Scientia Iranica E (2015) 22(3), 1227–1241

SCIENTIA IRANICA

Sharif University of Technology
Scientia Iranica
Transactions E: Industrial Engineering
www.scientiairanica.com

A hybrid cultural-harmony algorithm for multi-objective supply chain coordination

S. Alaei and F. Khoshalhan*

Department of Industrial Engineering, KNT University of Technology, Tehran, Iran.

Received 2 October 2013; received in revised form 20 April 14; accepted 17 November 2014

KEYWORDS
Supply chain coordination;
Meta-heuristics;
Taguchi method;
Supplier selection;
Cultural algorithm;
Harmony search.

Abstract. We investigate a one-buyer, multi-vendor, coordination model with a vendor selection problem in a centralized supply chain. In the proposed model, the buyer selects one or more vendor and orders an appropriate quantity. The quantity discount mechanism is used by all vendors with the aim of coordinating the supply chain. We formulate the problem as a multi-objective mixed integer nonlinear mathematical model. Using the Global Criterion method, the proposed model is transformed into a single objective optimization problem. Since the problem is NP-hard, we propose four meta-heuristic algorithms: Particle Swarm Optimization (PSO), Scatter Search (SS), population based Harmony Search (HS-pop) and Harmony Search based Cultural Algorithm (HS-CA). The Taguchi robust tuning method is applied in order to estimate the optimum values of parameters. Then, the solution quality and computational time of the algorithms are compared.

1. Introduction

Coordination between all entities in a supply chain and global planning is necessary to achieve effective Supply Chain Management (SCM) [1]. In the lack of cooperation, supply chain members are willing to optimize their own objectives independently, which may lead to channel inefficiency. Designing mechanisms for coordinating and aligning decisions between entities is of great importance in supply chain management. Several coordination mechanisms have been applied in the literature, such as the wholesale price contract [2], two-part tariff [3], buyback [4], revenue sharing [5], quantity flexibility [6], back-up [7], sales rebate [8], quantity discount [9], timing discount [10], and the revenue sharing reservation contract with penalty [11].

Supplier (vendor) selection is a strategic decision when a buyer tries to establish a win-win business relationship with its supplier. It is one of the most critical components of the purchasing function of a firm [12]. Vendor selection and order allocation are two main features to be considered in supply chain management.

Suppose a typical channel with a single buyer and multiple vendors. The buyer faces a four-objective constrained problem, i.e. selecting one or more vendor(s) in order to allocate his order quantity for satisfying market demand. All the vendors offer quantity discounts to motivate buyers to order more quantities. The coordination of this supply chain is studied in context to examine the performance of different metaheuristic algorithms.

Some models have been developed to investigate the coordination problem and the vendor selection problem. However, little attention has been paid to developing efficient algorithms in this area. In this paper, we consider four metaheuristic algorithms.

*. *Corresponding author. Tel.: +98 21 84063347;*
E-mail addresses: alaei.saeed@gmail.com (S. Alaei); khoshalhan@kntu.ac.ir (F. Khoshalhan)

The population based Harmony Search (HS-pop) and Harmony Search based Cultural Algorithm (HS-CA) are two hybrid algorithms that are proposed for comparison with already developed algorithms in this area (i.e. Particle Swarm Optimization (PSO) and Scatter Search (SS)).

The main contributions of the article are:

1. The Harmony Search (HS) algorithm and the Cultural Algorithm are incorporated (CA). We assume that the situational knowledge component of the CA belief space acts as a harmony memory. Moreover, offspring are generated using HS operators and CA belief space.
2. In the HS-pop algorithm, the reproduction process and updating the harmony memory are modified compared to the traditional HS algorithm.

The objectives of the research are:

1. To investigate multi-objective coordination of a supply chain using a quantity discount contract;
2. To compare the performance of two hybrid algorithms with other metaheuristic algorithms. Using numerical study, we perform analyses over solution qualities and computational effort.

We apply the Taguchi robust tuning method in order to estimate the optimum values of parameters. The Relative Percentage Deviation (RPD) is used to assess the quality of solutions. Moreover, in order to evaluate the computational time of the proposed algorithms, the time to reach a solution with r% error (i.e. 1% or 3%) is computed. We investigate the Convergence Index (CI) of the proposed algorithms, as the number of successful runs with which the algorithm reaches a solution, with r% error, over the total number of runs.

The remainder of the paper is organized as follows: Section 2 briefly reviews the related literature. Section 3 describes the problem and notation. Section 4 presents the solution procedure and four metaheuristic algorithms. In Section 5, the Taguchi method is used for tuning parameters of the algorithms. Section 6 provides an illustrative example. The proposed algorithms are evaluated using the numerical example in Section 7. Finally, Section 8 summarizes and concludes the paper.

2. Literature review

We provide a brief review of supply chain coordination and supplier selection models that have been studied in recent years. Rosenthal et al. [13] studied a supplier selection problem with multiple products, where suppliers offer discounts when a "bundle" of products is bought. Sarkis and Semple [14] discussed a single period supplier selection problem with business volume discounts, wherein the total purchasing cost should be optimized without taking inventory-related costs into account. Goossens et al. [15] studied a multi items supplier selection, wherein the suppliers offer an all-unit quantity discount and try to minimize the total cost of purchasing. Ghodsypour and O'Brien [16] developed an integrated AHP and linear programming model, in which both qualitative and quantitative factors are considered in the process of supplier selection and order allocation. Amid et al. [17] proposed a weighted additive fuzzy multi-objective model for the supplier selection problem under all-unit price discounts.

Jayaraman et al. [18] developed a Mixed Integer linear Programming (MIP) model, wherein quality production capacity, lead-time, and storage capacity limits are considered. Another MIP model for the supplier selection is proposed by Cakravastia et al. [19] where the objective is to minimize the level of customer dissatisfaction, which was evaluated by price and lead time. Dahel [20] developed a Multi-Objective Mixed Integer Programming (MO-MIP) model with multiple products and discounts on total business volume. Xia and Wu [21] presented a MO-MIP approach under a total business volume discount, and used an integrated AHP method and multi-objective programming to investigate the problem. Ebrahim et al. [22] developed a MO-MIP model with different types of discount and proposed a scatter search algorithm to solve the problem.

Herer et al. [23] were the first to propose a supplier selection problem together with coordination models. The limited annual production rate and inventory holding costs are taken into account in their model. Kim and Goyal [24] investigated two different shipment policies from the suppliers to the buyer in which suppliers deliver their production lots either simultaneously or successively. Kamali et al. [25] developed a multi-objective mixed integer nonlinear programming model to coordinate the system of a single-buyer- multi-vendors, under an all-unit quantity discount. They proposed Particle Swarm Optimization and Scatter Search for solving the problem. Gheidar-Kjeljani [26] proposed a nonlinear mathematical model which is a combination of a supplier selection model and a coordination model in a centralized supply chain. In their model, the buyer ordered quantities are split into small lot sizes and are delivered to the buyer over multiple periods.

