

# Green supply chain flow analysis with multi-attribute demand in a multi-period product development environment

Sahar Porkar<sup>1</sup> · Iraj Mahdavi<sup>1</sup> · Behzad Maleki Vishkaei<sup>1</sup> · Milad Hematian<sup>1</sup>

Received: 29 April 2017 / Revised: 21 November 2017 / Accepted: 25 January 2018 /  
Published online: 7 February 2018  
© Springer-Verlag GmbH Germany, part of Springer Nature 2018

**Abstract** Green supply chain management is based on performing environmental management into supply chain network in order to decrease the environmental side effects in the product life cycle. So, in this paper, a bi-objective nonlinear programming for an integrated forward/reverse logistics network with the aim of increasing total profit of the network and maximizing the score of green design and quality indicators and green scrap score is discussed. Quality and green design indicators are considered for the forward network and the green scrap score are defined for scrapped products collected and disassembled in the reverse network. The mentioned network includes three echelons in the forward flow (production centers (factories), distribution centers, and customer zones) and three echelons in the reverse flow (collection/disassembly centers, recycling centers, and disposal centers). The main contribution of this study is to consider green design indicators and quality indicators for developing final products. Furthermore, in the model of this research, some constraints are regarded to determine the amount of paper and plastic consumption in the packaging of the products. After presenting the considered model, Multi-objective Particle Swarm Optimization and Non-dominated Sorting Genetic (NSGA-II) meta-heuristic Algorithms are proposed to find a set of Pareto-optimal solutions. Then, their output is compared through some samples with different sizes to prove their workability.

**Keywords** Green supply chain · Green design indicators · Quality indicators · Meta-heuristic algorithm · Taguchi experimental design

---

✉ Iraj Mahdavi  
irajarash@rediffmail.com

<sup>1</sup> Department of Industrial Engineering, Mazandaran University of Science and Technology, Babol, Iran

## 1 Introduction

The increase of knowledge and consciousness of the buyers about the advantages of using environmental-friendly products in recent decades has prompted producers and executors to pay more attention to green supply chain management. Today, green supply chain managers in pioneering companies attempt to use green logistics and their environmental performance improvement as a strategic tool to achieve stable competitive advantage through making environmental utility and satisfaction in the whole supply chain; they establish their goals based on three important issues: green design (product), green production (process), and product recovery (Boks and Stevels 2007). Green design is a multi-aspects district and needs different expertise contexts like environmental risk management, safety of product, pollution prevention, keep the resources and residue management (Srivastava 2007). Over the last two decades, many companies such as Kodak and Xerox have focused on remanufacturing and recovery activities and have achieved significant successes in this area (Üster et al. 2007).

Today, a reduction in the amount of packaging used in products has been considered as a goal by some companies like Wal Mart and Sisko to provide natural resources protection and cost savings. The industrial sector in developed countries, such as in United States, European Union and Japan, has been enforced to adopt green supply chains because of government regulation on environmental subjects (Abu Seman et al. 2012).

Successful acceptance and adoption of the reverse logistics relies on various inter-organization factors such as an obligation to conserve the environment, code of ethics and the existence of sponsors and/or policy makers with an obligation to adopt environmental-friendly policies (Richey et al. 2005; Murphy and Poist 2003). Lu and Bostel (2007) proposed four types of main reverse logistics networks included Directly Reusable Network (DRN), Remanufacturing Network (RMN), Repair Service Network (RSN) and Recycling Network (RN).

Traditional supply chain network design in the forward flow executes location, the determination of the capacity and the number of production, manufacture and distribution facilities, and the definition of the types of transportation communication and fittings (Fleischmann et al. 1997). In the reverse networks and closed-loop networks, other facilities such as collection centers, sorting centers and reprocessing centers are added to the existing facilities in the forward network. On the other hand, the reverse and forward network design separately causes the network sub-optimality according to the purposes of the supply chain (Baumgarten et al. 2003; Schultmann et al. 2006).

Many efforts to model and optimize the supply chain network design problems have been done. Most of them are based on single-objective such as costs minimization and/or profit maximization (Govindan et al. 2015). Therefore, recently, two or more distinctive objectives included objectives to conserve the environmental interests in addition to the objectives mentioned above are considered to design green supply chain networks.

According to the discussion mentioned in this research, a bi-objective nonlinear programming model is presented for a reverse and forward supply chain network. In the considered network, three echelons included production centers, distribution centers and customer zones in the forward flow and three echelons included collection/disassembly centers, recycling centers and disposal centers in the reverse flow have been regarded. The developed model is a multi-echelon, multi-product, and multi-period model.

The rest of the paper is organized as follows. Section 2 gives the relevant literature review. Section 3 introduces the problem description and the determination of assumptions. A bi-objective nonlinear programming model is developed for the considered problem in Sect. 4. Sections 5 and 6 present the proposed solution method, solve some sample problems and evaluate the results. Finally, Sect. 7 is devoted to conclusions and future works.

## 2 Literature review

Recently, there has been an increasing interest in reverse logistics due to growing environmental concern and economic opportunities related to cost savings and revenues of returned products (Entezaminia et al. 2016). In many industries, reverse logistics can be implemented as a sustainable and profitable business strategy to allow the improvement of futures sales and customer loyalty (Roghianian and Pazhoheshfar 2014).

In the real world, when both reverse and forward networks are integrated and considered together, then a closed-loop supply chain is created (Ramezani et al. 2013). According to this matter, Pishvae et al. (2010) presented a bi-objective integrated logistics network design problem and proposed a memetic algorithm to solve it. Tarokh and Naseri (2012) studied a distribution network for a multi-echelon supply chain. They developed a mixed integer programming model and minimized the costs of the distribution network in the studied problem. They proposed a hybrid meta-heuristic method as a combination of Simulated Annealing and Genetic algorithms to solve the model. Alfonso-Lizarazo et al. (2013) presented a mathematical model in order to represent the dynamic interaction between flows in a closed-loop supply chain network. The objective function considered energy, cost and economic profits. Finally, the results were analyzed using proper statistical tools. Soleimani et al. (2013) developed a multi-echelon, multi-product, and multi-period model in a MILP structure. Their proposed model was solved by CPLEX optimization software and through a developed GA. Their results demonstrated the acceptable performances of the developed GA. Özceylan et al. (2014) developed a mixed integer nonlinear programming model that jointly optimized the strategic and tactical decisions of a closed-loop supply chain. The objective was to minimize costs. Pazhani et al. (2013) presented a two-objective network design problem for the multi-period, multi-product closed-loop supply chain in order to minimize the costs and maximize the efficiency. Their model was a two-objective MILP model to assist in the decision making on: (1) Operational/Location decisions for warehouses, combined facilities and manufacturing facilities, and (2) production and distribution

of products between stages in supply chain. Soleimani and Kannan (2015) developed a deterministic multi-product, multi-echelon, multi-period model for a closed-loop supply chain network. They considered two meta-heuristic algorithms to develop a new elevated hybrid algorithm: the Genetic Algorithm and Particle Swarm Optimization. They investigated a case study to evaluate their proposed approach. El-Sayed et al. (2010) developed a stochastic model for the closed-loop network. In their paper, it was assumed that the demand was a non-deterministic parameter and the model was designed for a multi-product situation. They investigated uncertainty in demand, return and cost. Gaur et al. (2017) developed a closed loop supply chain model which is included two decision level: tactical and strategic. A mixed-integer nonlinear programming model is proposed then the model reformulate as a mixed integer linear model using augmented penalty approach. Finally, they applied a battery manufacturer as a case study to indicate model validation. Dondo and Méndez (2016) proposed a multi-echelon forward and reverse supply chain in operational level. In this paper, the forward and backward flows on a supply chain network according to greenness aspects are computed to minimize total costs. They used decomposition approach to find near optimal solution for the problem.

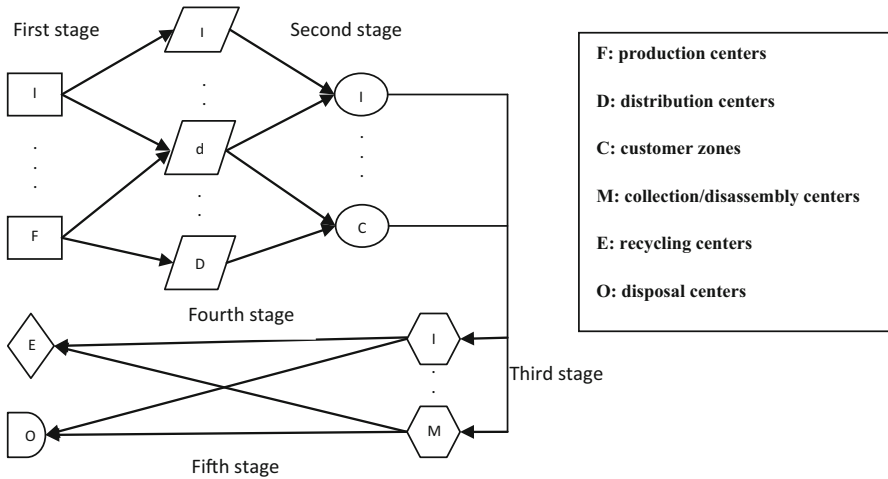
Furthermore, in several researches, greenness is considered as a measurement in supply chain objectives. Wang et al. (2011) studied a supply chain design problem according to the environment. In the design phase, they paid attention to the environmental investment decisions and developed a multi-objective model with the aim of minimizing the total cost and environmental effects. Comparison of the results of this model with the real numerical data demonstrated that their considered model can be used as a suitable tool for the green supply chain network programming. Giarola et al. (2011) proposed a MILP framework to optimize environmental financial performance indicators in a multi-period and multi-echelon biofuels supply chain. Because of the environmental rules importance for the green supply chain network, Yeh and Chuang (2011) presented a mathematical programming model with four criteria: cost, time, quality of product, and green evaluation degree. They proposed two multi-objective Genetic Algorithms based on Pareto archive to find optimal solutions. Jamshidi et al. (2012) combined total cost and environmental effect components in a multi-objective approach and used a memetic algorithm to solve it. Pishvae and Razmi (2012) presented a fuzzy multi-objective model for a reverse supply chain problem with the aim of minimizing costs and environmental effect. The considered model was single-product and its efficiency was proved by a real-world instance. Ghayebloo et al. (2015) presented a two-objective model with the aim of maximizing the profit of the whole network as the first objective and maximizing the amount of total greenness such as components and products greenness as the second objective. Their supply chain included suppliers, assembly facilities, consumption centers (customers), disassembly facilities, and recycling centers. They compared the weighted sums method with the limited constraints method to investigate their proposed model. Zohal and Soleimani (2016) developed a multi-objective integer linear programming model to minimize costs and CO<sub>2</sub> emission for a case study in gold industry and proposed a 7-layer integrated forward/reverse logistics network design. Then, in order to solve

