

# A meta-heuristic approach supported by NSGA-II for the design and plan of supply chain networks considering new product development

Zahra Alizadeh Afrouzy<sup>1,2</sup> · Mohammad Mahdi Paydar<sup>3</sup> · Seyed Hadi Nasseri<sup>2</sup> · Iraj Mahdavi<sup>4</sup>

Received: 7 March 2017 / Accepted: 29 May 2017 / Published online: 12 June 2017  
© The Author(s) 2017. This article is an open access publication

**Abstract** There are many reasons for the growing interest in developing new product projects for any firm. The most embossed reason is surviving in a highly competitive industry which the customer tastes are changing rapidly. A well-managed supply chain network can provide the most profit for firms due to considering new product development. Along with profit, customer satisfaction and production of new products are goals which lead to a more efficient supply chain. As new products appear in the market, the old products could become obsolete, and then phased out. The most important parameter in a supply chain which considers new and developed products is the time that developed and new products are introduced and old products are phased out. With consideration of the factors noted above, this study proposes to design a tri-objective multi-echelon multi-product multi-period supply chain model, which incorporates product development and new product production and their effects on supply chain configuration. The supply chain under consideration is assumed to consist of suppliers, manufacturers, distributors and customer groups. In terms of overcoming NP-hardness of the proposed model and in order to solve the complicated

problem, a non-dominated sorting genetic algorithm is employed. As there is no benchmark available in the literature, the non-dominated ranking genetic algorithm is developed to validate the results obtained and some test problems are provided to show the applicability of the proposed methodology and evaluate the performance of the algorithms.

**Keywords** Supply chain · New product development · NSGA-II · NREGA · Tri-objective problem

## Introduction and literature review

In most classical supply chain (SC) network designs, the major goal is to produce and send products from one echelon to another echelon to satisfy the demands of the market place in order to minimize the chains' costs or maximize the total benefit. Today's competitive market makes the supply chain management more important. Nowadays, to survive in a highly competitive market place, companies have to reduce supply chain risk, improve the distribution methods, optimize inventory levels and improve customer service and customer satisfaction. In the literature of supply chain network design problems, to make the models more realistic, several objectives are considered together (Murillo-Alvarado et al. 2015; Pasan-dideha et al. 2015; Godichaud and Amodeo 2015; Gholamian et al. 2015). Besides, the solution of multi-objective problems is both attractive and challenging (Kao et al. 2014; Jadidia et al. 2015; Ghodrattnama et al. 2015; Validi et al. 2014).

As customer interests are rapidly changing, strategies to collaborate with or compete with suitable firms within a network should be considered in the new product

✉ Zahra Alizadeh Afrouzy  
z.afrouzy@stu.umz.ac.ir

<sup>1</sup> Department of Mathematics, Parsa Institute of Higher Education, Babolsar, Iran

<sup>2</sup> Department of Mathematics, Faculty of Mathematical Sciences, University of Mazandaran, Babolsar, Iran

<sup>3</sup> Department of Industrial Engineering, Babol Noshirvani University of Technology, Babol, Iran

<sup>4</sup> Department of Industrial Engineering, Mazandaran University of Science and Technology, Babol, Iran

development (NPD) process. While in both fields, NPD and supply chain management, have received a considerable attention, they are hardly considered together. Since this paper deals with both the multi-objective optimization and integration of NPD and SC, a brief introduction to the concepts of multi-objective problems in supply chain and integration of NPD and SC is presented below.

In the next two subsections of this paper, the existing literature on multi-objective optimization problems and integration of NPD and SC is reviewed. “**Mathematical model and problem descriptions**” presents the mathematical model and model’s assumptions. The solving methodology is defined in “**Solving methodology**”. “**Applications and comparisons**” presents the computational results and, finally, conclusions are reported in “**Conclusions**”.

### Multi-objective optimization problem in SC

Real SCs are to be optimized simultaneously considering more than one objective. Problems which try to optimize many conflicting objectives simultaneously are called multi-objective optimization problems and have many optimal solutions. The general form of such problem is (Van Veldhuizen 1999):

$$\begin{aligned} \min F(x) &= (f_1(x), f_2(x), \dots, f_m(x)) \\ \text{s.t.} \quad &\begin{cases} g_i(x) \leq 0 & i = 1, \dots, q \\ h_j(x) = 0 & j = 1, \dots, p, \end{cases} \end{aligned}$$

where  $x = (x_1, x_2, \dots, x_n) \in X \subset R^n$  is called decision variable and  $X$  is  $n$ —dimensional decision space.  $f_m(x)$  is the  $m$ -th objective,  $g_i(x)$  is the  $i$ -th constraint inequality and  $h_j(x)$  is the  $j$ -th constraint equation.

Different methodologies found in literature for treating multi-objective optimization problems are the weighted-sum method, the  $\varepsilon$ -constraint method, goal programming method, lexicography methods, LP-metric, maxi-min, fuzzy method, etc. (Chen et al. 2003; Chen and Lee 2004; Guilléna et al. 2004; Hung 2011). In all these mentioned methods, multiple objectives are combined and form a single objective problem and then the optimal solution is obtained. As in real world situations different alternatives are willing by the decision makers, such solutions may not satisfy the decision maker.

For this reason, due to the exponential growth of the problem size and complexity, numerous nature-based multi-objective algorithms are introduced for solving mathematical optimization models. Among various multi-objective optimization algorithms, multi-objective evolutionary algorithms (MOEAs), the most significant ones are multi-objective genetic algorithms (GA) (Deb et al. 2002, Mohapatra et al. 2015), multi-objective particle swarm optimization algorithms (MPSO) (Kotinis 2014; Zhang et al. 2013;

Niknam et al. 2011; Tsaia et al. 2010), multi-objective evolutionary algorithms (Shin et al. 2011; Zhu et al. 2014; Tana et al. 2014), multi objective immune clone algorithms (Shang et al. 2012), group search optimizer (Wang et al. 2012), multi-objective frog leaping algorithms (Taher et al. 2010; Safaei Arshia et al. 2014) and so on. However, hybrid algorithms have received considerable attention recently. For instance, Govindan et al. (2015) combined multi-objective electromagnetism mechanism algorithm and adapted multi-objective variable neighborhood search; Bandyopadhyay and Bhattacharya (2013) proposed a modified NSGA-II; Diabat (2014) also combined genetic algorithm with simulated annealing algorithm.

In general, supply chain models naturally lead to complex, large-scale mathematical models which are hard to solve optimally in most real cases. Also, as real SC problems consider more than one objective, the appearance of conflict objectives does not allow simultaneous optimal solutions for all objectives. Thus, for achieving near optimal solutions in large size instances, applying MOEAs are helpful.

Sadeghi et al. (2014) presented a multi-objective combinatorial optimization model of a supply chain problem including one-vendor multi-retailers considering a vendor-managed inventory approach. Their proposed model included two objectives, minimization of inventory cost as the first objective and maximization of the system reliability of the machines that produce the goods as the second. Since the developed model was NP-hard, they applied two multi-objective genetic algorithms, namely, NSGA-II and NRGA, to find Pareto fronts. Latha Shankar et al. (2013) formulated a three-echelon SC network model for the optimal facility location and capacity allocation decisions. While making strategic decisions, they considered fixed location and variable material cost, production, inventory and transportation costs. Their proposed model contained two objective functions, minimizing total SC cost and maximizing fill rate. They applied an intelligent multi-objective hybrid particle swarm optimization algorithm optimizer. As another study, Bandyopadhyay and Bhattacharya (2014) proposed a tri-objective problem for a two-echelon serial supply chain. The considered objectives were the minimization of the total cost the chain, minimization of the variance of order quantity and minimization of the total inventory. To solve the model, they applied a proposed modified NSGA-II. Soleimani and Kannan (2015) developed a deterministic multi-echelon, multi-product, multi-period model for a closed-loop SC network and presented a new hybrid particle swarm optimization and genetic algorithm to solve various kinds of problems. Then a complete computational analysis was under taken to validate their proposed algorithm.

In this paper, a new application of meta-heuristic based on NSGA-II is demonstrated for the proposed tri-objective



optimization of SC network considering NPD. The algorithm provides a set of compromised solutions called Pareto optimal solutions which optimize all the conflicting objectives. Then, the solutions satisfying specific criteria can be chosen from the set of Pareto optimal solutions by the decision maker.