3. Problem description

A typical channel with one buyer and multiple vendors is considered. The buyer selects one or more vendors in order to allocate his order quantity for satisfying market demand without any type of shortage. All the

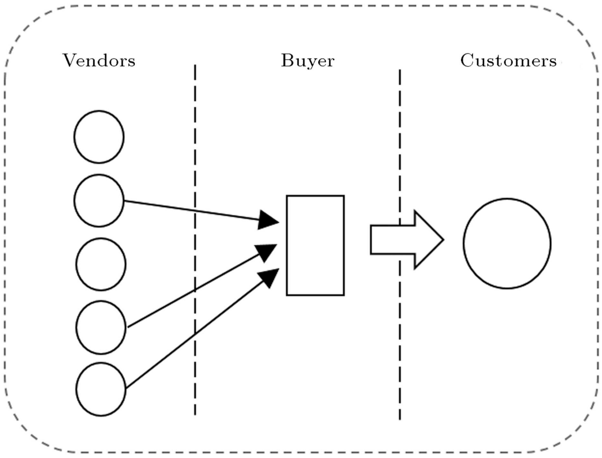


Figure 1. Supply chain members.

vendors offer quantity discounts to motivate buyers to order more quantities. A schematic view of the supply chain is illustrated in Figure 1. Selecting suitable vendors for supplying the products is an important part of the operation.

The buyer has to choose one or more vendor(s) and purchase an optimal level of single product from each vendor based on various objectives. The buyer has a dilemma, due to discount contracts offered by the vendors, which depend on the volume of the order quantity. The objectives considered in this paper for choosing potential vendors are similar to those of Kamali et al. [25]; they are:

a) Minimizing the whole supply chain annual costs;
b) Minimizing the total defective items ordered by the buyer;
c) Minimizing the total late delivered items;
d) Maximizing the total annual purchasing value.

Suppose that there are n vendors in the supply chain. Each of them (i.e., vendor i) offers an all-unit quantity discount with K_i price level, each level (i.e. level k) is characterized by an interval, $[u_{i,k-1}, u_{ik})$, and the price of c_{ik} is associated with this interval. For example, for each vendor, i, we have $u_{i1} < u_{i2} < \cdots < u_{i,K_i}$ and $c_{i1} > c_{i2} > \cdots > c_{i,K_i}$. Also, assume that vendor order quantities are dispatched to the buyer in a sequential order. In other words, after consuming the products of one vendor, the products of another vendor can be entered.

Parameters used in the problem:

D Buyer annual fixed market demand rate;
S_i Fixed setup cost associated with vendor i;
A_i Buyer fixed ordering cost for vendor i;
h_i Vendor i's inventory holding cost per unit, per unit time;
h Retailer's fixed inventory holding cost per unit, per unit time (independent of purchasing cost);
T_i Consumption time of an order quantity from vendor i;
T Cycle time of the retailer;
z_i Unit variable cost of vendor i;
P_i Production rate associated with vendor i;
b_{ik} Quantity at which the kth price break occurs by vendor i;
R_i Reliability of time of delivery of products for vendor i;
d_i Defective rate that vendor i maintains;
w_i Total weight assigned to vendor i.

Decision variables used in the model:

q_{ik} Number of units supplied by vendor i at price level k in a cycle;
y_{ik} Binary variable denoting whether order quantity is chosen from k's price level or not;
Q_i Order quantity supplied by vendor i in a cycle, i.e. $Q_i = \sum_{k=1}^{K_i} q_{ik}$;
Q Total order quantity supplied by all vendors in a cycle, i.e. $Q = \sum_{i=1}^{n} Q_i$.

The problem consists of four objectives:

a) **Cost:** To minimize annual costs of the whole supply chain:

$$Z_1 = \frac{D}{Q}\sum_{i=1}^{n}\sum_{k=1}^{K_i}(z_i + c_{ik})q_{ik} + \frac{D}{Q}\sum_{i=1}^{n}\sum_{k=1}^{K_i}(A_i + S_i)y_{ik} + \frac{D}{Q}\sum_{i=1}^{n}\left[\frac{1}{2}\left(\frac{h}{D} + \frac{h_i}{P_i}\right)\left(\sum_{k=1}^{K_i} q_{ik}\right)^2\right]. \quad (1)$$

The objective function, as shown in Eq. (1), consists of three parts: The first part includes variable and purchasing costs; The second part consists of ordering and setup costs; and the third part includes buyer and vendor holding costs.

b) **Quality:** To minimize the total defective items ordered by the buyer:

$$Z_2 = \frac{D}{Q}\sum_{i=1}^{n}\sum_{k=1}^{K_i} d_i q_{ik}. \quad (2)$$

c) **Delivery reliability:** To minimize the total late delivered items:

$$Z_3 = \frac{D}{Q} \sum_{i=1}^{n} \sum_{k=1}^{K_i} (1 - R_i) q_{ik}, \tag{3}$$

where $(1 - R_i)$ denotes the late delivery rate of products for vendor i.

d) **Purchasing value:** To maximize the total annual purchasing value:

$$Z_4 = \frac{D}{Q} \sum_{i=1}^{n} w_i \sum_{k=1}^{K_i} q_{ik}, \tag{4}$$

where w_i captures the overall performance of vendor i, and can be calculated by multiple criteria decision making methods; and $\sum_{k=1}^{K_i} q_{ik}$ is the total amount of products to be ordered from vendor i. Note that the vendor with the highest weight has a higher priority for purchasing. In a single objective framework, the decision maker (buyer) tends to purchase all his order quantities from the vendor with the highest weight. So, maximizing Eq. (4) ensures that the greater part of the buyer orders are allocated to vendors with higher performance weights and higher priorities.

Under the Global Criterion method, the relative weighted distance between each objective function's value and its reference point is minimized. The reference point for each objective m (Z_m^*), is obtained by optimizing the mth objective and neglecting other objectives subject to the problem constraints. Suppose that W_m is the weight of objective m that can be achieved by decision maker preferences. So, the problem can be rewritten as the following single objective optimization problem:

$$\min Z = \sum_{m=1}^{4} W_m \frac{|Z_m - Z_m^*|}{Z_m^*}. \tag{5}$$

Moreover, the problem has the following constraints:

- Capacity constraint: Each vendor, i, has maximum capacity,

$$\frac{D}{Q} \sum_{k=1}^{K_i} q_{ik} \leq P_i \qquad \forall i = 1 \cdots n. \tag{6}$$

- Demand constraint: The demand of the buyer has to be satisfied,

$$\sum_{i=1}^{n} \sum_{k=1}^{K_i} q_{ik} = Q. \tag{7}$$

- Discount constraints: The following constraints ensure that if vendor i is chosen, the amount of order quantity should fall into discount interval $[u_{i,k-1}, u_{ik})$:

$$\sum_{k=1}^{K_i} y_{ik} \leq 1 \qquad \forall i = 1 \cdots n, \tag{8}$$

$$u_{i,k-1} y_{ik} \leq q_{ik} \leq u_{ik} y_{ik}$$
$$\forall i = 1 \cdots n; \qquad \forall k = 1 \cdots K_i. \tag{9}$$

4. Solution procedure

For the problem studied in this paper, we investigate four algorithms. As mentioned before, they are PSO, SS, HS-pop and HS-CA. Here, each solution vector is demonstrated as $Q = [Q_1, Q_2, \cdots, Q_n]$, where Q_i is the order quantity assigned for vendor i, and is equal to $Q_i = [q_{i1}, q_{i2}, \cdots, q_{iK_i}]$. In all algorithms studied here, we apply a similar algorithm for generating initial solutions. In this algorithm, each vendor is selected with a probability of 0.5. Then, for each selected vendor, i, an order quantity is randomly assigned between $[0, u_{ik})$.

4.1. Repair algorithm

In order to avoid infeasible solutions caused by capacity constraints, we apply repairing strategies in all algorithms. A repair procedure transforms an infeasible solution into a feasible one. For the problem studied in this paper, if each vendor annual capacity constraints are violated by assigned annual order quantities, then, the extra amount of their order quantities is assigned to other vendors by a rule, as described below.