the model, an algorithm based on ant colony optimization was developed. The performance of the proposed algorithm has been compared with the optimum solutions of the LINGO software through various numerical examples. Further, they used a Taguchi method to calibrate the parameters of the algorithm. Tiwari et al. (2016) studied a closed loop green supply chain in semiconductor industries. They consider two conflict objective included maximizing total profit and minimizing pollution simultaneously. For this purpose, a hybrid model combining estimation of distribution algorithm and territory defined multi-objective algorithm is developed. Pedram et al. (2017) studied closed loop green supply chain including forward and reverse logistics. The proposed model considered uncertainty in demand, return products and their quality. To achieve this goal, a mixed integer linear programming model is developed. In order to solve the problem a scenario based approach are applied. They consider two main objectives contain maximizing profit and minimizing pollution. A tire case study is used to show model applicability. Soleimani et al. (2017) presented a new multi-objective green forward backward supply chain model under uncertainty. The forward supply chain includes supplier, manufacturers, distribution centers, customers and backward one contains return and recycling centers. A scenario analysis is used to solve the problem. They also develop a genetic algorithm to optimize total profit and reduction of lost working. Mohammed et al. (2017) proposed a multi-period closed loop green supply chain model. In this research, demand and return products are considered uncertain. A mixed integer linear programming is developed to minimize the total system cost. A robust approach is used to overcome the problem uncertainty.

According to the researches accomplished over the current issue, the main contribution of our work is to consider distinctive characteristics for the model presented in this paper, that is, the multi-product, multi-period integrated reverse and forward networks design, and finally, green design indicators and quality indicators as the most significant features for the development of final products. Furthermore, in the proposed model, some constraints are considered to determine the amount of paper and plastic consumption in the packaging of the products.

### 3 Problem definition

In this research, an integrated forward/reverse supply chain network is studied. It includes production centers (factories), distribution centers (warehouses) and customer zones in the forward flow and collection/disassembly centers, recycling centers and disposal centers in the reverse flow. As shown in Fig. 1, after producing the products in the factories, these products are delivered to the distribution centers (warehouses) and then transferred to customer zones. Because the considered model is multi-period, some final products in each period are held in distribution centers as inventory. Then, scrapped products are collected from customer zones. After disassembling them, two flows of components are made: (1) recyclable components which are sent to recycling centers; (2) disposable components which are sent to disposal centers. Two sets of indicators as green design of product and quality of product are proposed in order to present and develop the manufactured products in



**Fig. 1** An integrated forward/reverse supply chain network

this network. The set of green design indicator includes a design for: (1) easier disassembly of product; (2) utilization of reusable (recyclable) raw materials (components); (3) utilization of environmental-friendly (safe disposal) raw materials (components). Also, the set of quality indicator includes: (1) utilization of good quality components in the production of products; (2) utilization of more persistent components with more longevity in the production of products. Furthermore, the model presented in this research is a bi-objective model that its first objective is to maximize total profit of the network and its second objective is to maximize the sum of the score of final products produced by green design indicators and quality indicators in the forward network and green scrap score of scrapped products collected and disassembled in the reverse network. Moreover, a new parameter as green scrap score for scrapped products is also considered. Because the score of final products of mentioned indicators is increasing during the periods, the scrap of products returned from customer zones has more environmental advantage in comparison with the previous periods. So, this matter is posed by dedicating the green scrap score to these products.

The basic assumptions of the proposed model are as follows:

- The studied green supply chain network is multi-product and multi-period.
- The number of production centers (factories), distribution centers (warehouses), customer zones, collection/disassembly centers, recycling centers and disposal centers is identified and their locations are fixed and known.
- All parameters of the model are deterministic and definite.
- All of the customers' demands must be satisfied. In each period, customers' demands for products change according to the set of green design indicator of product and the set of quality indicator of product. Moreover, all of the products returned from customers must be collected in the reverse flow.

- As all demands must be satisfied, there is no need for production centers to keep inventory and they produce the amount needed.
- As the transportation vehicles are the same, the average transportation cost of all types of recyclable and disposable components are identical. To forbid extra unnecessary indices the average selling price of all types of recyclable components, and the average disposal cost of all types of disposable components don't rely on the type of recyclable and disposable components.
- Average disposal cost of each unit of disposable component is included in transportation cost of components to disposal centers.
- Purchasing cost of each unit of scrapped product from customer zones is assumed negligible.
- The score of products in mentioned indicators is increasing during the periods.
- The reverse flow is added to the network from period 2, because the return of the products is started from period 2. Then, the reverse logistics income and costs are added to the system from period 2.
- Production centers, distribution centers, collection/disassembly centers, recycling centers, and disposal centers have a limited capacity.
- There is not any initial inventory at the beginning of the planning horizon.

#### 4 Proposed mathematical programming model

The following notations are used in the formulation of the proposed programming model to study the flow of an integrated forward/reverse network:

##### *Indices*

- F Set of production centers ( $f = 1, 2, \dots, F$ )  
 D Set of distribution centers ( $d = 1, 2, \dots, D$ )  
 C Set of customer zones ( $c = 1, 2, \dots, C$ )  
 M Set of collection/disassembly centers ( $m = 1, 2, \dots, M$ )  
 E Set of recycling centers ( $e = 1, 2, \dots, E$ )  
 O Set of disposal centers ( $o = 1, 2, \dots, O$ )  
 I Set of products types ( $i = 1, 2, \dots, I$ )  
 N Set of planning periods ( $n = 1, 2, \dots, N$ )  
 G Set of green design indicators of product ( $g = 1, 2, 3$ ) which includes: easier disassembly of product, utilization of reusable raw materials and Utilization of environmental-friendly raw materials respectively  
 Q Set of quality indicators of product ( $q = 1, 2$ ) which includes: utilization of good quality components in the production of products, utilization of more persistent components with more longevity in the production of products respectively

##### *Parameters*

$D_{igqcn}$	Demand of product $i$ with green design indicator $g$ and quality indicator $q$ in customer zone $c$ in period $n$
$C1_{igqfn}$	Production cost per unit of product $i$ with green design indicator $g$ and quality indicator $q$ in production center $f$ in period $n$
$C2_{ifdn}$	Transportation cost per unit of product $i$ from production center $f$ to distribution center $d$ in period $n$
$C3_{idcn}$	Transportation cost per unit of product $i$ from distribution center $d$ to customer zone $c$ in period $n$
$C4_{icmn}$	Transportation cost per unit of scrapped product $i$ from customer zone $c$ to disassembly center $m$ in period $n$
$C5_{men}$	Average transportation cost per unit of recyclable component from disassembly center $m$ to recycling center $e$ in period $n$
$C6_{mon}$	Average transportation cost per unit of disposable component from disassembly center $m$ to disposal center $o$ in period $n$
$C7_{igqmn}$	Disassembly cost per unit of scrapped product $i$ with green design indicator $g$ and quality indicator $q$ in disassembly center $m$ in period $n$
$C8_{igqdn}$	Holding cost per unit of product $i$ with green design indicator $g$ and quality indicator $q$ in distribution center $d$ in period $n$
$R_{igfn}$	Score of each unit of product $i$ with green design indicator $g$ in production center $f$ in period $n$
$R'_{iqfn}$	Score of each unit of product $i$ with quality indicator $q$ in production center $f$ in period $n$
$u_i$	Weight of each unit of product $i$ with any green design indicator
$(1 - u_i)$	Weight of each unit of product $i$ with any quality indicator
$w'$	Weight factor (significant) for quality and environmental advantage in the forward network, in the second objective function
$(1 - w')$	Weight factor (significant) for environmental advantage in the reverse network, in the second objective function
$h_{igqn}$	Green scrap score of each unit of scrapped product $i$ with green design indicator $g$ and quality indicator $q$ in period $n$
$V1_{igqfn}$	Capacity of the production of product $i$ with green design indicator $g$ and quality indicator $q$ in production center $f$ in period $n$
$V2_{igqdn}$	Capacity of distribution center $d$ for the acceptance of product $i$ with green design indicator $g$ and quality indicator $q$ in period $n$
$V3_{igqmn}$	Capacity of disassembly center $m$ for disassembly of product $i$ with green design indicator $g$ and quality indicator $q$ in period $n$
$v4_{en}$	Acceptance capacity of recycling center $e$ in period $n$
$v5_{on}$	Acceptance capacity of disposal center $o$ in period $n$
$P_{igqcn}$	Selling price per unit of product $i$ with green design indicator $g$ and quality indicator $q$ in customer zone $c$ in period $n$
$p'_{en}$	Average selling price per unit of recyclable component in recycling center $e$ in period $n$
$w_{igq}$	Average total number of components per unit of product $i$ with green design indicator $g$ and quality indicator $q$