### Integration of NPD and SC

A supply chain network refers to a network which consists of suppliers, manufactures, distributes and customer groups to create products (see Fig. 1). Nowadays, as customer’s preferences change rapidly, to survive in a highly competitive industry, NPD process received more attention. To overcome these challenges, integration of the NPD process and the SC can offer a sustainable competitive advantage to achieve success according to the current competitive environment in the market place. While NPD and SC have drawn considerable attention from the researchers, separately, there is little effort in the literature for covering SC and NPD together.

A generously persuasive parameter for NPD problems in a SC is the strategy of introduction a new product and phasing out the old products which are developed in the planning horizon. As new products appear in the market, the old products could become obsolete, and then phased out. Before launching a new product, a manufacturer must decide the timing of the product launch and the production/sales plan over the planning horizon. According to Billington et al. (1998) there are two rollover strategies namely single product roll and dual product roll to introduce new products to the market. In the first strategy, single product roll, the time phasing out the old product and the time introducing the new product are the same which means that as the new product is introduced the old product is phased out. As such, in dual rollover strategy as the new product is introduced the firm continues producing the old product, in other words, the coexistence of both products is allowed during a certain period of time. In the proposed model the single product rollover is assumed.

SCs through which new products are manufactured need flexibility and responsiveness elements. Each element of a SC echelons may have options which are able to satisfy a

required function as the procurement, transportation and production of a product. These decisions after a new product design are completed get more importance because it increases the costs (Simchi-Levi et al. 2000). Narahariseti and Karimi (2010) proposed the SC redesign and new process introduction in multi-purpose plants. They added some production and inventory facilities, distribution and customer centers to their chain. An integrated optimization model for configuring the SC of new product is developed by Amini and Li (2011). Their work was a first attempt to model the interaction between new product diffusion and SC configuration. The aim of their research is the configuration of SC subject to demand dynamics and other SC parameters such as lead-time and inventory. Also, Li and Amini (2012) developed an integrated optimization model which allowed multiple-sourcing and safety stock placement decisions in coordinate with the demand dynamics during the new product diffusion process throughout its life cycle. As another study, Nepal et al. (2011) have extended a multi-objective optimization model for SC configuration during product development. Their presented model consists of two objectives: maximization of the total compatibility index in strategic alliance and minimization of the total SC costs. Their model improves SC efficiency and stability by jointly considering sourcing, inventory costs and compatibility decisions during the configuration of the SC. They solved their model by using genetic algorithm. As another study, Nager et al. (2014) developed a multi-objective two-stage stochastic programming supply chain model that incorporated imprecise production rate and supplier capacity under scenario dependent fuzzy random demand associated with new product supply chains. The objectives which they considered were maximizing the supply chain profit, achieving desired service level and minimizing the financial risk. They used the possibility measure to quantify the demand uncertainties and solved the model using fuzzy linear programming approach. Jafarian and Bashiri (2014) provided a five echelon dynamic SC model. Their proposed model considers the time of new product lurching in the SC, which is optimized with SC configuration simultaneously. In addition, production, sales, transportation planning and their lead times are considered in the model. In their proposed model each

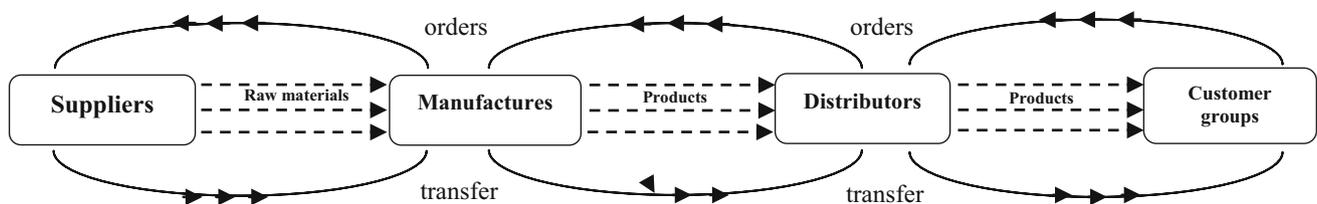


Fig. 1 The stages of a four-echelon SC

firm individually decides to introduce new products and their model considers developing a single product. Alizadeh Afrouzy et al. (2016) developed a multi-echelon multi-product multi-period optimization model for SC configuration and NPD. Their proposed model incorporates three kinds of products: products which are produced during the planning horizon, products which are decided to be developed and products that are newly introduced during the planning horizon. Their indicated model was capable to determine the optimized launching time and phasing time during a planning horizontal time in order to maximum the total profit. They applied priority based GA to solve the proposed model. Given the limited number of studies related to the optimization of SCs considering NPD, and in realm contexts in particular, this study tries to contribute to the literature by proposing a multi-objective multi-period model to optimize the profit of SCs considering NPD in accordance to the other two objectives, maximizing the customer satisfaction and maximizing the NPD production. Unlike previous researches, the model considers three categorizes of products; current products, developed products and completely new products. The obtained model belongs to NP-Hard class of the optimization problems, thus, NSGA-II is developed to find a near-optimum solution.

## Mathematical model and problem descriptions

### Model assumptions

In this paper, a tri-objective, multi-period, multi-product and multi-echelon network including customer groups, distribution centers, manufacturing centers and suppliers is inquired. In this chain, raw materials are supplied from suppliers to manufactures to be transformed into final products and the products are shipped to distribution centers and then according to demands of customer groups they are responded. NPD is assumed to respond to marketplace changes and customers' tastes to gain or maintain competitive advantage. Accordingly, the company must have the ability to develop and produce new products during the planning horizon and then flexibility is needed for the chain parts to respond to marketplace changes to gain or maintain competitive advantage. As the model is considered multi-product, the categorization of the products is old products, developing/changing products and completely new products. The rollover which is considered in the proposed model is a single-product roll for developing products. This rollover strategy considers the phasing out time, thereby only one of the old or developed products is produced in a moment.

## Model formulation

Definitions of variables and parameters in the multi-echelon supply chain network are summarized below. Then, a brief description about the objective functions and constraints of the model are presented.

### Indexing sets

|     |                          |     |                               |
|-----|--------------------------|-----|-------------------------------|
| $t$ | Index of period,         | $T$ | Number of periods             |
| $s$ | Index of supplier,       | $S$ | Number of suppliers           |
| $f$ | Index of manufacturer,   | $F$ | Number of manufacturers       |
| $d$ | Index of distributor,    | $D$ | Number of distributor centers |
| $g$ | Index of customer group, | $G$ | Number of customer groups     |
| $r$ | Index of raw material,   | $R$ | Number of raw materials       |
| $i$ | Index of products        | $I$ | Number of products            |

As discussed in the previous subsection, the products which the company wants to produce during the planning horizon contain three groups. As shown by the indices below, the first group indicates the products that are being produced by the company, the second group relates to the products which are the developed form of the old products and is decided to be produced and the third group contains the completely new products that the company decides to produce during the planning horizon. The indices are as the following:

$$i = \begin{cases} o \text{ (old)} & o = 1, \dots, O \\ j \text{ (old} \rightarrow \text{new)} & j = O + 1, \dots, J \\ n \text{ (new)} & n = J + 1, \dots, I \end{cases}$$

### Parameters

|            |   |
|------------|---|
| $P_{oj}$   | 1, if product $o$ can be developed to product $j$ , 0, otherwise        |
| $D_{git}$  | Demand of customer group $g$ for product $i$ in period $t$              |
| $PR_{srt}$ | Per unit price of raw material $r$ in the supplier $s$ in period $t$    |
| $PD_i$     | Per unit price of product $i$   |
| $B_{ir}$   | Number of the raw material $r$ needed to produce product $i$            |
| $dl_i$     | New product designing duration of product $i$                           |
| $CM_{ft}$  | Capacity of manufacturer $f$ in period $t$                              |
| $CD_d$     | Capacity of distributor $d$   |
| $CS_{srt}$ | Capacity of supplier $s$ for providing raw material $r$ in period $t$   |
| $CP_{fi}$  | Cost of producing product $i$ in manufacturer $f$                       |
| $fCP_{fi}$ | Fixed cost of producing product $i$ in manufacturer $f$                 |
| $TP_{fi}$  | Hour needed to produce per unit of product $i$ in manufacturer $f$      |
| $TCS_{sf}$ | Transportation cost between supplier $s$ and manufacturer $f$           |
| $TCM_{fd}$ | Transportation cost between manufacturer $f$ and distributor center $d$ |

|            |   |
|------------|---|
| $TCD_{dg}$ | Transportation cost between distributor center $d$ and customer group $g$ |
| $CDN_i$    | Cost of designing new product $i$   |
| $M$        | Big number  |