For any infeasible solution, we define $a_i = P_i - DQ_i/Q$ for each vendor, i, in which $a_i \geq 0$ points out that vendor i still has some capacity for assigning the order, and $a_i < 0$ indicates that the annual order quantity allocated to vendor i exceeds its annual production capacity. Then, we define two subset of vendors as $S^+ = \{i : a_i \geq 0\}$, which captures vendors that still have some capacity, and $S^- = \{i : a_i < 0\}$, which demonstrates vendors with violated capacity constraints. So, the following changes ensure the feasibility of the solution until we have $\sum_{i \in S^+} a_i \geq \sum_{i \in S^-} a_i$, otherwise, the solution should be rejected:

$$Q_i = \frac{Q}{D} P_i \qquad \forall i \in S^-, \tag{10}$$

$$Q_i = Q_i + \left(\frac{a_i}{\sum_{i \in S^+} a_i} \right) \sum_{i \in S^-} a_i \qquad \forall i \in S^+. \tag{11}$$

Eq. (10) sets the annual order quantity of already-capacity-violated vendors to be equal to their annual production capacity. Eq. (11) shares the extra quantity ($\sum_{i\in S^-} a_i$) among the other vendors proportional to their remainder capacity.

4.2. Particle swarm optimization

Particle swarm optimization is a population based metaheuristic inspired from swarm intelligence [27]. PSO has been successfully applied for continuous optimization problems [28]. A swarm of N particles flies around the search space. Each particle position is determined by its velocity and previous position. The step and direction of each particle, i, toward the global optimum is affected by two factors: $P\text{best}_i$, which is the best position visited by itself; and Gbest, which is the best position visited by all particles. The velocity of each particle is updated as follows:

$$v_i^{t+1} = w \times v_i^t + c_1 \times r_1 \times (P\text{best}_i^t - x_i^t) + c_2 \times r_2 \times (G\text{best}^t - x_i^t), \tag{12}$$

where $r_1, r_2 \in [0,1]$ are two random numbers; c_1 and c_2 are constant and denote the learning factors; t is the iteration number; and w is the inertia weight which controls the effect of previous velocity on the current one. We use a dynamic approach for the inertia weight, as:

$$w(t) = w_{\max} - \frac{w_{\max} - w_{\min}}{N} \times t, \tag{13}$$

where $w_{\min}$ and $w_{\max}$ are the minimum and maximum inertia weights, respectively, and N is the maximum number of iterations. Then, each particle position is updated with:

$$x_i^{t+1} = x_i^t + v_i^{t+1}. \tag{14}$$

At each iteration, the values of $P\text{best}_i$ and Gbest are updated if better solutions are obtained by particle i and all particles, respectively. Algorithm 1 presents the template for PSO.

4.3. Scatter search

The scatter search was firstly introduced by Glover [29] and is an evolutionary metaheuristic which recombines selected solutions from a reference set to build others [30]. There are five basic methods in the scatter search [31]:

```
Begin
Initialize the problem and algorithm parameters (t = 0);
Initialize the population space, Pop^0;
    Repeat
        For particle i in the swarm do
        Update particle i's velocity and position
        Update Pbesti and Gbest
        End for
    Until stopping criteria
End
```

Algorithm 1. Particle Swarm Optimization (PSO) pseudo code.

1. A *Diversification generation method* generates a set of diverse trial solutions. This method aims to diversify the search while selecting high-quality solutions.
2. An *Improvement method* transforms a trial solution into one or more enhanced trial solutions using any S-metaheuristic. Usually, a local search algorithm is applied and then a local optimum is generated.
3. In the *Reference set update method*, the aim is to guarantee diversity while keeping high-quality solutions. This method builds and maintains a reference set (with b individuals) that consists of two subsets: Ref-Set1 (with b_1 individuals), with the best fitness function, and Ref-Set2 (with b_2 individuals where $b_2 = b - b_1$), with the best diversity.
4. A *Subset Generation Method* operates on the reference set, and produces a subset of solutions as a basis for creating combined solutions. In this method, all the subsets of a fixed size, r (generally, $r = 2$), are selected.
5. The *Solution Combination Method* is a given subset of solutions produced by the Subset Generation Method transformed into one or more combined solution vectors.

Algorithm 2 presents the template for SS.

4.4. Proposed algorithms

4.4.1. Brief review of harmony search

Harmony Search (HS), a relatively new metaheuristic optimization algorithm, was introduced by Geem et al. [32]. It imitates musician behavior, where the instrument pitch is improvised upon by searching for a perfect state of harmony. According to the harmony search algorithm, the Harmony Memory (HM) is a matrix of individuals with a size of HMS:

$$\text{HM} = \begin{bmatrix} Q_1 \\ \vdots \\ Q_{\text{HMS}} \end{bmatrix} = \begin{bmatrix} q_{11} & \cdots & q_{n1} \\ \vdots & \ddots & \vdots \\ q_{1,\text{HMS}} & \cdots & q_{n,\text{HMS}} \end{bmatrix}, \tag{15}$$

```
Begin
Initialize the problem and algorithm parameters (t = 0);
Initialize the population space, Pop^0, using a
diversification generation method;
Improvise the population, Pop^0, by an improvement
method
Update the reference set, Refset^0;
    Repeat
    Apply the subset generation method;
        For each couple of solutions do
        Apply the Solution Combination Method;
        Improvise the solutions by an Improvement
        Method;
        End
    Update the reference set, Refset^t;
    Until stopping criteria
End
```

Algorithm 2. Scatter Search (SS) pseudo code.

where we have $f(Q_1) \leq f(Q_2) \leq \cdots \leq f(Q_{\text{HMS}})$. In order to update HM, if a new generated solution is better than the worst vector, then, the worst vector is replaced by the new one.

A new harmony vector or individual is generated by considering harmony memory and randomness.

$$q_i' \leftarrow \begin{cases} q_i' \in \{q_{i1}, q_{i2}, \cdots, q_{i,\text{HMS}}\} & \text{w.p. HMCR} \\ q_i' \in A_i & \text{w.p. } (1-\text{HMCR}) \end{cases} \tag{16}$$

Pitch adjusting for:

$$q_i' \leftarrow \begin{cases} \text{Yes} & \text{w.p. PAR} \\ \text{No} & \text{w.p. } (1-\text{PAR}) \end{cases}$$

$$q_i' \leftarrow q_i' \pm \text{rand}() \times bw, \tag{17}$$

where HMCR $\in [0,1]$ is the Harmony Memory Considering Rate; PAR $\in [0,1]$ is the Pitch Adjustment Rate; and bw is the maximum distance bandwidth of changing q_i'. At each iteration:

1. Each new variable either inherits its value from the historical values stored in HM with a probability of HMCR, or is chosen according to its possible range, with a probability of (1-HMCR).
2. Then, each decision variable is examined as to whether or not to be changed around its value, with a probability of PAR.

In this paper, we use the improved version of Harmony Search (IHS) proposed by Mahdavi et al. [33]. Their proposed algorithm includes dynamic adaptation for both PAR and bw values. The PAR value is linearly increased, and the bw value is exponentially decreased in each iteration of the HS using the following equations, respectively:

$$\text{PAR}(t) = \text{PAR}_{\min} + \frac{\text{PAR}_{\max} - \text{PAR}_{\min}}{\text{NI}} \times t, \tag{18}$$

$$bw(t) = bw_{\max} \times \exp(c \times t);$$

$$c = \frac{Ln(bw_{\min}/bw_{\max})}{\text{NI}}, \tag{19}$$

where $\text{PAR}_{\min}$ and $\text{PAR}_{\max}$ are the minimum and maximum pitch adjusting rates, respectively, NI is the maximum number of iterations, t is the generation number; $bw_{\min}$ and $bw_{\max}$ are the minimum and maximum bandwidths, respectively.