$\gamma_{1igqn}$	Expected percentage of recyclable components obtained by disassembling scrapped product $i$ with green design indicator $g$ and quality indicator $q$ in period $n$
$r_{igqcn'n}$	Return rate of scrapped product $i$ in period $n$ with green design indicator $g$ and quality indicator $q$ returned from customer zone $c$ and period $n'$
$sh_i$	Maximum quota of paper consumption determined by government for packaging of each unit of product $i$
$pl_i$	Maximum quota of plastic consumption determined by government for packaging of each unit of product $i$
$c_n$	Considered pecuniary penalty/cost discount per unit of more/less paper consumption than maximum quota determined by government in period $n$
$k_n$	Considered pecuniary penalty/cost discount per unit of more/less plastic consumption than maximum quota determined by government in period $n$
$\rho_i$	Percentage of paper consumption in the first period for packaging of each unit of product $i$ which determines minimum consumption in the last planning period (to keep the quality of packaging)
$\mu_i$	Percentage of plastic consumption in the first period for packaging of each unit of product $i$ which determines minimum consumption in the last planning period (to keep the quality of packaging)

#### *Decision variables*

$x1_{igqfn}$	Quantity of product $i$ with green design indicator $g$ and quality indicator $q$ produced in production center $f$ in period $n$
$x2_{igqfdn}$	Quantity of product $i$ with green design indicator $g$ and quality indicator $q$ transferred from production center $f$ to distribution center $d$ in period $n$
$x3_{igqdcn}$	Quantity of product $i$ with green design indicator $g$ and quality indicator $q$ transferred from distribution center $d$ to customer zone $c$ in period $n$
$x4_{igqcmn}$	Quantity of scrapped product $i$ with green design indicator $g$ and quality indicator $q$ transferred from customer zone $c$ to disassembly center $m$ in period $n$
$x5_{igqmen}$	Quantity of recyclable component obtained by disassembling product $i$ with green design indicator $g$ and quality indicator $q$ transferred from disassembly center $m$ to recycling center $e$ in period $n$
$x6_{igqmon}$	Quantity of disposable component obtained by disassembling product $i$ with green design indicator $g$ and quality indicator $q$ transferred from disassembly center $m$ to disposal center $o$ in period $n$
$x7_{igqmn}$	Quantity of product $i$ with green design indicator $g$ and quality indicator $q$ disassembled in disassembly center $m$ in period $n$
$x8_{igqdn}$	Inventory of product $i$ with green design indicator $g$ and quality indicator $q$ in distribution center $d$ in period $n$
$y_{igqfn}$	Quantity of paper consumption needed for packaging of each unit of product $i$ with green design indicator $g$ and quality indicator $q$ in production center $f$ in period $n$

$Z_{igqfn}$  Quantity of plastic consumption needed for packaging of each unit of product  $i$  with green design indicator  $g$  and quality indicator  $q$  in production center  $f$  in period  $n$

According to the notations presented above, the mathematical programming model to design the considered green supply chain network can be formulated as follows:

$$\begin{aligned}
 \max Z_1 = & \sum_{i=1}^I \sum_{g=1}^G \sum_{q=1}^Q \sum_{d=1}^D \sum_{c=1}^C \sum_{n=1}^N p_{igqcn} \cdot x3_{igqdcn} + \sum_{i=1}^I \sum_{g=1}^G \sum_{q=1}^Q \sum_{m=1}^M \sum_{e=1}^E \sum_{n=2}^N p'_{en} \cdot x5_{igqmen} \\
 & - \sum_{i=1}^I \sum_{g=1}^G \sum_{q=1}^Q \sum_{f=1}^F \sum_{n=1}^N c1_{igqfn} \cdot x1_{igqfn} - \sum_{i=1}^I \sum_{g=1}^G \sum_{q=1}^Q \sum_{f=1}^F \sum_{d=1}^D \sum_{n=1}^N c2_{ifdn} \cdot x2_{igqfdn} \\
 & - \sum_{i=1}^I \sum_{g=1}^G \sum_{q=1}^Q \sum_{d=1}^D \sum_{c=1}^C \sum_{n=1}^N c3_{idcn} \cdot x3_{igqdcn} - \sum_{i=1}^I \sum_{g=1}^G \sum_{q=1}^Q \sum_{c=1}^C \sum_{m=1}^M \sum_{n=2}^N c4_{icmn} \cdot x4_{igqcmn} \\
 & - \sum_{i=1}^I \sum_{g=1}^G \sum_{q=1}^Q \sum_{m=1}^M \sum_{e=1}^E \sum_{n=2}^N c5_{men} \cdot x5_{igqmen} - \sum_{i=1}^I \sum_{g=1}^G \sum_{q=1}^Q \sum_{m=1}^M \sum_{o=1}^O \sum_{n=2}^N c6_{mon} \cdot x6_{igqmon} \\
 & - \sum_{i=1}^I \sum_{g=1}^G \sum_{q=1}^Q \sum_{m=1}^M \sum_{n=2}^N c7_{igqmn} \cdot x7_{igqmn} - \sum_{i=1}^I \sum_{g=1}^G \sum_{q=1}^Q \sum_{d=1}^D \sum_{n=1}^N c8_{igqdn} \cdot x8_{igqdn} \\
 & - \sum_{i=1}^I \sum_{g=1}^G \sum_{q=1}^Q \sum_{f=1}^F \sum_{d=1}^D \sum_{n=1}^N x2_{igqfdn} \cdot c_n (y_{igqfn} - sh_i) \\
 & - \sum_{i=1}^I \sum_{g=1}^G \sum_{q=1}^Q \sum_{f=1}^F \sum_{d=1}^D \sum_{n=1}^N x2_{igqfdn} \cdot k_n (z_{igqfn} - pl_i)
 \end{aligned} \tag{1}$$

$$\begin{aligned}
 \max Z_2 = & w' \cdot \left( \sum_{i=1}^I \sum_{g=1}^G \sum_{q=1}^Q \sum_{f=1}^F \sum_{n=1}^N (R_{igqfn} \cdot u_i + R'_{igqfn} \cdot (1 - u_i)) \cdot x1_{igqfn} \right) \\
 & + (1 - w') \cdot \left( \sum_{i=1}^I \sum_{g=1}^G \sum_{q=1}^Q \sum_{m=1}^M \sum_{n=1}^N (h_{igqmn} \cdot x7_{igqmn}) \right)
 \end{aligned} \tag{2}$$

$$x1_{igqfn} = \sum_{d=1}^D x2_{igqfdn} \quad \forall i, g, q, f, n \tag{3}$$

$$x8_{igqd1} = \sum_{f=1}^F x2_{igqfd1} - \sum_{c=1}^C x3_{igqdc1} \quad \forall i, g, q, d \tag{4}$$

$$x8_{igqdn} = \sum_{f=1}^F x2_{igqfdn} - \sum_{c=1}^C x3_{igqdcn} + x8_{igqd(n-1)} \quad \forall i, g, q, d, n = \{2, \dots, N\} \tag{5}$$

$$\sum_{f=1}^F x2_{igqfdN} - \sum_{c=1}^C x3_{igqdcN} + x8_{igqd(N-1)} = 0 \quad \forall i, g, q, d \quad (6)$$

$$\sum_{f=1}^F x2_{igqfd1} \geq \sum_{c=1}^C x3_{igqdc1} \quad \forall i, g, q, d \quad (7)$$

$$\sum_{f=1}^F x2_{igqfdn} + x8_{igqd(n-1)} \geq \sum_{c=1}^C x3_{igqdcn} \quad \forall i, g, q, d, n = \{2, \dots, N-1\} \quad (8)$$

$$\sum_{d=1}^D x3_{igqdcn} = D_{igqcn} \quad \forall i, g, q, c, n \quad (9)$$

$$\sum_{m=1}^M x4_{igqcmn} = \sum_{n'=1}^{n-1} \sum_{d=1}^D r_{igqcn'n} \cdot x3_{igqdcn'} \quad \forall i, g, q, c, n = \{2, \dots, N\} \quad (10)$$

$$\sum_{c=1}^C x4_{igqcmn} = x7_{igqmn} \quad \forall i, g, q, m, n = \{2, \dots, N\} \quad (11)$$

$$w_{igq} \cdot x7_{igqmn} = \sum_{e=1}^E x5_{igqmen} + \sum_{o=1}^O x6_{igqmon} \quad \forall i, g, q, m, n = \{2, \dots, N\} \quad (12)$$

$$\gamma_{1igqn} \cdot w_{igq} \cdot x7_{igqmn} = \sum_{e=1}^E x5_{igqmen} \quad \forall i, g, q, m, n = \{2, \dots, N\} \quad (13)$$

$$\sum_{i=1}^I \sum_{g=1}^G \sum_{q=1}^Q \sum_{m=1}^M x5_{igqmen} \leq v4_{en} \quad \forall e, n = \{2, \dots, N\} \quad (14)$$

$$\sum_{i=1}^I \sum_{g=1}^G \sum_{q=1}^Q \sum_{m=1}^M x6_{igqmon} \leq v5_{on} \quad \forall o, n = \{2, \dots, N\} \quad (15)$$

$$x1_{igqfn} \leq V1_{igqfn} \quad \forall i, g, q, f, n \quad (16)$$

$$\sum_{f=1}^F x2_{igqfdn} + x8_{igqd(n-1)} \leq V2_{igqdn} \quad \forall i, g, q, d, n = \{2, \dots, N\} \quad (17)$$

$$\sum_{f=1}^F x2_{igqfd1} \leq v2_{igqd1} \quad \forall i, g, q, d \quad (18)$$

$$x7_{igqmn} \leq V3_{igqmn} \quad \forall i, g, q, m, n = \{2, \dots, N\} \quad (19)$$

$$y_{igqf1} \geq sh_i \quad \forall i, g, q, f \quad (20)$$

$$y_{igqfn} \leq y_{igqf(n-1)} \quad \forall i, g, q, f, n = \{2, \dots, N\} \quad (21)$$

$$y_{igqfN} \geq \rho_i y_{igqf1} \quad \forall i, g, q, f \quad (22)$$

$$z_{igqf1} \geq pl_i \quad \forall i, g, q, f \quad (23)$$

$$z_{igqfn} \leq z_{igqf(n-1)} \quad \forall i, g, q, f, n = \{2, \dots, N\} \quad (24)$$

$$z_{igqfN} \geq \mu_i z_{igqf1} \quad \forall i, g, q, f \quad (25)$$

$$\begin{aligned} x1_{igqfn}, x2_{igqfdn}, x3_{igqdcn}, x4_{igqcmn}, x5_{igqmen}, x6_{igqmon}, x7_{igqmn}, x8_{igqdn}, y_{igqfn}, z_{igqfn} \geq 0 \\ \forall i, g, q, f, d, c, m, e, o, n \end{aligned} \quad (26)$$