Decision variables

|              |  |
|--------------|--|
| $SRM_{sfrt}$ | Amount of raw material $r$ transferred from supplier $s$ to manufacturer $f$ in period $t$ |
| $SMD_{fdit}$ | Amount of product $i$ transferred from manufacturer $f$ to distributor $d$ in period $t$   |
| $SDG_{dgit}$ | Amount of product $i$ transferred from distributor $d$ to customer group $g$ in period $t$ |
| $Q_{fit}$    | Production quantity of product $i$ for manufacturer $f$ in period $t$                      |
| $ET_{it}$    | Binary variable which represents the new product entering time to distributor centers      |
| $DP_{it}$    | 1, if company decide to design product $i$ in period $t$ , 0, otherwise                    |

Objective functions

The objectives of the problem are formulated as the following:

$$\begin{aligned}
 \text{Maximize } Z_1 = & \sum_d \sum_g \sum_i \sum_t PD_i SDG_{dgit} \\
 & - \sum_s \sum_f \sum_r \sum_t PR_{srt} SRM_{sfrt} \\
 & - \sum_f \sum_i \sum_t (CP_{fi} Q_{fit} + fCP_{fi} DP_{it}) \\
 & - \sum_t \sum_s \sum_f \sum_r SRM_{sfrt} TCS_{sr} \\
 & - \sum_t \sum_d \sum_g \sum_i SDG_{dgit} TCD_{dg} \\
 & - \sum_t \sum_f \sum_d \sum_i SMD_{fdit} TCM_{fd} \\
 & - \sum_t \sum_i CDN_i DP_{it}
 \end{aligned} \tag{1}$$

$$\text{Maximize } Z_2 = \sum_t \sum_i \min_g \left\{ \frac{\sum_d SDG_{dgit}}{D_{git}} \right\} \tag{2}$$

$$\text{Maximize } Z_3 = \sum_{i \in j,n} \sum_t \sum_f Q_{fit} \tag{3}$$

The aim of the first objective function which consists of seven cost ingredients is to maximize the total profit at the end of the planning horizon. The first term computes the total revenue of sales. The second term of the objective function

represents the purchased quantity for each raw material cost and the next term is the production cost of each product. The next three terms are the transportation costs. Finally, term seventh formulate the cost of designing new products.

The second objective function maximizes the satisfaction level of customer demands. Its aim is to control the amount of lost sales when the chain has the same profit in order to satisfy more customers.

In addition to profit and customer satisfaction, the third objective aims to maximize the production of developed and new products. Although production of developed and new products may not be profitable but the firms have to introduce them to the market in order to survive in the competitive marketplace.

Constraints

$$\sum_s SRM_{sfrt} = \sum_i B_{ir} Q_{fit} \quad \forall f, t, r \tag{4}$$

$$Q_{fit} = \sum_d SMD_{fdit} \quad \forall f, i, t \tag{5}$$

$$\sum_f SMD_{fdit} = \sum_g SDG_{dgit} \quad \forall d, i, t \tag{6}$$

$$\sum_d SDG_{dgit} \leq D_{git} \quad \forall g, i, t \tag{7}$$

$$\sum_f SRM_{sfrt} \leq CS_{srt} \quad \forall s, r, t \tag{8}$$

$$Q_{fit} TP_{fi} \leq CM_{ft} \quad \forall f, i, t \tag{9}$$

$$\sum_f \sum_i SMD_{fdit} \leq CD_d \quad \forall d, t \tag{10}$$

$$Q_{fit} \leq M \times ET_{it} \quad \forall f, t, i \tag{11}$$

$$\sum_{h=t+dl_i+1}^T ET_{oh} < DP_{jt} P_{oj} \quad \forall t, i \in o, j \tag{12}$$

$$\sum_{h=1}^{t+dl_i} ET_{ih} < DP_{it} \quad \forall t, i \in j, n \tag{13}$$

$$\sum_t DP_{it} = 1 \quad \forall i \in j, n \tag{14}$$

$$\begin{aligned}
 SRM, SMD, SDG, Q \geq 0 \text{ and integer } DP, ET \in \{0, 1\} \\
 \forall i, t, f, d, s, g, r
 \end{aligned} \tag{15}$$

The Constraints are presented from Eqs. (4)–(15) and are described briefly in the following. Constrains (4)–(6) describe the amount of flow between the echelons of the chain. Constraint (7) represents that the sales of each product at each customer group during each time period should not exceed its corresponding demand. Inequality (8) ensures that the total raw materials transported from each

available supplier to the manufacturers cannot exceed the supplier capacity. Inequalities (9) and (10) also state the same issue about the maximum capacity of available plants and distributors, respectively. Inequality (11) describes the production allowance. Inequality (12) is related to the second category of the products, developing products, which describes the rollover strategy. It states that after deciding to develop or change a product, the manufacturers are able to launch them after the designing duration time. Inequality (13) describes the start of the designing time. Constraint (14) assures that during the planning horizon each developing or new products are decided to be designed once in the planning horizon. Constraint (15) is the logical binary and non-negativity integer requirements on the decision variables.

### Solving methodology

Most real-world optimization problems belong to the class of NP-hard problems. In order to solve NP-hard problems, there are not provably efficient algorithms. Frequently, exact methods cannot solve this class of problems in normal and reasonable time. To optimize this class of problems meta-heuristic algorithms are suitable tools.

For experimentation, this section presents the application of the proposed model along with the proposed NSGA-II algorithm on some random test problems. Some sets of small, medium and large sized instances are considered to evaluate the performance of the solution approach. In addition, since no benchmark is available in the literature to verify and validate the results obtained by NSGA-II, another popular MOEA called NPGA is suggested to solve the problem as well. To do this, some preliminary concepts and principles of NSGA-II and NPGA are first reviewed. Then, the required structures, i.e. chromosome encoding and decoding, crossover and mutation operators are described and the NSGA-II and NPGA are developed.

### Introduction to non-dominated sorting genetic algorithm optimization

One of the first evolutionary algorithms which employed the concept of Pareto optimality in solving multi-objective problems is Non-dominated Sorting Genetic Algorithm (Srinivas and Deb 1995). Then Deb et al. (2002) developed a powerful meta-heuristic technique known as NSGA-II which in comparison with NSGA has less computational complexity, less parameters, elitist strategy and is an efficient constraint-handling method. These features have made the NSGA-II very successful and popular in a wide range of engineering problems and some practical works in this area.

As earlier discussed, to improve NPD activities in a SC network the proposed multi-objective model has to be solved. Thus, a multi-objective genetic algorithm on the basis of the NSGA-II is utilized to provide Pareto fronts of the conflicting objectives. Since no benchmarks could be found, the purpose of employing the second algorithm (NPGA) is to verify the results obtained by NSGA-II.

It is noticed that since most real world cases are large scale and NP-hard, it justifies the use of meta-heuristics; however they don't guarantee to achieve optimal solutions and usually obtain near optima solutions (Talbi 2009; Coello et al. 2002).

The main structure of the NSGA-II which has been so far applied to many complex multi-objective optimization problems successfully is presented as follows.

In an evolution cycle of the standard NSGA-II, the optimization begins by an initial population that is randomly generated and then evaluated by all objective functions. The size of the population is one of the NSGA-II parameters and is often known as *pop size*. For each and every problem, the population size will depend on the complexity of the problem. Practically, a population size of around 100 individuals is quite frequent, but anyway this size can be changed according to the time and the available memory on the machine compared to the quality of the result to be obtained.

Considering the obtained chromosomes, the population is sorted based on the non-domination principle. Based on the tournament selection the offspring population is created and then the crossover and mutation operators are applied to the obtained chromosomes. Next, the population is sorted based on two criteria: (i) rank and (ii) crowding distance. First, each chromosome is assigned a rank number equal to its non-domination level and then the crowding distance operator is applied between two solutions with equal non-domination rank. Consequently, the required population is organized from the top elements of the front without losing good solutions (elitism). Solutions belonging to the best non-dominated fronts are directly transferred to the mating pool to create the next generation. These steps are repeated until a stopping condition is met. Regarding the termination criteria and iterating the stages mentioned above, the algorithm hopefully presents the best and approximate Pareto optimal solutions.

Important parameters, such as chromosome coding, chromosome decoding and NSGA-II operators are described in more details as follows.

### Chromosomes

As same as GA, NSGA-II starts with an initial set of random solutions generally called population. Each individual in the population is called a chromosome. Each chromosome consists of genes and as it is known, a gene in a

chromosome is characterized by two factors: locus, the position of the gene within the structure of chromosome and allele, the value the gene takes. In priority-based encoding, the position of a gene is used to represent a node (source/depot in transportation network), and the value is used to represent the priority of the corresponding node for constructing a tree among candidates (Gen and Cheng 2000).