4.4.2. Brief review of cultural algorithm

Cultural Algorithms (CAs) were introduced by Reynolds [34] and are special variants of evolutionary algorithms. They are inspired by the principle of cultural evolution. The CA framework consists of population space and belief space, which is used for forming, storing, and delivering knowledge experiences.

The belief space contains the five knowledge sources, i.e. the normative, situational, domain, topographical and history KS. Here, we apply two kinds of the most fundamental knowledge of the belief space: situational knowledge, S^t, and normative knowledge, N^t. That is, $\mathcal{B}^t = (S^t, N^t)$, where the situational knowledge, S^t, is the set of best individuals. For example, for K best individuals, the situational knowledge has the following structure:

$$S^t = \begin{bmatrix} x_1^1, x_2^1, \cdots, x_n^1, f(x)^1 \\ x_1^2, x_2^2, \cdots, x_n^2, f(x)^2 \\ \vdots \\ x_1^K, x_2^K, \cdots, x_n^K, f(x)^K \end{bmatrix}. \tag{20}$$

Moreover, the normative knowledge, N^t, is the set of interval information, together with the fitness for each extreme of the interval:

$$N^t = \left(\mathcal{X}_1^t, \mathcal{X}_2^t, \cdots, \mathcal{X}_n^t\right), \tag{21}$$

and for each domain variable, $\mathcal{X}_i^t$, the following information is stored:

$$\mathcal{X}_i^t = (l_i^t, u_i^t, L_i^t, U_i^t), \qquad i = 1 \cdots n, \tag{22}$$

where l_i^t and u_i^t, respectively, represent the lower and upper bounds of the closed interval for variable i, i.e. $l_i^t \leq q_i \leq u_i^t$; L_i^t and U_i^t are the performance scores of the individual for the lower and upper bounds, respectively.

There are two main phases in the CA: the influence phase and the acceptance phase. The influence function determines which knowledge source influences individuals. The original CA used the roulette wheel selection, based on knowledge source performance in previous generations [35]. The acceptance function decides which individuals and their properties can affect the belief space [36]. For example, a percentage of the best individuals (e.g. top 10%) can be accepted [36].

With the acceptance function and influence function on hand, the belief space is updated at each generation. Updating the situational knowledge can be done by any selection approach. For example, one can update situational knowledge by the k top individuals in the population, or use the Tournament selection.

In addition, the normative knowledge of belief space is updated at each generation. The lower and upper boundaries of decision variables and their responding fitness values are updated as follows. For each $Q_j^t, j = 1..n_{\mathcal{B}^t}$;

$$l_i^{t+1} = \begin{cases} q_{ij}^t & \text{if } q_{ij}^t \leq l_i^t \text{ or } f(Q_j^t) < L_i^t \\ l_i^t & \text{otherwise} \end{cases} \tag{23}$$

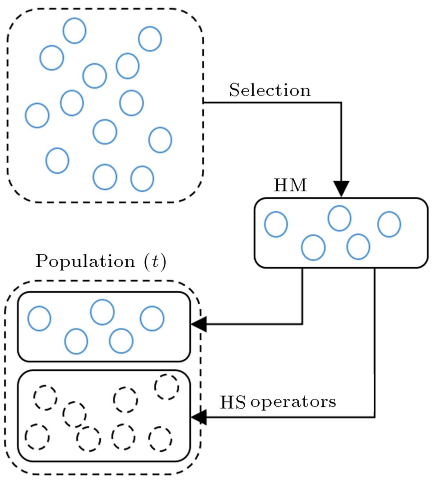


Figure 2. Procedure of the proposed HS-pop algorithm.

$$u_i^{t+1} = \begin{cases} q_{ij}^t & \text{if } q_{ij}^t \geq u_i^t \text{ or } f(Q_j^t) < U_i^t \\ u_i^t & \text{otherwise} \end{cases} \tag{24}$$

$$L_i^{t+1} = \begin{cases} f(Q_j^t) & \text{if } q_{ij}^t \leq l_i^t \text{ or } f(Q_j^t) < L_i^t \\ L_i^t & \text{otherwise} \end{cases} \tag{25}$$

$$U_i^{t+1} = \begin{cases} f(Q_j^t) & \text{if } q_{ij}^t \geq u_i^t \text{ or } f(Q_j^t) < U_i^t \\ U_i^t & \text{otherwise} \end{cases} \tag{26}$$

4.4.3. Population based harmony search (HS-pop)

In a traditional harmony search, only one new harmony vector or individual is generated during the reproduction process, and only one individual is examined in order to update the harmony memory. However, in this algorithm, a population based approach is applied in the harmony search algorithm. In other words, K new harmonies are generated at each generation. At each generation, as illustrated in Figure 2, K new individuals are generated with the following rules:

a) K' best individuals are selected from the previous population, $K' < K$. We assume that these individuals form the harmony memory ($K' = \text{HMS}$);

b) Then, $(K - K')$ offspring are generated based on harmony search operators, i.e. PAR and HMCR.

Algorithm 3 presents the template for HS-pop.

4.4.4. Harmony search based cultural algorithm (HS-CA)

Gao et al. [37] proposed a hybrid optimization method in which the HS algorithm is merged together with CA. First, the knowledge of the belief space is extracted from the harmony memory, and then used to direct the mutation of the new offspring. Using the historical values of individuals, we incorporate the HS algorithm into CA. We assume that the situational knowledge component of the CA belief space acts as a harmony memory matrix with the size of HMS. The procedure of the algorithm is illustrated in Figure 3. Here, all steps of the algorithm are similar to those of Algorithm 3, except for the reproduction process. Algorithm 4 presents the template for HS-CA.

The procedure of Algorithm 4 is described below in detail.

```
Begin
Initialize the problem and algorithm parameters (t = 0);
Initialize the population space, Pop^0;
    Repeat
      t = t + 1;
      Select the best K' individuals of Pop^t from Pop^(t-1),
      and form the HM matrix.
      Generate the remaining (K - K') individuals via
      HMCR and PAR operators;
Until stopping conditions = false
End
```

Algorithm 3. Population based harmony search (HS-pop) pseudo code.

```
Begin
Initialize the problem and algorithm parameters (t = 0);
Initialize the population space, Pop^0;
Initialize the normative component of the belief space, N^0;
    Repeat
      t = t + 1;
      Select the best K' individuals of Pop^t from Pop^(t-1),
      and form the HM matrix.
      Generate the remaining (K - K') individuals using
      HS and CA operators;
      Update the belief space;
    Until stopping conditions = false
End
```

Algorithm 4. Harmony search based cultural algorithm (HS-CA) pseudo code.

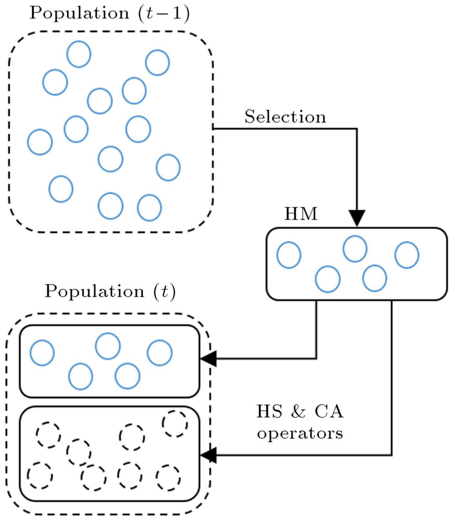


Figure 3. Procedure of the proposed HS-CA algorithm.