The first objective function (1) is to maximize total profit of the network (total income minus total cost). Total income is obtained by selling the final products, recyclable components, and cost discount due to the proper paper and plastic consumption in the packaging of products. Also, the costs of the considered network design include production cost of products, transportation cost between facilities, holding cost, disassembly cost of scrapped products, pecuniary penalty due to the additional paper and plastic consumption in the packaging of products, and disposal cost put in transportation cost of disposable components to disposal centers. The second objective function (2) is to maximize the sum of the score of products produced with green design indicators and quality indicators in the forward network and the green scrap score of scrapped products disassembled in the reverse network. Constraint (3) indicates that all products produced in each production center (factory) must be transferred to all distribution centers (warehouses). Constraints (4) and (5) are the inventory of warehouses from the initial periods to the end of planning horizon. Constraint (6) indicates that the amount of inventory at the end of planning horizon must be zero. Constraint (7) states that the amount of input into distribution centers in the first period must be more than the amount of output from them. Constraint (8) states that, from period 2 to period  $N - 1$ , the sum of the amount of input into distribution centers and the inventory transferred from the previous period must be more than the amount of output from them. According to the assumption that all customers' demands are satisfied, constraint (9) ensures that all products transferred from distribution centers to each customer zone are equal to the demand of that customer zone. Constraint (10) indicates that the quantity of scrapped products returned from each customer zone to disassembly centers in each period is the percentage of all products sent to that customer zone from the first period to the previous period of the considered period. Constraint (11) implies that all scrapped products transferred to the considered disassembly center must be equal to the disassembled scrapped products in that center. Constraint (12) states that all components obtained by disassembling the products in the considered disassembly center must be transferred to recycling and disposal centers. Constraint (13) states

that the percentage of all components obtained by disassembling the scrapped products in the considered disassembly center is transferred to recycling centers as recyclable components. Constraints (14)–(19) demonstrate capacity limitation in the considered network facilities. Constraint (20) indicates the amount of paper consumption in the packaging of each unit of product in each production center in the first period of the planning horizon. Constraint (21) indicates the amount of paper consumption in the packaging of each unit of product in each production center from period 2 to the end of planning horizon. This amount must be descending. Constraint (22) indicates minimum amount of paper consumption in the packaging of each unit of product in each production center at the end of planning horizon. Constraint (23) indicates the amount of plastic consumption in the packaging of each unit of product in each production center in the first period of the planning horizon. Constraint (24) indicates the amount of plastic consumption in the packaging of each unit of product in each production center from period 2 to the end of planning horizon. This amount must be descending. Constraint (25) indicates minimum amount of plastic consumption in the packaging of each unit of product in each production center at the end of planning horizon. Constraint (26) enforces the non-negativity restriction on the decision variables.

## 5 Proposed solution methodology

Because of the complexity of supply chain models from computational point of view especially in large sizes, these problems are considered as Non-deterministic Polynomial-hard problems (Dimopoulos and Zalzalá 2000). This is a constrained nonlinear-programming model which is hard to be solved analytically by an exact method. Moreover, the number of production and distribution centers, customer zones and product types affect the number of constraints. To check its hardness, all of the numerical examples are coded in Lingo software and after 7800 s none of them were computed. So, in comparison with other methods, meta-heuristic methods which can find close-to-optimum solutions in more proper time are usually used to solve these problems. Therefore, in order to solve the considered problem in this research, Multi-objective Particle Swarm Optimization (MOPSO) and Non-dominated Sorting Genetic (NSGA-II) meta-heuristic Algorithms are used to find Pareto-optimal solutions. These algorithms are well-known and their workability for different problems is proven. The reason of using two different meta-heuristic algorithms is that these two algorithms have not been used for a supply chain model identical to this research. SO, their output can be compared to verify their workability for the proposed model.

In the last two decades, various Evolutionary Algorithms (EAs) have been proposed and presented to solve multi-objective optimization problems. One of these algorithms is NSGA-II algorithm presented by Deb et al. (2002) as a population-based evolutionary algorithm to solve various problems and its efficiency in the generation of various Pareto-optimal solutions has been proved. Also, Multi-objective Particle Swarm Optimization (MOPSO) algorithm was presented by Coello Coello et al. (2004). A significant matter about Particle Swarm Optimization (PSO)

algorithm is that this algorithm is naturally a continuous optimization algorithm. But in the current research, a discrete PSO approach presented by Kazemi and Tavakkoli-Moghaddam (2011) is used for multi-objective problems. In this approach, changing the particles position is done by crossover and mutation operators. It means that for each solution (particle) as a parent, three other solutions are generated as child and the best one is selected as the next position of a particle.

### 5.1 Solution representation

The first step in applying and implementing a meta-heuristic algorithm is to select an appropriate method for solution representations. Converting a solution from the solution space into a chromosome is named encoding and converting a chromosome into a solution from the solution space is named decoding. Then, a priority-based representation method according to a coding method used by Gen et al. (2006) for two-stage Transportation Problem (tsTP) solving is presented to represent the solution chromosomes of the current green supply chain problem. In the proposed method, for a transportation network composed of  $m$  sources and  $n$  depots, a feasible chromosome is considered as a  $(m + n)$  bit permutation. The values corresponding to each node is actually a priority given to that node to contribute to the construction of a transportation network tree. In this paper, the proposed chromosome will be considered for Multi-objective Particle Swarm Optimization (MOPSO) and Non-dominated Sorting Genetic (NSGA-II) in the goal of reaching a qualified output.

Because the proposed green supply chain includes five-stage, the structure of chromosomes is also composed of five segments. According to Fig. 1, the first segment is related to the first stage and shows the transportation network between production centers and distribution centers. The second segment is related to the second stage and demonstrates the transportation network between distribution centers and customer zones. The third segment is related to the third stage and displays the transportation network between customer zones and collection/disassembly centers. The fourth segment is related to the fourth stage and shows the transportation network between collection/disassembly centers and recycling centers. Finally, the fifth segment is related to the fifth stage and demonstrates the transportation network between collection/disassembly centers and disposal centers. In Fig. 2, a random sample of a chromosome is presented for the priority-based coding method in the current research in which the green supply chain has three production centers, two distribution centers, three customer zones, two collection/disassembly centers, two recycling centers and two disposal centers.

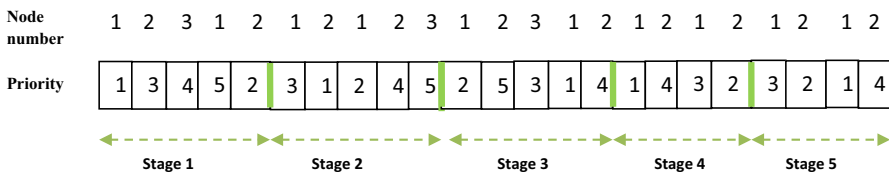


Fig. 2 A random sample of a chromosome by priority-based coding method

The priority-based decoding method for converting a chromosome into a solution from the problem solution space in the current research is as follows:

---

Inputs:  $a_i$ : capacity of sources in the forward flow (portable quantity from sources in the reverse flow)  
 $b_j$ : demand on depots in the forward flow (receivable quantity in depots in the reverse flow)  
 $c_{ij}$ : transportation cost of one unit of product or components  
 $c_j$ : holding cost in distribution centers for decoding the first segment of a chromosome or negative selling price during decoding the second segment of a chromosome or disassembly cost in disassembly centers for decoding the third segment of a chromosome or negative selling price in recycling centers for decoding the fourth segment or zero for decoding the fifth segment of a chromosome  
 $c_i$ : production cost in production centers during decoding the first segment of a chromosome or holding cost in distribution centers during decoding the second segment of a chromosome or negative selling price in customer zones for decoding the third segment of a chromosome or disassembly cost in disassembly centers for decoding the fourth and fifth segments of a chromosome  
 $v$ : chromosome

---

Output: transportation network graph ( $x_{ij}$ )

Step 1: put  $x_{ij} = 0$ ;  $\forall i \in m, \forall j \in n$

Step 2: select a node with maximum priority;  $l = \operatorname{argmax}\{v(k), k \in m + n\}$

Step 3: if  $l \in m$ , put  $j^* = \operatorname{argmin}\{c_{i^*j} + c_j, v(j + m) \neq 0, j \in n\}$  and  $i^* \leftarrow l$   
 otherwise, put  $i^* = \operatorname{argmin}\{c_{ij^*} + c_i, v(i) \neq 0, i \in m\}$  and  $j^* \leftarrow l - m$

Step 4: put  $x_{i^*j^*} = \min\{a_{i^*}, b_{j^*}\}$ ;  $b_{j^*} = b_{j^*} - x_{i^*j^*}$  and  $a_{i^*} = a_{i^*} - x_{i^*j^*}$

Step 5: if  $a_{i^*} = 0, v(i^*) = 0$ ; and  
 if  $b_{j^*} = 0, v(j^* + m) = 0$

Step 6: during decoding the forward flow, if  $v(j + m) = 0, \forall j \in n$ , stop; otherwise, return to step 2.  
 (during decoding the reverse flow, if  $v(i) = 0, \forall i \in m$ , stop; otherwise, return to step 2).