The proposed SC problem involves four-echelons (suppliers, manufacturers, distributor centers, and customer groups). Each chromosome consists of three segments, in which the segments are used to demonstrate the relationship among the four echelons. The first part (Segment 1, Fig. 2) with a dimension of  $[(S + F) \times R]$  represents the relationship between suppliers and manufacturers in each period which are given as integer numbers belong to 1 to  $[(S + F) \times R]$  as a priority value. Production of the products in different periods and distribution quantities of the products from manufacturers to distributor centers are considered in the second part (Segment 2, Fig. 2) with a dimension of  $[(D + F) \times I]$ . Finally, quantities of the distributed products from distributor centers to customer groups in each period are decisions that should be determined in the third part (Segment 3, Fig. 2) with a dimension of  $[(D + G) \times I]$ . As decisions are made for  $I$  products and  $R$  raw materials in  $T$  periods for  $S$  suppliers,  $F$  manufactures,  $D$  distributor centers, and  $G$  customer groups, the full length of a chromosome is the sum of the lengths of these three parts multiplied by the numbers of periods  $[(D + G) \times I + (D + F) \times I + (S + F)$

$\times R] \times T$ . As an example, the structure of a chromosome for one period with two raw materials, five suppliers, two manufacturers, four distributors, two customer groups, three products is shown in Fig. (2).

Since the proposed model improves NPD, as the company decides to produce new products and also develop current products in some periods, this has to be shown in the chromosome. This is done by randomly generating a period and depending on the new product designing duration of each product parameter, the following conversion statements are going to be done:

1. For products which are decided to be produced completely new and for the first time during the planning horizon, depending on the period obtained by randomly generated period plus the designing duration parameter, the genes value for this product is updated in a way that the priority of the related product for the periods up to the determined period is considered zero and the rest remain unchanged.
2. For products which are decided to be developed to another product, depending on the period obtained by sum of randomly generated period and the designing duration parameter, the value of the genes for these products are updated in a way that the priority of the old product from the determined period up to the end of the planning horizon is considered zero and on contrary, the priority of the developed product for the periods up to the determined period is considered zero and the rest remain unchanged.

**Segment 3: Distributor centers ( $d$ ) → Products ( $i$ ) → Customer groups ( $g$ )**

| Product 1    |   |   |           |    |   | Product 2    |   |   |           |   |    | Product 3    |    |    |           |   |    |   |   |    |   |    |   |
|--------------|---|---|-----------|----|---|--------------|---|---|-----------|---|----|--------------|----|----|-----------|---|----|---|---|----|---|----|---|
| Distributors |   |   | customers |    |   | Distributors |   |   | customers |   |    | Distributors |    |    | customers |   |    |   |   |    |   |    |   |
| 1            | 2 | 3 | 4         | 1  | 2 | 1            | 2 | 3 | 4         | 1 | 2  | 1            | 2  | 3  | 4         | 1 | 2  |   |   |    |   |    |   |
| [            | 4 | 8 | 2         | 10 | 7 | 12           | ] | [ | 17        | 1 | 15 | 5            | 14 | 18 | ]         | [ | 16 | 6 | 9 | 13 | 3 | 11 | ] |

**Segment 2: Manufactures ( $f$ ) → Products ( $i$ ) → Distributor centers ( $d$ )**

| Product 1    |   |              |   | Product 2    |    |              |   | Product 3    |   |              |    |    |   |    |   |   |    |   |   |    |    |   |   |
|--------------|---|--------------|---|--------------|----|--------------|---|--------------|---|--------------|----|----|---|----|---|---|----|---|---|----|----|---|---|
| Manufactures |   | Distributors |   | Manufactures |    | Distributors |   | Manufactures |   | Distributors |    |    |   |    |   |   |    |   |   |    |    |   |   |
| 1            | 2 | 1            | 2 | 3            | 4  | 1            | 2 | 1            | 2 | 3            | 4  |    |   |    |   |   |    |   |   |    |    |   |   |
| [            | 2 | 11           | 6 | 13           | 15 | 5            | ] | [            | 3 | 7            | 14 | 10 | 4 | 16 | ] | [ | 18 | 1 | 9 | 12 | 17 | 8 | ] |

**Segment 1: Suppliers ( $s$ ) → Raw materials ( $r$ ) → Manufactures ( $f$ )**

| Raw material 1 |   |    |              |   |   | Raw material 2 |    |   |              |    |   |   |    |   |   |   |   |
|----------------|---|----|--------------|---|---|----------------|----|---|--------------|----|---|---|----|---|---|---|---|
| Suppliers      |   |    | Manufactures |   |   | Suppliers      |    |   | Manufactures |    |   |   |    |   |   |   |   |
| 1              | 2 | 3  | 4            | 5 | 1 | 2              | 1  | 2 | 3            | 4  | 5 | 1 | 2  |   |   |   |   |
| [              | 6 | 12 | 10           | 2 | 9 | 5              | 14 | ] | [            | 11 | 1 | 8 | 13 | 3 | 7 | 4 | ] |

**Fig. 2** An example of representation for multi-product multi-echelon SC for one period

A chromosome generated by the above procedure is feasible, and the population will not lose diversity.

#### *Decoding procedure for the SC problem*

Transportation tree between customer groups and distributor centers, distribution centers and manufacturers, manufacturers and suppliers which are obtained with decoding of the third, second and first segment of a chromosome, respectively, is generated by sequential arc appending between sources and depots. At each step of the decoding procedure, the highest priority in the chromosome is determined. Although its position gives the product type, it gives a source (depot) to realize a shipment for the product type, too. Depending on the selected source (depot), a depot (source) is determined considering minimum transportation cost and an arc between them is added to the corresponding tree (Altıparmak et al. 2009).

The chromosome of SC is decoded on the backward direction. More specifically, the following strategy is taken:

- First, a transportation tree between customer groups and distribution centers is obtained.
- In the second step, by the outputs of the third segment and the periods obtained for introducing new and developed products from the third segment, the transportation tree between distribution centers and manufacturers is derived for the second segment.
- Lastly, the transportation tree between manufacturers and suppliers is obtained for the first segment by the outputs of the second segment. The difference of decoding the first segment from the previous segments is that the obtained periods do not affect this segment and it remains unchanged and the general decoding procedure is applied.

#### **Crossover and mutation operators**

In an NSGA-II, similar to GA, the offspring population is determined by a set of operators that recombine and mutate selected members of the current population.

To perform the crossover, two parents are selected based on tournament selection. Then, their periods for developing and introducing new products are exchanged. In other words, when two parents are selected by tournament selection, their priority based chromosomes are updated by the exchanged periods and then is decoded and the fitness function is obtained.

The swap mutation is considered as the mutation operator. The procedure of mutation is summarized as follows: Firstly, a chromosome is randomly extracted. Then, the swap mutation is done. Since the second segments priority based chromosome is related to the third segment, mutation

is done only on the third and first segment which are independent. A period is randomly selected for each segment. The operator selects two gens from the corresponding segments and exchanges their places. Since the proposed chromosome considers NPD, note that for the third segment the gens with nonzero value must be swapped.

#### **Introduction to non-dominated ranking genetic algorithm optimization**

According to Al Jadaan et al. (2008) in NPGA, roulette wheel selection strategy is considered instead of tournament selection in NSGA-II. In the selection strategy of NPGA, they considered two ranked- based roulette wheel tires; the first tier is for selecting the front, and the second tier is for selecting solution from the front. The individuals in a front are ranked based on their crowding distance, and the fronts ranked based on the non-dominated rank. Initially, a random parent population is created. The procedure is different after the initial generation. Now, two tiers ranked based roulette wheel selection is applied and the solutions belonging to the best non-dominated set to the less probability are chosen. Then the same crossover and mutation operators considered in NSGA-II are used. In addition, to stop the NPGA algorithm, a similar procedure is considered as the one for NSGA-II.

#### **Applications and comparisons**

In this section, the application of the two MOGA's, NSGA-II and NPGA, is presented on some random test problems. Then, they are compared based on some comparison metrics and statistical approaches. To do so, the parameters of the algorithms are tuned via Taguchi method. It should be mentioned that, both algorithms are coded by Matlab 8.2.0.701 (R2013b) software to complete the process of comparison. All computations are run on an Intel(R) Core(TM) i7-4501U CPU@ 2.60 GHz processor laptop.