Initialization: The first population, including K solution vectors, is generated. Suppose that we denote the mth solution vector at iteration t with $Q_m^t = [Q_{m1}^t, Q_{m2}^t, \cdots, Q_{mn}^t]$, where $Q_{mi}^t, i = 1, 2, \cdots, n$, is the order quantity assigned for vendor i at iteration t, and n is the number of vendors. For each solution vector, an order quantity for vendor i is randomly chosen from its production interval. That is, Q^0_{mi} = an integer random number, $\in [0, u_i]$, $i = 1, 2, \cdots, n$, $m = 1, 2, \cdots, K$. Then, infeasible solutions are transformed to feasible ones by repairing the algorithm described in Section 4.1. All the individuals are evaluated by the fitness function, $f(.)$, and sorted, in ascending order, according to their fitness value in Pop^0:

$$Pop^0 = \begin{bmatrix} Q^0_1 \\ Q^0_2 \\ \vdots \\ Q^0_K \end{bmatrix} = \begin{bmatrix} Q^0_{11} & Q^0_{12} & \cdots & Q^0_{1n} \\ Q^0_{21} & Q^0_{22} & \cdots & Q^0_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ Q^0_{K1} & Q^0_{K2} & \cdots & Q^0_{Kn} \end{bmatrix}, \qquad (27)$$

where Q^0_1 and Q^0_K are the best and worst solutions, respectively, and we have $f(Q^0_1) \leq f(Q^0_2) \leq \cdots \leq f(Q^0_K)$. Recall that the belief space comprises normative knowledge and situational knowledge. Normative knowledge is denoted by $N^t = (\mathcal{X}^t_1, \mathcal{X}^t_2, \cdots, \mathcal{X}^t_n)$, where $\mathcal{X}^t_i = (l^t_i, u^t_i, L^t_i, U^t_i)$. Note that the closed interval characteristic for each vendor is initialized as below:

$$l^0_i = 0, \qquad u^0_i = u_i, \qquad L^0_i = \infty, \qquad U^0_i = \infty;$$
$$i = 1 \cdots n. \qquad (28)$$

We assume that at each generation, K' best individuals from the previous generation form the harmony memory matrix, with HMS being equal to K' (where $K' < K$). So, situational knowledge is initialized as below:

$$S^0 = \begin{bmatrix} Q^0_{11} & Q^0_{12} & \cdots & Q^0_{1n} \\ Q^0_{21} & Q^0_{22} & \cdots & Q^0_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ Q^0_{K',1} & Q^0_{K',2} & \cdots & Q^0_{K',n} \end{bmatrix}. \qquad (29)$$

Reproduction: The hybrid HS-CA applies elitism for K' the best individuals, in order to keep them from one generation to the next. The remaining $(K - K')$ individuals are generated based on harmony search operators, i.e. PAR and HMCR, as described in the following.

With an influence function, the knowledge in the belief space can be used to influence the creation of the offspring. We assume that both normative and situational (harmony memory) components are used during the offspring generation. Each variable in the new harmony vector or individual is generated as below:

$$Q^t_{.i} \leftarrow \begin{cases} Q^t_{.i} \in \{Q^{t-1}_{1i}, Q^{t-1}_{2i}, \cdots, Q^{t-1}_{K'i}\} \\ \quad \text{w.p. HMCR} \\ \\ Q^t_{.i} = Q^{t-1}_{.i} + (u^{t-1}_i - l^{t-1}_i) N_i(0,1) \\ \quad \text{w.p. } (1 - \text{HMCR}) \end{cases} \qquad (30)$$

where HMCR $\in [0, 1]$ is the Harmony Memory Considering Rate. Note that each variable of an individual is generated using either situational knowledge with the probability of HMCR, or normative knowledge with the probability of (1-HMCR). Then, each decision variable is examined as to whether or not be changed around its value, with a probability of PAR, using the following formula:

Pitch adjusting for:

$$Q^t_{.i} \leftarrow \begin{cases} \text{Yes} & \text{w.p. PAR} \\ \text{No} & \text{w.p. } (1 - \text{PAR}) \end{cases}$$

$$Q^t_{.i} \leftarrow Q^t_{.i} \pm \text{rand}() \times bw, \qquad (31)$$

where PAR $\in [0, 1]$ is the Pitch Adjustment Rate, and bw is maximum distance bandwidth of changing q^t_i.

Updating the belief space: We use the dynamic formula (32) to determine how many individuals should be selected from the Pop^t in order to shape the belief space:

$$n_{\mathcal{B}^t} = \left\lceil \frac{K\gamma}{t} \right\rceil, \qquad \gamma \in [0, 1], \qquad (32)$$

where t is the iteration number. These $n_{\mathcal{B}^t}$ best performers are selected to update the normative knowledge, i.e. for each Q^t_{mi}; $m = 1 \cdots n_{\mathcal{B}^t}$; $i = 1 \cdots n$, Eqs. (23)-(26) should be updated. Moreover, the situational knowledge or harmony memory (S^t = HM^t) should be updated by replacing the previous individuals by K' best individuals.

5. Parameters tuning

The parameters of the algorithms impress the solution quality. There are several ways of tuning the parameters. One is the Taguchi method. The Taguchi robust tuning method is a powerful tool in the DOE (design of experiment), in order to estimate the optimum values of parameters. This method applies the S/N (signal to noise) ratio for measuring the quality characteristics deviating from the desired values.

There are three categories in S/N ratio performance evaluations, depending on the goal of the problem, i.e. smaller-the-better, larger-the-better, and nominal-the-best. In this paper, the smaller-the-better quality characteristic is taken into account with the following formula:

$$S/N = -10 \log \left(\frac{1}{n} \sum_{i=1}^{n} y_i^2 \right), \qquad (33)$$

where y is the observed fitness, and n is the number of observations. Here, we use the Taguchi method only for tuning parameters of HS-pop and HS-CA. For SS and PSO parameters, please refer to Kamali et al. [25].

Table 1. Optimal levels of PSO's parameters.

	Swarm-size	$W_{\min}$	$W_{\max}$	c_1	c_2
Optimal level	100	0.2	0.7	0.8	0.8

Table 2. Optimal levels of SS's parameters.

	Pop-size	b_1	b_2
Optimal level	100	15	15

5.1. PSO and SS parameters

Kamaliet al. [25] applied the Taguchi method for parameter tuning, and determined appropriate levels for each algorithm parameter. The best levels of parameter for PSO and SS are summarized in Tables 1 and 2.

5.2. Tuning HS-pop parameters

Here, by applying the Taguchi method, we evaluate the impact of seven parameters on the output parameter (fitness function). These parameters and their levels are shown in Table 3. There are seven parameters, and each one has three levels. In order to conduct the experiment, the appropriate orthogonal array is $L_{18}(3^7 \times 2^1)$, which is tabulated in Table 4. Since the selected array has one additional parameter with 2 levels (P1), the additional parameter column can be easily ignored from the experiment.

Taking the full factorial model into account, $3^7 = 2187$ different combinations for each problem are reduced to 18 problems using the Taguchi method. In order to reduce experimental error, we repeat each experiment 5 times. The ANOVA test on the S/N ratio with 99.5% confidence limit is implemented and the results are shown in Table 5. The ANOVA indicates that Pop-size, HMS, HMCR and $\text{PAR}_{\max}$ (with P-values lower than 0.005) are the most significant parameters. These four parameters have the most sensitivity effect on the quality of solution, and other parameter impacts can be ignored.

