiaghaei-Keshteli, M. (2022). Designing a closed-loop supply chain network considering social factors; A case study on avocado industry. *Applied Mathematical Modelling, 101*, 600–631. https://doi.org/10.1016/j.apm.2021.08.035

Salema, M. I. G., Barbosa-Povoa, A. P., & Novais, A. Q. (2007). An optimization model for the design for a capacitated multi-product reverse logistics networks with uncertainty. *European Journal of Operations Research, 179*(3), 1063–1077. https://doi.org/10.1016/j.ejor.2005.05.032

Savku, E., & Weber, G.-W. (2020). Stochastic differential games for optimal investment problems in a Markov regime-switching jump-diffusion market. *Annals of Operations Research*. https://doi.org/10.1007/s10479-020-03768-5

Shafei Kisomi, M., Solimanpur, M., & Doniavi, A. (2016). An integrated supply chain configuration model and procurement management under uncertainty: A set-based robust optimization methodology. *Applied Mathematical Modelling, 40*(17–18), 7928–7947. https://doi.org/10.1016/j.apm.2016.03.047

Soleimani, H. (2018). A new sustainable closed-loop supply chain model for mining industry considering fixed-charged transportation: A case study in a travertine quarry. *Resources Policy, 74*, Article 101230. https://doi.org/10.1016/j.resourpol.2018.07.006

Soleimani, H., Govindan, K., Saghafi, H., & Jafari, H. (2017). Fuzzy multi-objective sustainable and green closed-loop supply chain network design. *Computers & Industrial Engineering, 109*, 191–203. https://doi.org/10.1016/j.cie.2017.04.038

Soleimani, H., & Kannan, G. (2015). A Hybrid particle swarm optimization and genetic algorithm for closed-loop supply chain network design in large-scale networks. *Applied Mathematical Modeling, 39*(14), 3990–4012. https://doi.org/10.1016/j.apm.2014.12.016

Soleimani, H., Seyyed-Esfahani, M., & Akbarpour Shirazi, M. (2013). Designing and planning a multi-echelon multi-period multi-product closed-loop supply chain utilizing genetic algorithm. *The International Journal of Advanced Manufacturing Technology, 68*, 917–931. https://doi.org/10.1007/s00170-013-4953-6

Storn, R., & Price, K. (1997). Differential evolution-A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization, 11*(4), 341–359. https://doi.org/10.1023/A:1008202821328

Syswerda, G. (1989). Uniform crossover in genetic algorithms. *In Proceedings of the third international conference on genetic algorithms* (pp. 2-9). San Mateo, CA.

Syswerda, G. (1991). Schedule optimization using genetic algorithms. In L. Davis (Ed.), *Handbook of genetic algorithms* (pp. 332–349). New York: Van Nostrand Reinhold.

Taheri-Bavil-Oliaei, M., Zegordi, S. H., & Tavakkoli-Moghaddam, R. (2021). Bi-objective build-to-order supply chain network design under uncertainty and time-dependent demand: An automobile case study. *Computers & Industrial Engineering, 154*, Article 107126. https://doi.org/10.1016/j.cie.2021.107126

Tsai, J. F. (2007). An optimization approach for supply chain management models with quantity discount policy. *European Journal of Operational Research, 177*(2), 982–994. https://doi.org/10.1016/j.ejor.2006.01.034

Tsai, J. T., Liu, T. K., & Chou, J. H. (2004). Hybrid Taguchi-genetic algorithm for global numerical optimization. *IEEE Transactions on Evolutionary Computation, 8*(4), 365–377. https://doi.org/10.1109/TEVC.2004.826895

Wang, H. F., & Hsu, H. W. (2010). A closed-loop logistic model with a spanning-tree based genetic algorithm. *Computers & Operations Research, 37*(2), 376–389. https://doi.org/10.1016/j.cor.2009.06.001

Zeballos, L. J., Mendez, C. A., Barbosa-Povoa, A. P., & Novais, A. Q. (2014). Multi-period design and planning of closed-loop supply chains with uncertain supply and demand. *Computers & Chemical Engineering, 66*, 151–164. https://doi.org/10.1016/j.compchemeng.2014.02.027