#### **Multi-objective performance measures**

In order to evaluate the performance of multi-objective algorithms, the algorithms are assessed according to multi-objective measures. The metrics which are used to evaluate the performance of the two meta-heuristic algorithms proposed are as follows:

1. Number of Pareto solutions (NPS) which counts the Pareto solutions in the Pareto optimal front.
2. Mean Ideal Distance (MID) which measures the convergence of the Pareto fronts is provided in expression (16) below (Deb 2001):

$$MID = \frac{\sum_{i=1}^N d_i}{N}, \tag{16}$$

where  $N$  is number of the solutions in the Pareto optimal front and  $d_i$  is the Euclidian distance between solution  $i$  and the origin and is given as the expression below.

$$d_i = \sqrt{f_{1i}^2 + f_{2i}^2 + \dots + f_{ki}^2}, \tag{17}$$

where  $f_{ki}$  is the value of the  $k$ -th objective function of the  $i$ -th solution.

3. The Divergence Metric (DM) which is provided in expression (18) measures the extension of the Pareto front (Deb 2001).

$$DM = \frac{\sqrt{\sum_{i=1}^N (\bar{d} - d_i)^2}}{N}, \tag{18}$$

where  $d_i$  is the Euclidian distance between solution  $i$  and solutions  $j$  and is given by expression (19) and  $\bar{d}$  is the mean value of the evaluated  $d_i$ .

$$d_i = \min_{j=1,2,\dots,N} \sum_{k=1}^M (f_{ki} - f_{kj}) \quad \forall i = 1, \dots, N, \tag{19}$$

where  $f_{ki}$  and  $f_{kj}$  are values of the  $k$ -th objective function of the  $i$ -th and  $j$ -th solutions.

4. The CPU time of running the algorithms to reach near optimum solutions.

### Setting model parameters

By considering three problem sizes as small, medium and large, the model’s parameters are set. 12 test problems are considered and generated randomly. They are categorized based on different numbers of supply chain elements. The considered tests and their characteristics are presented in Table 1. Moreover, the following information is also given (Alizadeh Afrouzy et al. 2016).

- The demand rate of the customer group  $g$  for each product in a period follows a uniform distribution, i.e.,  $D_{git} \sim \text{Uniform} [180, 500]$ .
- The designing cost of new products and the price of each final product have uniform distribution as  $CDN_i \sim \text{Uniform} [170, 300]$  and  $PD_i \sim \text{Uniform} [650, 850]$ .
- Procurement cost of raw material  $r$  in suppliers in a period follows a uniform distribution, i.e.,  $PR_{srt} \sim \text{Uniform} [5, 18]$ .
- Transportation cost from supplier  $s$  to manufacturers follows a uniform distribution, i.e.,  $TCS_{sf} \sim \text{Uniform} [0.12, 0.19]$ .
- Transportation cost from manufacturer  $f$  to distribution centers follows a uniform distribution, i.e.,  $TCM_{fd} \sim \text{Uniform} [0.1, 0.7]$ .
- Transportation cost from distribution center  $d$  to customer groups follows a uniform distribution, i.e.,  $TCD_{dg} \sim \text{Uniform} [1.2, 1.9]$ .
- Production fixed cost and cost of producing products in manufactures have uniform distributions as  $fCP_{fi} \sim \text{Uniform} [2, 8]$  and  $CP_{fi} \sim \text{Uniform} [16, 27]$ .

**Table 1** Categorization of small-, medium- and large-sized random test problem

| No. of problems | Raw materials | Suppliers | Manufacturers | Distributors | Customer groups | Products | Periods |
|-----------------|---------------|-----------|---------------|--------------|-----------------|----------|---------|
| Small sized     |               |           |               |              |                 |          |         |
| 1               | 6             | 3         | 3             | 4            | 2               | 4        | 10      |
| 2               | 6             | 3         | 4             | 4            | 2               | 4        | 10      |
| 3               | 7             | 3         | 4             | 4            | 3               | 4        | 12      |
| 4               | 7             | 4         | 5             | 5            | 4               | 6        | 12      |
| Medium sized    |               |           |               |              |                 |          |         |
| 5               | 7             | 5         | 4             | 5            | 4               | 6        | 13      |
| 6               | 8             | 5         | 5             | 5            | 5               | 7        | 14      |
| 7               | 8             | 5         | 5             | 6            | 5               | 8        | 15      |
| 8               | 8             | 6         | 6             | 6            | 5               | 8        | 15      |
| Large sized     |               |           |               |              |                 |          |         |
| 9               | 9             | 6         | 6             | 7            | 6               | 8        | 18      |
| 10              | 9             | 8         | 6             | 7            | 7               | 10       | 18      |
| 11              | 10            | 8         | 7             | 7            | 7               | 10       | 18      |
| 12              | 12            | 10        | 7             | 7            | 7               | 10       | 18      |

**Table 2** Algorithms’ parameters along with their levels

| No. of factors | Level Parameters  | Low 1 | Medium 2 | High 3 |
|----------------|-------------------|-------|----------|--------|
| 1              | Max <sub>It</sub> | 100   | 150      | 200    |
| 2              | N <sub>pop</sub>  | 30    | 50       | 70     |
| 3              | P <sub>c</sub>    | 0.7   | 0.8      | 0.9    |
| 4              | P <sub>m</sub>    | 0.2   | 0.15     | 0.1    |

**Table 3** The orthogonal arrays of the L9 design

| No. of runs | Factors           |                  |                |                |
|-------------|-------------------|------------------|----------------|----------------|
|             | Max <sub>It</sub> | N <sub>pop</sub> | P <sub>c</sub> | P <sub>m</sub> |
| 1           | 1                 | 1                | 1              | 1              |
| 2           | 1                 | 2                | 2              | 2              |
| 3           | 1                 | 3                | 3              | 3              |
| 4           | 2                 | 1                | 2              | 3              |
| 5           | 2                 | 2                | 3              | 1              |
| 6           | 2                 | 3                | 1              | 2              |
| 7           | 3                 | 1                | 3              | 2              |
| 8           | 3                 | 2                | 1              | 3              |
| 9           | 3                 | 3                | 2              | 1              |

- The designing duration time for developing a product and producing a new product is assumed 4 and 2 periods, respectively.
- For simplicity, in all 12 test problems one product is going to be developed and one product is going to be produced for the first time.
- The number of raw materials needed to consume a product follows a uniform distribution as  $B_{ir} \sim \text{Uniform} [0, 4]$ .

- Capacities of the manufacturers and distributor centers, and supply capacity of raw material  $r$  in a period follow uniform distributions as  $CM_{ft} \sim \text{Uniform} [800, 1500]$ ,  $CD_d \sim \text{Uniform} [1200, 1900]$  and  $CS_{srr} \sim \text{Uniform} [3100, 3200]$ , respectively.
- The hours needed for producing per unit of product  $i$  in manufacturers follow a uniform distribution, i.e.,  $TP_{fi} \sim \text{Uniform} [3, 6]$ .