The results of ANOVA indicate that there are no major differences between the levels of that parameter for each insignificant parameter. However, the level with the highest S/N ratio is the optimal level. Table 6 indicates the average S/N ratio for parameters. The significance of the parameters is calculated by the difference between max and min values for each parameter. As shown in Table 6, Pop-size gets the first highest value, HMS gets the second highest value, and so on. Hence, this result confirms the ANOVA results.

In order to determine the optimal levels of significant parameters, including Pop-size, HMS, HMCR and $\text{PAR}_{\max}$, the SNK (Student-Newman-Keuls) range test is applied and the results show that there is no significant difference between levels 2 (75) and 3 (90) of

Table 4. Experimental plan using L_{18} orthogonal array.

	Parameters							
Experiments	P1	P2	P3	P4	P5	P6	P7	P8
1	1	1	1	1	1	1	1	1
2	1	1	2	2	2	2	2	2
3	1	1	3	3	3	3	3	3
4	1	2	1	1	2	2	3	3
5	1	2	2	2	3	3	1	1
6	1	2	3	3	1	1	2	2
7	1	3	1	2	1	3	2	3
8	1	3	2	3	2	1	3	1
9	1	3	3	1	3	2	1	2
10	2	1	1	3	3	2	2	1
11	2	1	2	1	1	3	3	2
12	2	1	3	2	2	1	1	3
13	2	2	1	2	3	1	3	2
14	2	2	2	3	1	2	1	3
15	2	2	3	1	2	3	2	1
16	2	3	1	3	2	3	1	2
17	2	3	2	1	3	1	2	3
18	2	3	3	2	1	2	3	1

Table 5. ANOVA results for S/N ratio of Pop-HS.

Source	DF	ANOVA SS	Mean squares	F-value	P-value
Pop-size	2	0.0451	0.0226	704.5	0.0001
HMS	2	0.0107	0.0053	166.3	0.0008
HMCR	2	0.0048	0.0024	75.4	0.003
$\text{PAR}_{\min}$	2	0.0019	0.0010	29.8	0.010
$\text{PAR}_{\max}$	2	0.0052	0.0026	81.6	0.002
$\text{bw}_{\min}$	2	0.0008	0.0004	11.8	0.038
$\text{bw}_{\max}$	2	0.0011	0.0006	17.8	0.022
Error	3	0.0001	0.00003		
Total	17	0.0698			

Table 3. Introducing levels of parameters for Pop-HS.

	Pop-size	HMS	HMCR	$\text{PAR}_{\min}$	$\text{PAR}_{\max}$	$bw_{\min}$	$bw_{\max}$
Level 1	50	15	0.9	0.1	0.6	1	10
Level 2	75	30	0.93	0.15	0.75	5	20
Level 3	90	40	0.96	0.25	0.9	10	50

Table 6. Average S/N ratio and significance of Pop-HS parameters.

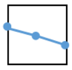
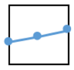
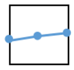
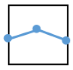
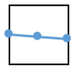
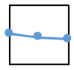
	Pop-size	HMS	HMCR	PAR_{min}	PAR_{max}	bw_{min}	bw_{max}
Level 1	18.022	18.122	18.073	18.080	18.083	18.101	18.103
Level 2	18.128	18.094	18.092	18.094	18.117	18.092	18.090
Level 3	18.128	18.062	18.113	18.105	18.078	18.085	18.084
Significance	0.107	0.060	0.040	0.025	0.038	0.017	0.019
Figure							

Table 7. Optimal levels of Pop-HS parameters.

	Pop-size	HMS	HMCR	PAR_{min}	PAR_{max}	bw_{min}	bw_{max}
Optimal level	75	15	0.96	0.25	0.75	1	10

Table 8. Introducing levels of parameters for HS-CA.

	Pop-size	HMS	HMCR	PAR_{min}	PAR_{max}	bw_{min}	bw_{max}
Level 1	50	15	0.9	0.1	0.6	10	500
Level 2	75	30	0.93	0.15	0.75	100	1000
Level 3	90	40	0.96	0.25	0.9	500	2000

Pop-size, but the S/N ratio of level 1(50) is significantly smaller than that of levels 2 and 3. So, we choose level 2 (75) for reducing computational time. Furthermore, there are no significant differences between levels of HMS, HMCR and PAR_{max}. So, for each parameter, we choose the level that gets the higher S/N ratio.

We repeat the experiments for $n = 4$ and derive the ANOVA results again. Results indicate similar levels for parameters. We conclude that the optimal levels for the parameters are as in Table 7.

5.3. Tuning HS-CA parameters

Here, similar to the previous subsection, there are seven parameters, and each one has three levels. These parameters and their levels are shown in Table 8. Table 9 summarizes the ANOVA test on S/N ratio with 99.5% confidence limit. It is obvious that Pop-size is the most significant parameter. We use the SNK test

Table 9. ANOVA results for S/N ratio for HS-CA.

Source	DF	ANOVA SS	Mean squares	F-value	P-value
Pop-size	2	0.1053	0.0526	49.78	**0.005**
HMS	2	0.0009	0.0005	0.44	0.678
HMCR	2	0.0087	0.0043	4.10	0.139
PAR_{min}	2	0.0213	0.0106	10.05	0.047
PAR_{max}	2	0.0150	0.0075	7.11	0.073
bw_{min}	2	0.0004	0.0002	0.19	0.833
bw_{max}	2	0.0121	0.0060	5.72	0.095
Error	3	0.0032	0.0011		
Total	17	0.1668			

for Pop-size, and other parameter optimal levels are determined based on the value of S/N ratio, which are tabulated in Table 10. So, the optimum level of each parameter is shown in Table 11.

6. Illustrative example

Suppose that a single buyer would like to purchase a product from 4 vendors. The same data as in Kamali et al. [25] is used here. The annual market demand is 100,000, and the unit holding cost per time unit (h_b) is 2.6. Table 12 summarizes vendor information. Also, Table 13 gives the information about the vendor offered quantity discount. Moreover, the optimal values of each objective, by neglecting the other objectives, are $Z^* = (1488623, 1813.621, 13822.47, 60744.51)$. Also, the decision maker preferences about weights on the objectives are: $W = (0.4, 0.1, 0.2, 0.3)$.

Table 14 shows that all algorithms result in almost the same value for each objective function. However, our proposed HS-CA algorithm yields a slightly better overall objective function (Z), by 0.04% over that of Kamali et al. [25]. Moreover, the order quantities obtained by our PSO and Pop-HS coincide with the results of Kamali et al. [25], but, SS and HS-CA result in different order quantities with better overall objective function.

7. Performance comparison of algorithms

For evaluating the performance of the proposed algorithms, we generate four problems with a different number of vendors, i.e. $n = 5$, 10, 15 and 20. Then,

Table 10. Average S/N ratio and significance of HS-CA parameters.

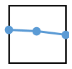
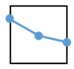
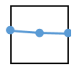
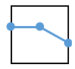
	Pop-size	HMS	HMCR	PAR_{min}	PAR_{max}	bw_{min}	bw_{max}
Level 1	17.901	18.016	18.035	18.055	18.043	18.015	18.026
Level 2	18.049	18.01	18.008	17.996	17.972	18.006	18.027
Level 3	18.075	17.998	17.981	17.973	18.009	18.004	17.972
Significance	0.174	0.018	0.054	0.082	0.071	0.011	0.055
Figure							

Table 11. Optimal levels of HS-CA parameters.