---

It should be noted that in the chromosome of the current model, first, the second segment is decoded. Then, the first, third, fourth and fifth segments are decoded respectively. Also, the values of variables  $y_{igqfn}$  and  $z_{igqfn}$  are calculated by relations (20)–(25).

## 5.2 Crossover

Crossover operator applied in this research is the type of two cut points. In this crossover method, in each segment of two selected parent chromosomes, two cut points are selected randomly; so, each segment is divided into three parts.

Therefore, children chromosomes are created by applying crossover operator on all segments of two parent chromosomes. According to Figs. 3, 4 and 5, for instance, we apply the two-point crossover operator only on the first segment of two parent chromosomes and show how to create the first segment of the first and second child. Similarly, these procedures are performed on the rest of the segments of parents' chromosomes and finally the rest of the segments of children chromosomes are completed.

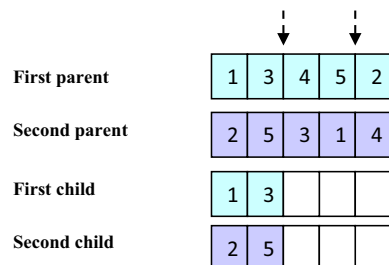
Consider the first segment of parents' chromosome:

- Step 1** We randomly select two cut points for parents' chromosome. Therefore, each parent is divided into three segments: left side, middle side, right side. Then, gens of the left side of the first parent are transferred to the left side of the first child. Similarly, the second child is also generated by the second parent.
- Step 2** For the middle side of the first child, gens of the second parent which are similar to gens of the left side of the first child are omitted. Then, among the rest of gens of the second parent, we start from the left and select gens based on the number of gens needed for the middle side of the first child. So, these selected gens are transferred to there. Similarly, the second child is also generated by the first parent.
- Step 3** Gens of the first parent which are similar to gens of the first child are omitted. Then, among the rest of genes of the first parent, we start from the left and select gens based on the number of gens needed for the right side of the first child. So, these selected gens are transferred to there. Similarly, the second child is also generated by the second parent.

### 5.3 Mutation

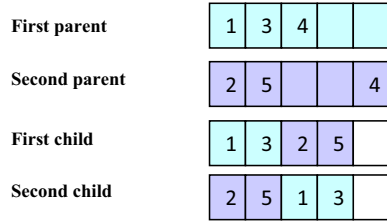
As shown in Fig. 6, in this research, mutation operator is the great displacement type. The performance of this operator is that two genes along each segment of the considered chromosome are randomly selected and the values of them are replaced with each other. According to the five-segment structure of the chromosome in this problem, mutation operator is applied on all segments of the parent chromosome. How to apply mutation operator on the first segment of the parent chromosome is shown in Fig. 6. This procedure is also done to the rest of the segments.

**Fig. 3** Step 1 representation for applying crossover operator

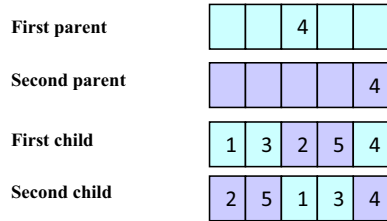




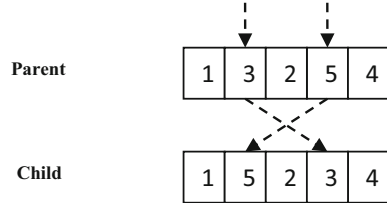
**Fig. 4** Step 2 representation for applying crossover operator



**Fig. 5** Step 3 representation for applying crossover operator



**Fig. 6** Great displacement mutation



**5.4 Steps of the proposed NSGA-II algorithm**

After determining a structure to represent the solution chromosomes of the problem, the first step of the algorithm is to generate initial population of solutions. Most of the population-based evolutionary meta-heuristic algorithms use a random approach to generate initial solutions. This approach is applied here. In the next step of the algorithm, generated population must be evaluated. In this step, first, generated chromosomes are converted into the equivalent solutions by described decoding methods. Then, each objective function is separately calculated to solutions. Finally, solutions ranking and putting them in various fronts based on the non-domination degree are done by the fast non-dominated sorting method. Also, crowding distance is calculated to solutions.

Among the solutions of each generation in NSGA-II algorithm, some of them are selected by the binary tournament selection method. In this method, two solutions are randomly selected from the population. Then, these two solutions are compared and the best one is finally selected. Selection criteria in NSGA-II algorithm are the rank of a solution and crowding distance related to the solution respectively. If the rank of a solution is less and its crowding distance is more, the solution is more optimum.

By the repetition of the binary selection operator on the population of each generation, a set of the members of that generation is selected to participate in crossover and mutation. Crossover procedure is executed on part of the set of selected members and mutation procedure is executed on the rest of the set; thus, a

population of children is created through these two procedures. Then, this population is integrated with the main population. At first, the members of the new generated population are sorted in ascending order by the rank. But, the members of the population which have the same rank are sorted in descending order by crowding distance. The members from the head of the sorted list being equal to the number of the members of the main population are selected and the rest of the members of the population are omitted. The selected members make the population of the next generation. The mentioned cycle in this section is repeated until the termination condition is achieved and finally the members of the first front of the last generation are selected as the set of Pareto-optimal solutions.

### 5.5 Steps of the proposed MOPSO algorithm

Before pointing to the steps of MOPSO algorithm, it is necessary to introduce two notations used in it:

- $Pbest_i$  The best positions in which particle  $i$  has been placed so far (Pareto-optimal archive of the best memory of particle  $i$ ).
- $Gbest$  The best positions achieved by all particles' experience (Pareto-optimal archive of the best overall memory of that generation).

All steps of MOPSO algorithm used in this research are explained here:

- Step 1 According to the number of initial population, solutions are randomly generated which show the position of each particle.
- Step 2 Each of the chromosomes of initial population is evaluated after decoding. Then,  $Pbest_i$  as the best memory of particle  $i$  is put as the equivalent of current position of particle  $i$ .
- Step 3 The members of initial population are fronted based on their ranks and the members of the first front are saved as Pareto-optimal archive of the best overall memory of this generation.
- Step 4 Chance of the selection of each member of the first front as a selected member for applying crossover operator to generate the second child in the next step is calculated (Noori-Darvish et al. 2012).
- Step 5 For each members of initial population, three new solutions are generated as child and the best one is selected as the next position of a particle. The first child is created by crossing a particle with one of the members of its best memory archive ( $Pbest_i$ ). Because the considered problem is bi-objective, after comparing the new position of a particle with its best memory, none of these two positions may be superior to the other one in the next generations. Therefore, both of them are saved in that particle's best memory archive and then, based on which members of this archive have more crowding distance, selected for crossing with the considered particle. The second child is created by crossing with one of the members of the current generation's best overall memory archive ( $Gbest$ ) and the third child is created by applying mutation on that particle (Kazemi and Tavakkoli-Moghaddam 2011).

- Step 6 Each particle's best memory archive is updated by comparing that particle's best memory archive with the new position of that particle.
- Step 7 After generating the new position for each members of initial population, the current generation's best overall memory and the new position of all particles are fronted and the members of the first front are transferred to the next generation as updated current generation's best overall memory Pareto-optimal archive.
- Step 8 If the termination condition of the algorithm is achieved, the last earned generation's best overall memory is extracted as Pareto solutions obtained by this algorithm, otherwise return to step 4.

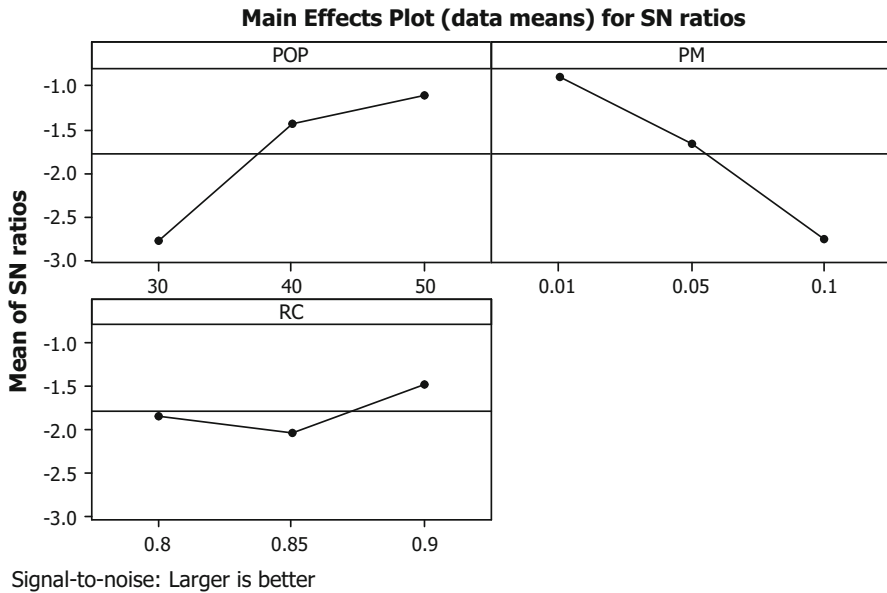
It should be noted that in the proposed algorithm in this research, no capacity restriction is enforced on the generations' best overall memory Pareto-optimal archive. Hence, all best found solutions are kept and their chance of presence is not missed. Also, time is considered as the termination condition for both algorithms in order that both of them stop after passing the determined time.