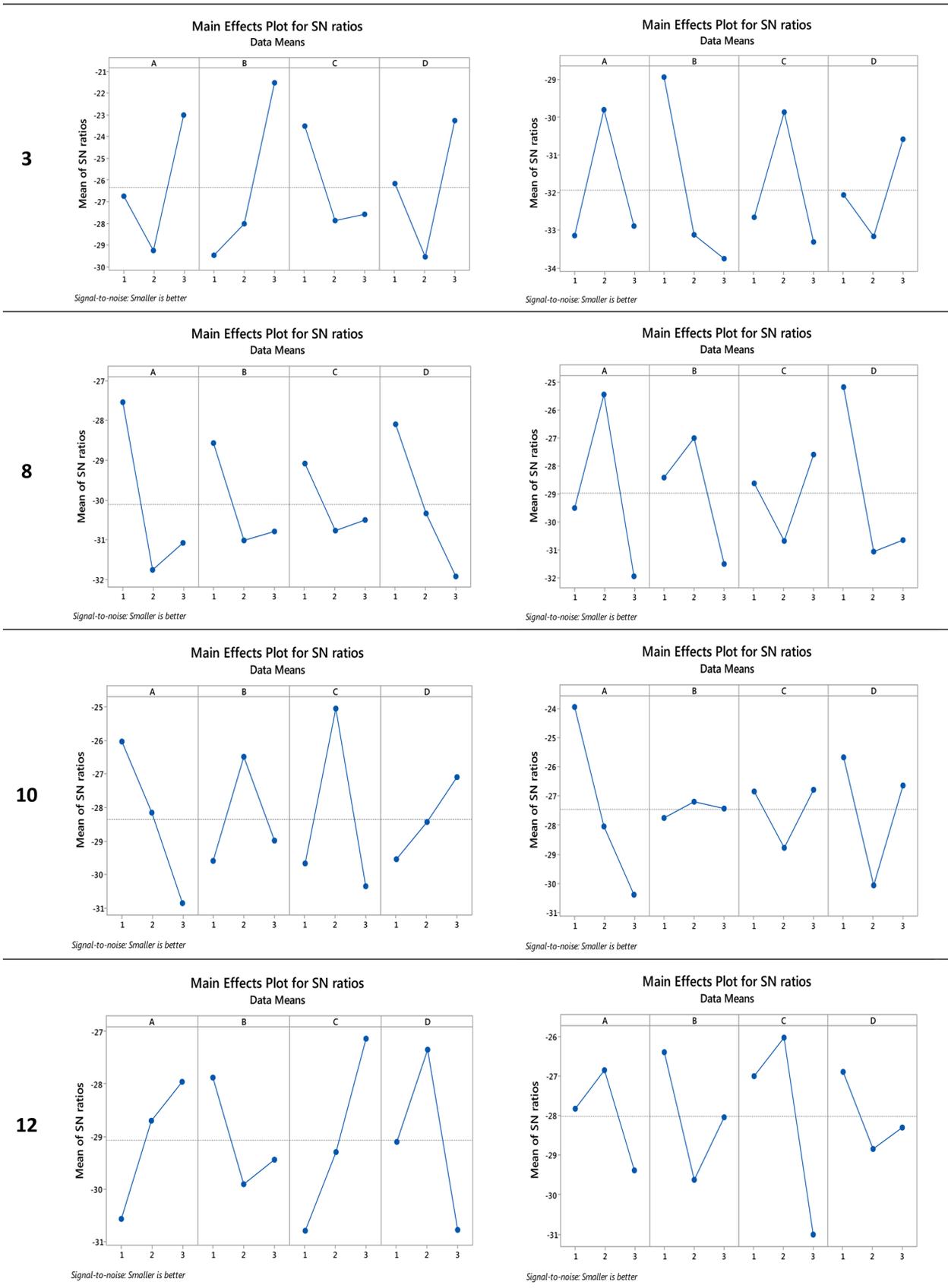
**Tuning algorithms’ parameters**

There are several ways to tune the parameters of the evolutionary algorithms. As choosing suggested amounts by other researches or a trial and error procedure are different ways for tuning the algorithm’s parameters, but methods maybe used to certify qualifications of the solutions which is based on some comparison metrics and statistical approaches. Here the Taguchi method is implemented. Taguchi (1986) developed a family of Fractional Factorial Experiment (FFE) matrices that eventually reduces the total number of the experiments for tuning considerably, but still provides sufficient information. In order to utilize the Taguchi method, the levels of the factors of the considered algorithms are first determined. The aim is to find the amounts of the algorithm’s parameters as input variables to gain an optimal result. Thus, the factors which have an effect on the result are considered as: number of iterations (Max<sub>It</sub>); number of population (N<sub>pop</sub>); crossover rate (P<sub>c</sub>) and mutation rate (P<sub>m</sub>). The levels of the factors are presented in Table 2.

According to Table 2, it can be seen that three levels are considered for each factor involved in the algorithms. By the use of Minitab Software, the L9 design which is shown

**Table 4** Algorithms’ comparison results by Taguchi method

| No. of ex. | NSGA-II |            |            |         | NRGA |            |            |         |
|------------|---------|------------|------------|---------|------|------------|------------|---------|
|            | NPS     | MID        | DM         | W       | NPS  | MID        | DM         | W       |
| 1          | 15      | 9.9496e+06 | 6.3055e+04 | 7.4584  | 10   | 9.5791e+06 | 2.4074e+05 | 11.6779 |
| 2          | 21      | 9.8479e+06 | 1.3618e+05 | 8.2362  | 10   | 9.5791e+06 | 2.4074e+05 | 12.7768 |
| 3          | 14      | 1.7174e+07 | 5.3622e+04 | 8.6949  | 7    | 1.5259e+07 | 3.4239e+05 | 14.7087 |
| 4          | 46      | 3.2687e+07 | 4.5734e+05 | 5.6407  | 31   | 3.0042e+07 | 4.9030e+05 | 1.0074  |
| 5          | 32      | 3.0151e+07 | 3.2799e+05 | 6.4208  | 21   | 2.7767e+07 | 4.9991e+05 | 8.3333  |
| 6          | 53      | 3.6208e+07 | 2.0919e+05 | 7.8493  | 30   | 3.3566e+07 | 2.2374e+05 | 11.2445 |
| 7          | 30      | 4.1996e+07 | 3.4454e+05 | 10.0000 | 21   | 4.4479e+07 | 4.2519e+05 | 14.1046 |
| 8          | 22      | 4.1207e+07 | 5.2957e+05 | 14.0116 | 50   | 4.4138e+07 | 3.7420e+05 | 8.2146  |
| 9          | 30      | 6.8674e+07 | 1.2349e+06 | 13.3541 | 30   | 6.9246e+07 | 5.4240e+05 | 13.4551 |
| 10         | 31      | 6.0367e+07 | 3.3987e+05 | 11.1429 | 30   | 6.2572e+07 | 3.1831e+05 | 12.3180 |
| 11         | 30      | 5.5957e+07 | 2.8571e+05 | 11.3043 | 50   | 5.7692e+07 | 4.6396e+05 | 7.0743  |
| 12         | 20      | 5.0883e+07 | 4.2106e+05 | 14.2857 | 30   | 5.0172e+07 | 2.6474e+05 | 14.9647 |



**Fig. 3** S/N ratios of the algorithms on test problems 3, 8, 10 and 12

in Table 3 is used for both NSGA-II and NPGA algorithms. Since, 12 test problems are conducted and L9 design is used, 216 problems have been solved and a large amount of data is obtained as results.

## Results and discussion

Based on the introduced measures in “Multi-objective performance measures”, the performance of the proposed algorithms, NSGA-II and NPGA, are evaluated. Thus, four performance measures are used to compare NSGA-II and NPGA. Under different scenarios of the parameter combinations based on L9 design, first, normalization is done based on the best solution of each measure. This procedure is done using expression (20). Where, for obtaining  $R$  the results of each measure is considered separately and *Best Sol* presents the desirable value which for different measures due to its concept it differs. The proper values for the *Best Sol* of NPS and DM are the higher value of each measure and conversely, for MID the lower value is considered as the *Best Sol*. Then the value of  $R$  is calculated for each run. Then, by expression (21) they are summed together, which  $w_i$  represents the weight of each measure based on decision maker’s preference and  $P_i$  presents the normalized values of the measures. Here the weights are considered 1, 2 and 2 for NPS, MID and DM measures, respectively. The solution with the lower combined value,  $W$ , of the normalized objective function values is selected for the best solution from the L9 design of each test problem. Table 4 shows the results of these metrics obtained by NSGA-II and NPGA algorithms using the 12 test problems.

For both algorithms, the  $S/N$  ratio is obtained by Minitab Software and the results of problem tests 3, 8, 10 and 12 are being shown in Fig. (3). As it is shown in Fig. (3), the optimal solution is obtained in different levels of NSGA-II and NPGA algorithm’s parameters. For instance, in test problem 3, first row of Fig. (3), the optimal results of NSGA-II is obtained when the number of iterations ( $Max_{It}$ ), number of population ( $N_{pop}$ ), crossover rate ( $P_c$ ) and mutation rate ( $P_m$ ) are considered on levels 3, 3, 1 and 3, respectively, while in NPGA the levels are 2, 1, 2 and 3, respectively.

$$R = \frac{|Method\ Sol - Best\ Sol| \times 100}{|Best\ Sol|} \quad (20)$$

$$W = \sum_{i=1}^I w_i P_i \quad (21)$$

As presented in Table 5, since the average of the obtained combined value of three performance measures,  $W$ , for NSGA-II is less than NPGA, NSGA-II is more

preferable than NPGA. Besides, the CPU time of running the algorithms can be important for decision makers.

In order to compare the combined value of three performance measures,  $W$ , and the CPU time, a popular multi-criteria decision-making method named TOPSIS (Triantaphyllou (2000)) can be used. The considered method ranks the solutions of the problem, which includes conflicting criteria and is helpful for the decision makers to finally decide and make a decision. Table 6 illustrates the decision matrix related to the test problems. As two algorithms and two performance measures are considered, the decision matrix consists of two alternatives and two criteria.