	Pop-size	HMS	HMCR	PAR_{min}	PAR_{max}	bw_{min}	bw_{max}
Optimal level	75	15	0.9	0.1	0.6	10	1000

Table 12. Vendors' information.

	Vendor			
	1	**2**	**3**	**4**
z	4.04	6.48	7.17	5.87
S	43	39	42	30
P	35108	29898	35785	68777
A	40	19	25	39
h	2.29	1.96	2.74	0.54
d	0.0344	0.0551	0.0121	0.0215
H	0.1444	0.1806	0.116	0.1581
w	0.7968	0.3629	0.326	0.505

Table 13. Discount intervals offered by vendors.

Vendor	Intervals	Unit prices
1	(0, 5000)	9
	[5000, 10000)	8.9
	[10000, 15000)	8.8
	[15000, 20000)	8.7
	[20000, 25000)	8.6
	[25000, 30000)	8.5
	[30000, 35108)	8.4
2	[0, 2000)	9.1
	[2000, 4000)	9
	[4000, 6000)	8.9
	[6000, 8000)	8.8
	[8000, 10000)	8.7
	[10000, 20000)	8.6
3	[0, 3000)	8.7
	[3000, 6000)	8.6
	[6000, 9000)	8.5
	[9000, 12000)	8.4
	[12000, 15000)	8.3
	[15000, 18000)	8.2
	[18000, 21000)	8.1
	[21000, 30000)	8
4	[0, 4000)	10.5
	[4000, 8000)	10.4
	[8000, 12000)	10.3
	[12000, 16000)	10.2
	[16000, 68777)	10.1

we run each algorithm ten times for each problem. In order to assess the quality of solutions, we use the Relative Percentage Deviation (RPD) as a performance measure, that is:

$$\text{RPD} = \frac{f^* - f}{f^*} \times 100, \qquad (34)$$

where f^* is the global optimum or best known solution, and f is an obtained solution for an instance. Table 15 shows the objective function values. It can be inferred from Table 15 that the HS-pop and HS-CA perform better than standard HS, and they all perform better than SS and PSO in finding best solutions. In order to better compare each algorithm performance, the RPD of each algorithm for each problem size is computed and illustrated by a box plot, as in Figure 4.

It can be implied from Figure 4 that HS-pop and HS-CA perform better than Scatter Search and Particle Swarm optimization algorithms; both in best known solutions and solution variability.

In order to evaluate the computational time of the proposed algorithms, the time to reach a solution with r% error is computed, where r is the maximum value of average RPD of different algorithms for given n. The value of r for $n = 5$, 10, 15 and 20 is equal to 7.8, 7.7, 9.2 and 9.4, respectively. We define the Convergence Index (CI) as the number of successful runs in which the algorithm reaches a solution with r% error in a time less than 300 seconds, over the total number of runs.

$$\text{CI} = \frac{\text{number of successful runs}}{\text{total number of runs}}. \qquad (35)$$

All the algorithms are coded in MATLAB 2012 and run

Table 14. Optimal solution for base data.

	HS	Pop-HS	HS-CA	SS	PSO	Kamali et al. [25]
Z	0.063098	0.063098	0.063064	0.063087	0.063095	0.063095
$Z_1(*10^6)$	1.5128	1.5128	1.5127	1.5127	1.5128	1.5128
$Z_2(*10^6)$	0.0023	0.0023	0.0023	0.0023	0.0023	0.0023
$Z_3(*10^6)$	0.0138	0.0138	0.0138	0.0138	0.0138	0.0138
$Z_4(*10^6)$	0.0543	0.0543	0.0543	0.0543	0.0543	0.0543
Q_1	5890	5890	2943	2972	5887	5887
Q_2	0	0	0	0	0	0
Q_3	6004	6004	3000	3030	6000	6000
Q_4	4884	4884	2440	2465	4881	4880

Table 15. Computational results of the proposed algorithms.

n	PSO		SS		HS-pop		HS-CA		HS	
	$f_{\min}$	$\bar{f}$	$f_{\min}$	$\bar{f}$	$f_{\min}$	$\bar{f}$	$f_{\min}$	$\bar{f}$	$f_{\min}$	$\bar{f}$
5	0.04861	0.0524	0.04868	0.04925	0.04861	0.04862	0.04861	0.04861	0.04861	0.04864
10	0.10519	0.10814	0.1028	0.10499	0.10045	0.10056	0.10044	0.10045	0.10053	0.10080
15	0.12271	0.12396	0.1198	0.12263	0.11367	0.11479	0.11357	0.11521	0.11402	0.11569
20	0.12613	0.13425	0.13139	0.13468	0.12331	0.12398	0.12309	0.12427	0.12409	0.12551

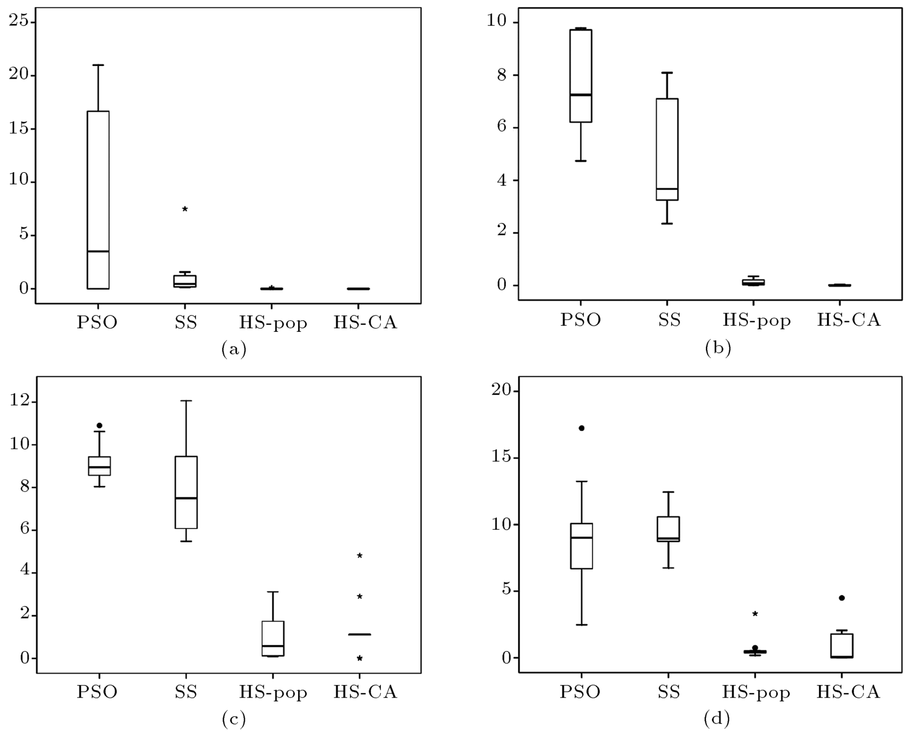


Figure 4. RPD of the proposed algorithms for a) $n = 5$, b) $n = 10$, c) $n = 15$, and d) $n = 20$.

on an Intel Core i3 2.10 GHz, HP Pavilion g6 at 4 GB RAM under a Microsoft Windows 7 environment. We run each algorithm 20 times and results are tabulated in Table 16. Note that the CPU time represents the average elapsing time of the algorithm in successful runs.