### 5.6 Parameters setting of algorithms: Taguchi experimental design

Quality of a meta-heuristic algorithm extremely depends on the proper determination of its parameters. There are different methods for the determination of parameters in meta-heuristic algorithms. In this research, Taguchi experimental design has been used for parameters setting of developed algorithms (Taguchi 1986). After determining the number of experiments and how to combine parameters in each algorithm, by the selected array for each algorithm, algorithms are executed based on Taguchi design in the next stage. To compare and analyze the obtained results, it is necessary to introduce an indicator which can compare algorithms for various combinations of parameters. In this research, after providing the values of quality percent obtained by comparing the set of Pareto solutions of two algorithms in each combination of parameter, these values are given to MINITAB.14 software as inputs and analyzed by S/N indicator (ratio of signal to noise; signal indicates a desirable value and noise indicates a non-desirable value). The values of S/N ratios obtained for each level of parameter in each algorithm are depicted in Figs. 7 and 8. If the value of S/N is larger for a level of parameter, the algorithm has better performance in that level. According to the obtained results, tuned values for all parameters of algorithms are shown in Tables 1 and 2.

## 6 Computational results

To evaluate and compare the performance of the developed algorithms in this research, a scheme is used to generate the data for sample problems. As shown in Table 3, ten different sizes based on the number of periods, products types, production centers, distribution centers, customer zones, collection/disassembly centers, recycling and disposal centers are regarded for the problems; so, ten sample problems have been designed for testing. It should be noted that three predefined

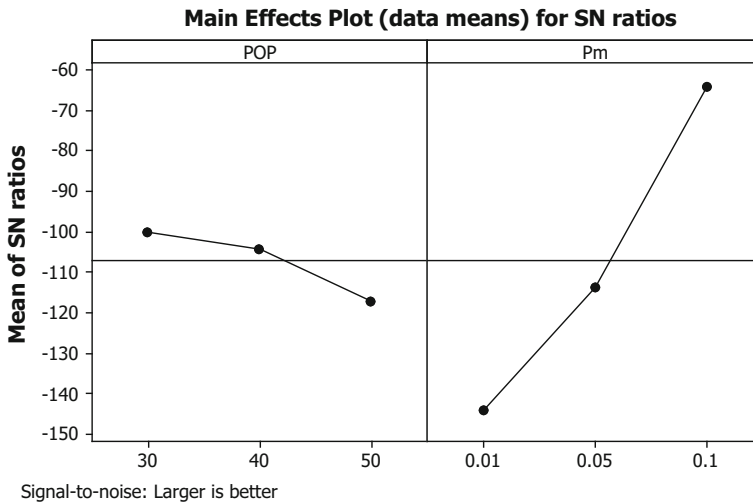


**Fig. 7** S/N ratios for different levels of parameters of NSGA-II

green design indicators and two predefined quality indicators are the same for all problems. Finally, test sample problems are solved by two proposed algorithms coded in MATLAB 7.0 on a Laptop Computer at Core i3-2.53 GHz with 4 GB of RAM and the performance of both algorithms are compared by special indicators of multi-objective problems.

The values of parameters used in the sample problems, except some parameters referred to specific reference, are generated randomly using uniform distributions specified in Tables 4, 5, 6, 7, 8, 9 and 10.

- Reasonable procedure for generating data for parameters related to the score of product (referred to the score obtained by green design indicator ( $R_{igfn}$ ) or quality indicator ( $R'_{iqfn}$ ) for final products or the green scrap score ( $h_{igqn}$ ) for scrapped products) is that it should be in  $[0, 1]$  and increases from a period to the next period. Because the considered problem is multi-period, the range  $([0, 1])$  is divided by the number of periods. Then, we start from the initial range for the first period and go on to reach the last range for the end of planning horizon. For example, consider the first problem with 9 periods; as mentioned above,  $1/9 = 0.11$ . Therefore, the parameters related to the score of product in the first period are selected randomly using uniform distribution from  $(0.005, 0.11)$ ; then, the defined range is increased by the constant value 0.11 from each period to the next period. So, giving the values of the parameters in the first problem is according to the ranges specified in Table 6. It should be noted that the beginning of the range for the first period is not considered zero. Also, the values of these parameters for the rest of the problems are generated by the same



**Fig. 8** S/N ratios for different levels of parameters of MOPSO

**Table 1** Tuned parameters for NSGA-II algorithm

	Pop-size	Mutation probability (pm)	Crossover rate (RC)
NSGA-II	50	0.01	0.9

**Table 2** Tuned parameters for MOPSO algorithm

	Pop-size	Mutation probability (pm)
MOPSO	30	0.1

procedure. Furthermore, the parameter related to the green scrap score is valued from period 2.

- The value of parameter  $w_{igq}$  for each type of product is selected randomly using uniform distribution from the ranges specified in Table 7.
- The value of parameter  $\gamma_{1igqn}$  must be increased from each period to the next period only based on the second green design indicator. Therefore, the value of this parameter is generated only based on this indicator and according to the calculation process of the score of product; the only difference is that this parameter isn't valued in the first period. For example, consider the second problem with 10 periods. As mentioned above,  $1/10 = 0.1$ . So, the value of this parameter in the second period is selected randomly using uniform distribution from (0.1, 0.2) and this defined range is increased by the constant value 0.1 from each period to the next period. Therefore, the values of this parameter in the

**Table 3** Structure of sample problems

Problem number	Problem size ( $F \times D \times C \times M \times O \times E$ )	Number of products Types (I)	Number of periods (N)
1	$(4 \times 3 \times 5 \times 3 \times 1 \times 2)$	3	9
2	$(6 \times 5 \times 7 \times 5 \times 3 \times 4)$	3	10
3	$(9 \times 8 \times 10 \times 8 \times 6 \times 7)$	4	11
4	$(11 \times 10 \times 12 \times 10 \times 8 \times 9)$	4	12
5	$(13 \times 12 \times 14 \times 12 \times 10 \times 11)$	5	13
6	$(16 \times 15 \times 17 \times 15 \times 13 \times 14)$	5	14
7	$(18 \times 17 \times 19 \times 17 \times 15 \times 16)$	5	15
8	$(21 \times 20 \times 22 \times 20 \times 18 \times 19)$	6	16
9	$(21 \times 20 \times 22 \times 22 \times 19 \times 19)$	6	17
10	$(21 \times 20 \times 22 \times 23 \times 21 \times 19)$	7	18

**Table 4** Range of the values of some parameters used for sample problems

Parameter	Range
$D_{igqcn}$	Uniform(80,250) (Pishvae et al. 2010)
$V1_{igqfn}$	Uniform(600,1500) (Pishvae et al. 2010)
$V2_{igqdn}$	Uniform(1,000,000,2,000,000)
$V3_{igqmn}$	Uniform(1,000,000,2,000,000)
$v4_{en}$	Uniform(5,000,000,7,000,000)
$v5_{on}$	Uniform(7,000,000,9,000,000)
$sh_i$	Uniform(60,80)
$pl_i$	Uniform(50,70)
$\rho_i$	Uniform(0.8,0.9)
$\mu_i$	Uniform(0.7,0.9)
$u_i$	= 0.5
$w'$	= 0.5 (Pishvae et al. 2010)

second problem for different periods are based on data specified in Table 8. Also, the values related to this parameter based on the second green design indicator for the rest of the problems are generated by the same procedure. Furthermore, the value of this parameter for the rest of the indicators and all problems is selected randomly using uniform distribution from (0.05, 0.6) from the second period.

- Parameter  $r_{igqcn'n}$  for production period  $n' = 1$  [the return of the products starts from period 2 ( $n = 2$ )] for the second problem is obtained randomly using uniform distribution from the ranges specified in Table 9. It shows that at the beginning of the planning horizon, the return rate in the primary periods is more than the end periods. But for production period  $n' = 2$  [the return of the products starts from period 3 ( $n = 3$ )], it is completely different, which means that the

**Table 5** Range of the values of some parameters used for sample problems

Parameter	Range of data in the first period	Range of data in the second period	Explanation
$c1_{igqfn}$	Uniform(20,30)	Uniform(30,40)	Range of the value of this parameter is increased by the constant value 10 (the distance between the end and beginning of the range) from each period to the next period
$c2_{jdm}, c3_{idcn}$	Uniform(1,3)	Uniform(3,5)	Range of the value of this parameter is increased by the constant value 2 from each period to the next period
$c4_{icmn}, c5_{men}, c6_{mon}$	–	Uniform(3,5)	Range of the value of this parameter is increased by the constant value 2 from each period to the next period
$c7_{igqmn}$	–	Uniform(6,9)	Range of the value of this parameter is increased by the constant value 3 from each period to the next period
$c8_{igqdn}$	Uniform(8,12)	Uniform(12,16)	Range of the value of this parameter is increased by the constant value 4 from each period to the next period
$P_{igqcn}$	Uniform(2000,3000)	Uniform(3000,4000)	Range of the value of this parameter is increased by the constant value 1000 from each period to the next period
$c_n, k_n$	= 10	= 11	The value of this parameter is increased by the constant value 1 from each period to the next period
$p'_{en}$	–	Uniform(100,200)	Range of the value of this parameter is increased by the constant value 100 from each period to the next period

**Table 6** Range of the values of parameters related to the score of product in the first problem

Period 1	Period 2	Period 3	Period 4	Period 5	Period 6	Period 7	Period 8	Period 9
Uniform (0.005 0.11)	Uniform (0.11 0.22)	Uniform (0.22 0.33)	Uniform (0.33 0.44)	Uniform (0.44 0.55)	Uniform (0.55 0.66)	Uniform (0.66 0.77)	Uniform (0.77 0.88)	Uniform (0.88 0.99)

**Table 7** Range of the values of parameter  $w_{igq}$ 

Product type 1	Product type 2	Product type 3	Product type 4	Product type 5	Product type 6	Product type 7
Uniform (8 13)	Uniform (9 14)	Uniform (10 15)	Uniform (11 16)	Uniform (12 17)	Uniform (8 13)	Uniform (9 14)

return rate in the primary periods is less than the end periods which shows the increase of the quality of products. Therefore, the value of the considered parameter for production period  $n' = 2$  is obtained randomly using uniform distribution from the ranges specified in Table 10. The same procedure is used for the rest of the production periods. For example, if  $n' = 3$ , the return rate in the fourth period ( $n = 4$ ) is obtained randomly using uniform distribution from (0.02, 0.06). We also use the defined ranges in Table 10 for the rest of the periods. Also, this procedure and these ranges are used for the rest of the problems.