For employing the TOPSIS method, the decision maker is able to consider the value of the weights according to the criterion’s importance. Based on the criterion’s importance three modes can occur; (1)  $W$  and CPU time have the same importance so their weights are equal, (2)  $W$  is more important for the decision maker so its weight is greater than CPU time, (3) CPU time has more importance than  $W$  so a greater weight is considered for it.

The decision problems are divided into three categories; operational, tactical and strategic problems. As the optimal solution and CPU time, both are important but in the three categories mentioned above their importance are somehow different. The optimal solution is more important

**Table 5** Algorithms’ comparison results

| No. of ex. | NSGA-II |            | NPGA    |            |
|------------|---------|------------|---------|------------|
|            | $W$     | CPU        | $W$     | CPU        |
| 1          | 7.4584  | 364.67913  | 11.6779 | 308.09,585 |
| 2          | 8.2362  | 428.56486  | 12.7768 | 349.28851  |
| 3          | 8.6949  | 496.56231  | 14.7087 | 246.18175  |
| 4          | 5.6407  | 719.27633  | 1.0074  | 808.78046  |
| 5          | 6.4208  | 441.26564  | 8.3333  | 249.89928  |
| 6          | 7.8493  | 2184.82376 | 11.2445 | 812.08685  |
| 7          | 10.0000 | 810.18686  | 14.1046 | 830.15926  |
| 8          | 14.0116 | 952.86671  | 8.2146  | 2175.73618 |
| 9          | 13.3541 | 1352.94741 | 13.4551 | 1437.83610 |
| 10         | 11.1429 | 1574.92751 | 12.3180 | 1924.74610 |
| 11         | 11.3043 | 1829.00736 | 7.0743  | 2013.73645 |
| 12         | 14.2857 | 2197.28008 | 14.9647 | 1928.15139 |
| Average    | 9.87    | 1112.70    | 10.82   | 1090.39    |

**Table 6** Decision matrix for the test problems

|         | $W$   | CPU     |
|---------|-------|---------|
| NSGA-II | 9.87  | 1112.70 |
| NPGA    | 10.82 | 1090.39 |

**Table 7** The results of ranking using TOPSIS method

|         | Weight of $W$ | Weight of CPU | $C_i^*$ | Ranking algorithms |
|---------|---------------|---------------|---------|--------------------|
| NSGA-II | 0.5           | 0.5           | 0.8182  | 1                  |
| NRGA    |               |               | 0.1818  | 2                  |
| NSGA-II | 0.7           | 0.3           | 0.9137  | 1                  |
| NRGA    |               |               | 0.0863  | 2                  |
| NSGA-II | 0.3           | 0.7           | 0.6599  | 1                  |
| NRGA    |               |               | 0.3401  | 2                  |

\* Shows the relative closeness to the ideal solution

than CPU time in tactical and strategic problems so it can get a greater weight. However, as the CPU time is more important in operational problems, the considered weight is greater.

Since Production and Location problems are considered as tactical and strategic problems and the proposed model is a Production and Location problem, the optimal solution has more importance. For this reason, the second mode is

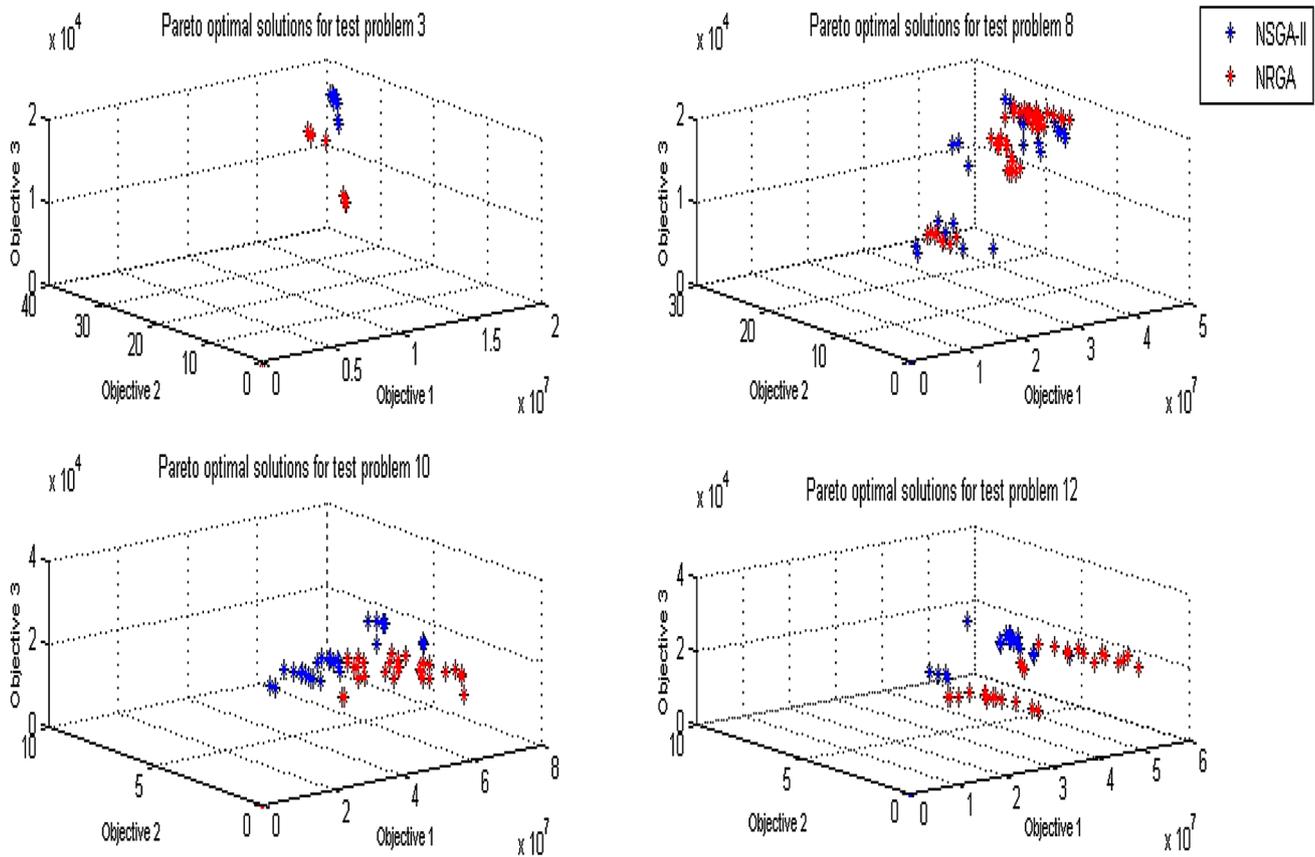
applied which the weights of  $W$  and CPU are considered 0.7 and 0.3, respectively. As it can be seen in the second row of Table 6, NSGA-II showed better results compared with NRGA. However, the other two modes are considered and the results are obtained. Table 6 illustrates the results of ranking using TOPSIS method for the three modes mentioned above.

Based on the results presented in Table 7, NSGA-II showed better results compared with NRGA.

For further clarifications of the Pareto based algorithms, NSGA-II and NRGA, the Pareto optimal solutions are shown for examples 3, 8, 10 and 12 of the test problems in Fig. 4.

### Conclusions

This article designs a multi-period multi-echelon multi-product supply chain model. As the aim of the article is considering the profit of the chain due to new product development, the proposed model consists of three objectives which are the maximization of total profit, maximization of the satisfaction level of customer demands and



**Fig. 4** Pareto fronts of the algorithms on test problems 3, 8, 10 and 12

maximization of new product production. Besides the model objectives, the goal is to determine the best and efficient time for introducing new products or changing and developing the current products.

In order to tackle the complex model, the model is solved using two Pareto-based multi-objective evolutionary algorithms known as NSGA-II and NPGA. Next, 12 test problems of different sizes are considered and solved by the proposed algorithms. The parameters of the algorithms were tuned using the Taguchi method. After the algorithms were compared in terms of five performance metrics, the TOPSIS method was implemented to compare the algorithms in terms of multi-objectives metrics. Based on the results, NSGA-II as a multi-objective Pareto-based optimization algorithm, showed better results compared with NPGA.

**Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

## References

- Al Jadaan O, Rajamani L, Rao CR (2008) Non-dominated ranked genetic algorithm for solving multi-objective optimisation problems: NPGA. *J Theor Appl Inf Technol* 2:60–67
- Alizadeh Afrouzy Z, Nasserli SH, Mahdavi I (2016) A genetic algorithm for supply chain configuration with new product development. *Comput Ind Eng* 101:440–454
- Altıparmak F, Gen M, Lin L, Karaoglan I (2009) A steady-state genetic algorithm for multi-product supply chain network design. *Comput Ind Eng* 56(2):521–537
- Amini M, Li H (2011) Supply chain configuration for diffusion of new products: an integrated optimization approach. *Omega* 39:313–322
- Bandyopadhyay S, Bhattacharya R (2013) Solving multi-objective parallel machine scheduling problem by a modified NSGA-II. *Appl Math Model* 37(10–11):6718–6729
- Bandyopadhyay S, Bhattacharya R (2014) Solving a tri-objective supply chain problem with modified NSGA-II algorithm. *J Manuf Syst* 33(1):41–50
- Billington C, Lee H, Tang C (1998) Successful strategies for product rollovers. *Sloan Manag Rev* 39:23–30
- Chen CL, Lee WC (2004) Multi-objective optimization of multi-echelon supply chain networks with uncertain product demands and prices. *Comput Chem Eng* 28:1131–1144
- Chen CL, Wang BW, Lee WC (2003) Multi-objective optimization for a multi-enterprise supply chain network. *Ind Eng Chem Res* 42:1879–1889
- Coello CAC, Lamont GB, Van Veldhuizen DA (2002) *Evolutionary algorithms for solving multi-objective problems*. Springer, New York
- Deb K (2001) *Multi-objective optimisation using evolutionary algorithms*. Wiley, New York
- Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput* 6(2):182–197
- Diabat A (2014) Hybrid algorithm for a vendor managed inventory system in a two echelon supply chain. *Eur J Op Res* 238(1):114–121
- Gen M, Cheng R (2000) *Genetic algorithms and engineering optimization*, Wiley, New York
- Ghodratnama A, Tavakkoli-Moghaddam R, Azaron A (2015) Robust and fuzzy goal programming optimization approaches for a novel multi-objective hub location-allocation problem: a supply chain overview. *Appl Soft Comput* 37:255–276
- Gholamian N, Mahdavi I, Tavakkoli-Moghaddam R, Mahdavi-Amiri N (2015) Comprehensive fuzzy multi-objective multi-product multi-site aggregate production planning decisions in a supply chain under uncertainty. *App Soft Comput* 37:585–607
- Godichaud M, Amodeo L (2015) Efficient multi-objective optimization of supply chain with returned products. *J Manuf Syst* 37:683–691
- Govindan K, Jafarian A, Nourbakhsh V (2015) Bi-objective integrating sustainable order allocation and sustainable supply chain network strategic design with stochastic demand using a novel robust hybrid multi-objective metaheuristic. *Comput Op Res* 62:112–130
- Guillén G, Melea FD, Bagajewicz MJ, Espuña A, Puigjanera L (2004) Multi objective supply chain design under uncertainty. *Chem Eng Sci* 60:1535–1553
- Hung S-J (2011) Activity-based divergent supply chain planning for competitive advantage in the risky global environment: a DEMATEL-ANP fuzzy goal programming approach. *Expert Syst Appl* 38(8):9053–9062
- Jadidia O, Cavalierib S, Zolfaghari S (2015) An improved multi-choice goal programming approach for supplier selection problems. *Appl Math Model* 39(14):4213–4222
- Jafarian M, Bashiri M (2014) Supply chain dynamic configuration as a result of new product development. *Appl Math Model* 38(3):1133–1146
- Kao H-Y, Chan C-Y, Wu D-J (2014) A multi-objective programming method for solving network DEA. *Appl Soft Comput* 24:406–413
- Kotinis M (2014) Improving a multi-objective differential evolution optimizer using fuzzy adaptation and K-medoids clustering. *Soft Comput* 18(4):757–771
- Latha Shankar B, Basavarajappa S, Kadadevaramath RS, Chen JCH (2013) A bi-objective optimization of supply chain design and distribution operations using non-dominated sorting algorithm: a case study. *Expert Syst Appl* 40:5730–5739
- Li H, Amini M (2012) A hybrid optimisation approach to configure a supply chain for new product diffusion: a case study of multiple-sourcing strategy. *Int J Prod Res* 50(11):3152–3171
- Mohapatra P, Nayak A, Kumar SK, Tiwari MK (2015) Multi-objective process planning and scheduling using controlled elitist non-dominated sorting genetic algorithm. *Int J Prod Res* 53(6):1712–1735
- Murillo-Alvarado PE, Guillén-Gosálbez G, Ponce-Ortega JM, Castro-Montoya AJ, Serna-González M, Jiménez L (2015) Multi-objective optimization of the supply chain of biofuels from residues of the tequila industry in Mexico. *J Clean Prod* 108:422–441
- Nager L, Dutta P, Jain K (2014) An integrated supply chain model for new products with imprecise production and supply under scenario dependent fuzzy random demand. *Int J Syst Sci* 45(5):873–887
- Narahariseti P, Karimi IA (2010) Supply chain redesign and new process introduction in multipurpose plants. *Chem Eng Sci* 65:2596–2607
- Nepal B, Monplaisir L, Famuyiwa O (2011) A multi-objective supply chain configuration model for new products. *Int J Prod Res* 49(23):7107–7134



- Niknam T, Meymand Hamed Z, Mojarrad Hasan DA (2011) Practical multi objective PSO algorithm for optimal operation management of distribution network with regard to fuel cell power plants. *Renew Energy* 36(5):1529–1544
- Pasandideha SHR, Akhavan Niakib ST, Asadi K (2015) Optimizing a bi-objective multi-product multi-period three echelon supply chain network with warehouse reliability. *Expert Syst Appl* 42(5):2615–2623
- Sadeghi J, Sadeghi S, AkhavanNiaki ST (2014) A hybrid vendor managed inventory and redundancy allocation optimization problem in supply chain management: an NSGA-II with tuned parameters. *Comput Op Res* 41:53–64
- Safaei Arshia S, Zolfagharib A, Mirvakili SM (2014) A multi-objective shuffled frog leaping algorithm for in-core fuel management optimization. *Comput Phys Commun* 185(10):2622–2628
- Shang RH, Jiao LC, Liu F, Ma WP (2012) A novel immune clonal algorithm for MO problems. *IEEE Trans Evol Comput* 16(1):35–50
- Shin KS, Park J-O, Kim YK (2011) Multi-objective FMS process planning with various flexibilities using a symbiotic evolutionary algorithm. *Comput Op Res* 38(3):702–712
- Simchi-Levi D, Kaminsky P, Simchi-Levi E (2000) *Designing and managing the supply chain: concepts, strategies, and case studies*. McGraw-Hill, New York
- 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(14):3990–4012
- Srinivas N, Deb K (1995) Multi objective optimization using non dominated sorting in genetic algorithm. *Evol Comput* 2:221–248
- Taguchi G (1986) *Introduction to quality engineering*. Asian Productivity Organisation/UNIPUB, White Plains
- Taher N, Ehsan AF, Majid N (2010) An efficient multi-objective modified shuffled frog leaping algorithm for distribution feeder configuration problem. *Eur Trans Electr Power* 21(1):721–739
- Talbi E-G (2009) *Metaheuristics: from design to implementation*. Wiley, New York
- Tana CJ, Limb CP, Cheaha Y-N (2014) A multi-objective evolutionary algorithm-based ensemble optimizer for feature selection and classification with neural network models. *Neurocomputing* 125:217–228
- Triantaphyllou E (2000) *Multi-criteria decision making methods: a comparative study*. Kluwer Academic Publishers, Dordrecht
- Tsaia S-J, Suna T-Y, Liub C-C, Hsiehc S-T, Wua W-C, Chiu S-Y (2010) An improved multi-objective particle swarm optimizer for multi-objective problems. *Expert Syst Appl* 37(8):5872–5886
- Validi S, Bhattacharya A, Byrne PJ (2014) A case analysis of a sustainable food supply chain distribution system—a multi-objective approach. *Int J Prod Econ* 152:71–87
- Van Veldhuizen DA (1999) *Multiobjective evolutionary algorithms: classifications, analyses, and new innovations*. Air Force Institute of Technology, Wright Patterson AFB, OH
- Wang L, Zhong X, Liu M (2012) A novel group search optimizer for multi-objective optimization. *Expert Syst Appl* 39(3):2939–2946
- Zhang Y, Gong DW, Gong N (2013) Multi-objective optimization problems using cooperative evolution particle swarm optimizer. *J Comput Theor Nanosci* 10(3):655–663
- Zhu YS, Wang J, Qu BY (2014) Multi-objective economic emission dispatch considering wind power using evolutionary algorithm based on decomposition. *Int J Electr Power Energy Syst* 63:434–445