## 6.1 Comparison indicators

The appropriate definition of criteria is very important to assess an algorithm. So, in this research, three quality indicators, namely Pareto front solutions, indicator of the distance (dispersion) between Pareto front solutions, and indicator of variety of Pareto front solutions have been used to compare the performance of two algorithms.

### 6.1.1 Quality indicator

Quality indicator has been proposed by Schaffer (1985). This indicator compares the quality of Pareto solutions obtained by each algorithm. Actually, this indicator fronts all Pareto solutions obtained by two algorithms and characterizes what percentage of the solutions of echelon (front) 1 belongs to each algorithm. The higher the percentage, the better the quality of the algorithm. The mean quality indicator achieved by three runs of each problem for two considered algorithms and total mean achieved by this indicator for both algorithms are reported in Table 11.

### 6.1.2 Distance indicator

Distance indicator has been used by Srinivas and Deb (1994) and Coello Coello et al. (2007). This indicator helps to measure the uniformity in the distribution of Pareto points in the set of obtained Pareto solutions. Distance indicator is calculated as follows (Noori-Darvish et al. 2012):



**Table 8** Range of the values of parameter  $\gamma_{i,igqm}^1$  for the second problem

Period 1	Period 2	Period 3	Period 4	Period 5	Period 6	Period 7	Period 8	Period 9	Period 10
-	Uniform (0.1 0.2)	Uniform (0.2 0.3)	Uniform (0.3 0.4)	Uniform (0.4 0.5)	Uniform (0.5 0.6)	Uniform (0.6 0.7)	Uniform (0.7 0.8)	Uniform (0.8 0.9)	Uniform (0.9 1)

**Table 9** Range of the values of parameter  $r_{igqcn'n}$  for production period  $n' = 1$  for the second problem

Period 1	Period 2	Period 3	Period 4	Period 5	Period 6	Period 7	Period 8	Period 9	Period 10
–	Uniform (0.2 0.3)	Uniform (0.1 0.2)	Uniform (0.05 0.1)	Uniform (0.05 0.09)	Uniform (0.03 0.08)	uniform (0.03 0.07)	Uniform (0.02 0.06)	0	0

**Table 10** Range of the values of parameter  $r_{igqcn'n}$  for production period  $n' = 2$  for the second problem

Period 1	Period 2	Period 3	Period 4	Period 5	Period 6	Period 7	Period 8	Period 9	Period 10
–	–	Uniform (0.02 0.06)	Uniform (0.03 0.07)	Uniform (0.03 0.08)	Uniform (0.05 0.09)	Uniform (0.05 0.1)	Uniform (0.1 0.2)	Uniform (0.2 0.3)	0

$$SM = \frac{\sum_{i=1}^{N-1} |\bar{d} - d_i|}{(N - 1) \times \bar{d}} \tag{27}$$

where  $d_i$ , Euclidean distance between the consecutive solutions in the set of Pareto solutions;  $\bar{d}$ , Mean distances; N, the number of solutions in the set of Pareto.

In comparing two algorithms, the smaller the distance indicator of an algorithm, the better that algorithm. The mean distance indicator achieved by three runs of each problem for two considered algorithms and total mean achieved by this indicator for both algorithms are reported in Table 12.

**Table 11** Comparison of quality indicator for proposed algorithms (based on percentage)

Problems	NSGA-II	MOPSO
1	70.9091	29.0909
2	68.8492	31.1508
3	47.7778	52.2222
4	65.0794	34.9206
5	60.1515	39.8485
6	58.8889	41.1111
7	72.6984	27.3016
8	58.1145	41.8855
9	55	45
10	51.8519	48.1481
Total mean	60.93207	39.06793

### 6.1.3 Variety indicator

Variety indicator has been used by Zitzler (1999). This indicator measures the distribution of the solutions in the set of obtained Pareto solutions. Its definition is as follows (Noori-Darvish et al. 2012):

$$D = \sqrt{\sum_{i=1}^N \max(\|x_i - y_j\|)} \quad (28)$$

where  $\|x_i - y_j\|$ , Euclidean distance between two Pareto solutions  $x_i$  and  $y_j$ ;  $N$ , the number of objective functions.

In comparing two algorithms, the more variety of solutions in an algorithm, the better that algorithm. The mean variety indicator achieved by three runs of each problem for two considered algorithms and total mean achieved by this indicator for both algorithms are reported in Table 13.

According to the last row of Tables 11, 12 and 13 showing total mean achieved by quality, distance, and variety indicators of Pareto solutions for each algorithm, we conclude that NSGA-II algorithm has the best performance in quality and variety indicators and MOPSO algorithm has the better performance in distance indicator. Therefore, NSGA-II algorithm can generate more quality and various solutions in the considered problem. Users can choose one of them according to the indicator that is more proper for them. Moreover, users can choose one solution from the best rank that is gained via normalizing by  $\sqrt{(x_i - X)^2 + (y_i - Y)^2}$  in which  $X$  and  $Y$  are the best amounts that are gained for the first and second objective function in the best rank and  $(x_i, y_i)$  are different solutions of the proposed rank.

**Table 12** Comparison of distance indicator for proposed algorithms

Problems	NSGA-II	MOPSO
1	0.7833	0.7612
2	0.9775	0.7026
3	1.1032	0.8037
4	0.9525	1.1464
5	1.2195	1.2566
6	0.7343	0.6559
7	1.3528	0.9787
8	1.1154	0.9920
9	0.6050	0.8344
10	1.014	0.4097
Total mean	0.98575	0.85412

**Table 13** Comparison of variety indicator for proposed algorithms

Problems	NSGA-II	MOPSO
1	1764466.667	1565700
2	6528570	1581433.333
3	25242033.33	6973866.667
4	39973800	37016333.33
5	85171666.67	52428066.67
6	62920216.67	63391233.33
7	122117333.3	149640000
8	255716666.7	197386666.7
9	246166666.7	246926666.7
10	367806666.7	172400000
Total mean	121340808.7	92930996.67

## 7 Conclusion

Due to increasing importance of environmental issues in the supply chain network, this research presents a bi-objective nonlinear programming model in order to determine the flow of a green supply chain network with the aim of increasing total profit of the network and maximizing the sum of the score of products produced by green design indicators and quality indicators in the forward network and the green scrap score of scrapped products disassembled in the reverse network. In order to develop the products in the current model, two indicators about green design of a product and its quality for final products have been considered.

To solve the proposed model, Multi-objective Particle Swarm Optimization (MOPSO) and Non-dominated Sorting Genetic (NSGA-II) meta-heuristic Algorithms have been used with a priority-based coding method to represent the solution chromosomes. Because the appropriate selection of the values of parameters of algorithms is very effective to converge them on a good set of Pareto-optimal solutions, all parameters have been tuned by Taguchi Experimental Design method. Then, the performance of two algorithms has been investigated and compared for 10 sample problems by some special indicators of multi-objective problems. Computational results and solutions analysis based on the defined indicators showed that NSGA-II algorithm had the best performance in quality and variety indicators and MOPSO algorithm had the better performance in distance indicator. Therefore, NSGA-II algorithm can generate more quality and various solutions in the considered problem. It should be noted that in order to compare in the same condition, the comparison between algorithms was performed under the condition that the execution time of algorithms as the termination condition was considered the same for both algorithms.

To develop the current model and present new researches, we can add some strategic decisions such as the selection of production centers, distribution centers, and collection/disassembly centers to this problem. Furthermore, the parameters of the considered problem are deterministic. But in the real world, most of these parameters like demand are non-deterministic. So, a new problem can be presented

by considering non deterministic parameters for this model. Also, in relation to the structure of the proposed algorithms, we can use the common method, that is, the introduction of two separate algorithms and a new hybrid algorithm as a combination of them.

## References

- Abu Seman NA, Zakuan N, Jusoh A, Shoki M (2012) Green supply chain management: a review and research direction. *Int J Manag Value Supply Chains* 3:1–18. <https://doi.org/10.5121/ijmvsc.2012.3101>
- Alfonso-Lizarazo EH, Montoya-Torres JR, Gutiérrez-Franco E (2013) Modeling reverse logistics process in the agro-industrial sector: the case of the palm oil supply chain. *Appl Math Model* 37:9652–9664. <https://doi.org/10.1016/j.apm.2013.05.015>
- Baumgarten H, Butz C, Fritsch A, Sommer-Dittrich T (2003) Supply chain management and reverse logistics-integration of reverse logistics processes into supply chain management approaches. In: 2003 IEEE international symposium on electronics and the environment, pp 79–83
- Boks C, Stevels A (2007) Essential perspectives for design for environment. Experiences from the electronics industry. *Int J Prod Res* 45:4021–4039. <https://doi.org/10.1080/00207540701439909>
- Coello Coello CA, Toscano Pulido G, Salazar Lechuga M (2004) Handling multiple objectives with particle swarm. *Optimization*. <https://doi.org/10.1109/tevc.2004.826067>
- Coello Coello CA, Lamont GB, Van Veldhuizen DA (2007) Evolutionary algorithms for solving multi-objective problems, 2nd edn. Springer, US. <https://doi.org/10.1007/978-0-387-36797-2>
- Deb K, Pratap A, Agarwal S (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput* 6:182–197
- Dimopoulos C, Zalzalá AMS (2000) Recent developments in evolutionary computation for manufacturing optimization: problems, solutions, and comparisons. *IEEE Trans Evol Comput* 4:93–113
- Dondo RG, Méndez CA (2016) Operational planning of forward and reverse logistic activities on multi-echelon supply-chain networks. *Comput Chem Eng* 88:170–184. <https://doi.org/10.1016/j.compchemeng.2016.02.017>
- El-Sayed M, Afia N, El-Kharbotly A (2010) A stochastic model for forward–reverse logistics network design under risk. *Comput Ind Eng* 58:423–431. <https://doi.org/10.1016/j.cie.2008.09.040>
- Entezaminia A, Heydari M, Rahmani D (2016) A multi-objective model for multi-product multi-site aggregate production planning in a green supply chain: considering collection and recycling centers. *J Manuf Syst* 40:63–75. <https://doi.org/10.1016/j.jmsy.2016.06.004>
- Fleischmann M, Bloemhof-Ruwaard JM, Dekker R, van der Laan E, van Nunen JAEE, Van Wassenhove LN (1997) Quantitative models for reverse logistics: a review. *Eur J Oper Res* 103:1–17. [https://doi.org/10.1016/S0377-2217\(97\)00230-0](https://doi.org/10.1016/S0377-2217(97)00230-0)
- Gaur J, Amini M, Rao AK (2017) Closed-loop supply chain configuration for new and reconditioned products: an integrated optimization model. *Omega* 66:212–223. <https://doi.org/10.1016/j.omega.2015.11.008>
- Gen M, Altıparmak F, Lin L (2006) A genetic algorithm for two-stage transportation problem using priority-based encoding. *OR Spectr* 28:337–354. <https://doi.org/10.1007/s00291-005-0029-9>
- Ghayebloo S, Tarokh MJ, Venkatadri U, Diallo C (2015) Developing a bi-objective model of the closed-loop supply chain network with green supplier selection and disassembly of products: the impact of parts reliability and product greenness on the recovery network. *J Manuf Syst* 36:76–86. <https://doi.org/10.1016/j.jmsy.2015.02.011>
- Giarola S, Zamboni A, Bezzo F (2011) Spatially explicit multi-objective optimisation for design and planning of hybrid first and second generation biorefineries. *Comput Chem Eng* 35:1782–1797. <https://doi.org/10.1016/j.compchemeng.2011.01.020>
- Govindan K, Soleimani H, Kannan D (2015) Reverse logistics and closed-loop supply chain: a comprehensive review to explore the future. *Eur J Oper Res* 240:603–626. <https://doi.org/10.1016/j.ejor.2014.07.012>

- Jamshidi R, Fatemi Ghomi SMT, Karimi B (2012) Multi-objective green supply chain optimization with a new hybrid memetic algorithm using the Taguchi method. *Sci Iran* 19:1876–1886. <https://doi.org/10.1016/j.scient.2012.07.002>
- Kazemi FS, Tavakkoli-Moghaddam R (2011) Solving a multi-objective multi-mode resource-constrained project scheduling problem with particle swarm optimization. *Int J Acad Res* 3:103–110
- Lu Z, Bostel N (2007) A facility location model for logistics systems including reverse flows: the case of remanufacturing activities. *Comput Oper Res* 34:299–323. <https://doi.org/10.1016/j.cor.2005.03.002>
- Mohammed F, Selim SZ, Hassan A, Syed MN (2017) Multi-period planning of closed-loop supply chain with carbon policies under uncertainty. *Transp Res Part D Transp Environ* 51:146–172. <https://doi.org/10.1016/j.trd.2016.10.033>
- Murphy PR, Poist RF (2003) Green perspectives and practices: a “comparative logistics” study. *Supply Chain Manag Int J* 8:122–131. <https://doi.org/10.1108/13598540310468724>
- Noori-Darvish S, Mahdavi I, Mahdavi-Amiri N (2012) A bi-objective possibilistic programming model for open shop scheduling problems with sequence-dependent setup times, fuzzy processing times, and fuzzy due dates. *Appl Soft Comput* 12:1399–1416. <https://doi.org/10.1016/j.asoc.2011.11.019>
- Özceylan E, Paksoy T, Bektaş T (2014) Modeling and optimizing the integrated problem of closed-loop supply chain network design and disassembly line balancing. *Transp Res Part E Logist Transp Rev* 61:142–164. <https://doi.org/10.1016/j.tre.2013.11.001>
- Pazhani S, Ramkumar N, Narendran TT, Ganesh K (2013) A bi-objective network design model for multi-period, multi-product closed-loop supply chain. *J Ind Prod Eng* 30:264–280. <https://doi.org/10.1080/21681015.2013.830648>
- Pedram A, Yusoff NB, Udoney OE, Mahat AB, Pedram P, Babalola A (2017) Integrated forward and reverse supply chain: a tire case study. *Waste Manag* 60:460–470. <https://doi.org/10.1016/j.wasman.2016.06.029>
- Pishvae MS, Razmi J (2012) Environmental supply chain network design using multi-objective fuzzy mathematical programming. *Appl Math Model* 36:3433–3446. <https://doi.org/10.1016/j.apm.2011.10.007>
- Pishvae MS, Farahani RZ, Dullaert W (2010) A memetic algorithm for bi-objective integrated forward/reverse logistics network design. *Comput Oper Res* 37:1100–1112. <https://doi.org/10.1016/j.cor.2009.09.018>
- Ramezani M, Bashiri M, Tavakkoli-Moghaddam R (2013) A new multi-objective stochastic model for a forward/reverse logistic network design with responsiveness and quality level. *Appl Math Model* 37:328–344. <https://doi.org/10.1016/j.apm.2012.02.032>
- Richey GR, Genchev SE, Daugherty PJ (2005) The role of resource commitment and innovation in reverse logistics performance. *Int J Phys Distrib Logist Manag* 35:233–257. <https://doi.org/10.1108/09600030510599913>
- Roghani E, Pazhoheshfar P (2014) An optimization model for reverse logistics network under stochastic environment by using genetic algorithm. *J Manuf Syst* 33:348–356. <https://doi.org/10.1016/j.jmsy.2014.02.007>
- Schaffer JD (1985) Multiple objective optimization with vector evaluated genetic algorithms. Paper presented at the Proceedings of the 1st international conference on genetic algorithms
- Schulmann F, Zumkeller M, Rentz O (2006) Modeling reverse logistic tasks within closed-loop supply chains: an example from the automotive industry. *Eur J Oper Res* 171:1033–1050. <https://doi.org/10.1016/j.ejor.2005.01.016>
- Soleimani H, Kannan G (2015) A hybrid particle swarm optimization and genetic algorithm for closed-loop supply chain network design in large-scale networks. *Appl Math Model* 39:3990–4012. <https://doi.org/10.1016/j.apm.2014.12.016>
- Soleimani H, Seyyed-Esfahani M, Shirazi MA (2013) Designing and planning a multi-echelon multi-period multi-product closed-loop supply chain utilizing genetic algorithm. *Int J Adv Manuf Technol* 68:917–931. <https://doi.org/10.1007/s00170-013-4953-6>
- Soleimani H, Govindan K, Saghafi H, Jafari H (2017) Fuzzy multi-objective sustainable and green closed-loop supply chain network design. *Comput Ind Eng* 109:191–203. <https://doi.org/10.1016/j.cie.2017.04.038>
- Srinivas N, Deb K (1994) Multiobjective optimization using nondominated sorting in genetic algorithms. *Evol Comput* 2:221–248
- Srivastava SK (2007) Green supply-chain management: a state-of-the-art literature review. *Int J Manag Rev* 9:53–80

- Taguchi G (1986) Introduction to quality engineering: designing quality into products and processes. Asian Productivity Organization
- Tarokh MJ, Naseri A (2012) Genetic algorithm and hybrid method to minimize total distribution cost in multi-level supply chain. *J Ind Eng* 46:15–26
- Tiwari A, Chang P-C, Tiwari MK, Kandhway R (2016) A hybrid territory defined evolutionary algorithm approach for closed loop green supply chain network design. *Comput Ind Eng* 99:432–447. <https://doi.org/10.1016/j.cie.2016.05.018>
- Üster H, Easwaran G, Akçali E, Çetinkaya S (2007) Benders decomposition with alternative multiple cuts for a multi-product closed-loop supply chain network design model. *Naval Res Logist* 54:890–907
- Wang F, Lai X, Shi N (2011) A multi-objective optimization for green supply chain network design. *Decis Support Syst* 51:262–269. <https://doi.org/10.1016/j.dss.2010.11.020>
- Yeh W-C, Chuang M-C (2011) Using multi-objective genetic algorithm for partner selection in green supply chain problems. *Expert Syst Appl* 38:4244–4253. <https://doi.org/10.1016/j.eswa.2010.09.091>
- Zitzler E (1999) Evolutionary algorithm for multi-objective optimization: methods and applications. Computer Engineering and Networks Laboratory
- Zohal M, Soleimani H (2016) Developing an ant colony approach for green closed-loop supply chain network design: a case study in gold industry. *J Clean Prod* 133:314–337. <https://doi.org/10.1016/j.jclepro.2016.05.091>