It is obvious from Table 16 that HS-pop and HS-CA significantly perform better than PSO and SS in both convergence index and CPU time. Moreover, there is no distinguishable difference between HS-pop and HS-CA. In order to make a comprehensive comparison between standard HS, HS-pop and HS-

Table 16. CPU time and convergence index of the proposed algorithms.

n	PSO		SS		HS-pop		HS-CA		HS	
	CI	CPU	CI	CPU	CI	CPU	CI	CPU	CI	CPU
5	70%	0.75	100%	0.41	100%	0.015	100%	0.015	100%	0.034
10	60%	1.3	90%	3.2	100%	0.020	100%	0.022	100%	0.053
15	55%	3.5	70%	4.9	100%	0.032	100%	0.031	100%	0.095
20	65%	5.6	40%	13.3	100%	0.038	100%	0.041	100%	0.255

Table 17. CPU time and convergence index of HS-pop and HS-CA.

n	$r = 3\%$						$r = 1\%$					
	HS-pop		HS-CA		HS		HS-pop		HS-CA		HS	
	CI	CPU	CI	CPU	CI	CPU	CI	CPU	CI	CPU	CI	CPU
5	100%	0.025	100%	0.027	100%	0.102	100%	0.076	100%	0.044	100%	0.619
10	100%	0.026	100%	0.043	100%	0.138	100%	0.070	100%	0.073	100%	2.890
15	90%	0.046	80%	0.061	80%	3.693	90%	1.323	35%	0.178	55%	4.922
20	100%	0.303	80%	0.095	90%	5.735	80%	1.180	25%	0.228	50%	7.923

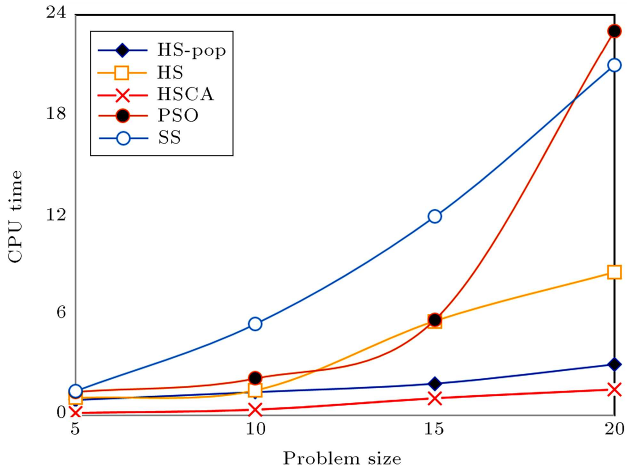


Figure 5. "CPU time-problem size" curve for optimization techniques.

CA, we run these algorithms again for $r = 1\%$ and $r = 3\%$. Table 17 shows the results. It can be inferred from Table 17 that although there is no considerable difference between the CPU times of the two algorithms, obviously, HS-pop performs better than HS-CA in convergence. Moreover, the standard HS takes much more CPU time to converge the solution.

We provide the "CPU time-problem size" curve for all optimization techniques. As shown in Figure 5, HS-CA and HS-pop have a better performance than standard HS, and they all perform better than SS and PSO.

8. Conclusion

Due to the existence of competition and market pressure, coordinating all entities within a supply chain is becoming increasingly critical. Some models have been developed to investigate the coordination problem, together with the vendor selection problem. However, little attention has been paid to developing efficient algorithms in this area. By applying the Global Criterion method, the multi-objective mixed integer nonlinear mathematical model is transformed into a single objective optimization problem. Due to the complexity of the problem, we propose four metaheuristics: PSO, SS, and two hybrid algorithms, i.e., HS-pop and HS-CA. Then, the comparison is performed among the parameter-tuned algorithms. Solving the sample problems, it is shown that the modified harmony search algorithms (HS-pop and HS-CA) have better performance than standard HS and they all perform better than SS and PSO in finding high quality solutions in less computational time.

References

1. Thomas, D.J. and Griffin, P.M. "Coordinated supply chain management", *Eur. J. Oper. Res.*, **94**(1), pp. 1-15 (1996).

2. Shin, H. and Tunca, T.I. "Do firms invest in forecasting efficiently? The effect of competition on demand forecast investment and supply chain coordination", *Oper. Res.*, **58**, pp. 1592-1610 (2010).

3. Fauli-Oller, R. and Sandonis, J. "Optimal two part tariff licensing contracts with differentiated goods and endogenous R&D", University of Allicante (2007).

4. Donohue, K.L. "Efficient supply contracts for fashion goods with forecast updating and two production models", *Manag. Sci.*, **46**, pp. 1397-1411 (2000).

5. Cachon, G.P. and Lariviere, M. "Supply chain coordi-

nation with revenue sharing contracts: Strengths and limitations", *Manag. Sci.*, **51**, pp. 30-44 (2005).

6. Tsay, A.A. "The quantity flexibility contract and supplier-customer incentives", *Manag. Sci.*, **45**, pp. 1339-1358 (1999).

7. Eppen, G.D. and Iyer, A.V. "Backup agreements in fashion buying - the value of upstream flexibility", *Manag. Sci.*, **43**, pp. 1469-1484 (1997).

8. Taylor, T.A. "Supply chain coordination under channel rebates with sales effort effects", *Manag. Sci.*, **48**, pp. 992-1007 (2002).

9. Li, X. and Wang, Q. "Coordination mechanisms of supply chain systems", *Eur. J. Oper. Res.*, **179**, pp. 1-16 (2007).

10. Sarlak, R. and Nookabadi, A. "Synchronization in multi-echelon supply chain applying timing discount", *Int. J. Adv. Manuf. Technol.*, **59**, pp. 289-297 (2012).

11. Pezeshki, Y., Baboli, A. and Akbari-Jokar, M.R. "Simultaneous coordination of capacity building and price decisions in a decentralized supply chain", *Int. J. Adv. Manuf. Technol.*, **64**, pp. 961-976 (2013).

12. Florez-Lopez, R. "Strategic supplier selection in the added-value perspective: A CI approach", *Inf. Sci.*, **177**(5), pp. 1169-1179 (2007).

13. Rosenthal, E.C., Zydiak, J.L. and Chaudhry, S.S. "Vendor selection with bundling", *Dec. Sci.*, **26**, pp. 35-48 (1995).

14. Sarkis, J. and Semple, J.H. "Vendor selection with bundling: A comment", *Dec. Sci.*, **30**(1), pp. 265-271 (1999).

15. Goossens, D.R., Maas, A.J.T., Spieksma, F.C.R. and Klundert, J.J. "Exact algorithms for procurement problems under a total quantity discount structure", *Eur. J. Oper. Res.*, **178**, pp. 603-626 (2007).

16. Ghodsypour, S.H. and O'Brien, C. "A decision support system for supplier selection using an integrated analytic hierarchy process and linear programming", *Int. J. Prod. Econ.*, **56**, pp. 199-212 (1998).

17. Amid, A., Ghodsypour, S.H. and O'Brien, C. "A weighted additive fuzzy multi-objective model for the supplier selection problem under price breaks in a supply chain", *Int. J. Prod. Econ.*, **121**, pp. 323-332 (2009).

18. Jayaraman, V., Srivastava, R. and Benton, W.C. "Supplier selection and order quantity allocation: A comprehensive model", *J. Suppl. Chain Manag.*, **35**, pp. 50-58 (1999).

19. Cakravastia, A., Toha, I.S. and Nakamura, N. "A two-stage model for the design of supply chain network", *Int. J. Prod. Econ.*, **80**, pp. 231-248 (2002).

20. Dahel, N.E. "Vendor selection and order quantity allocation in volume discount environments", *Suppl. Chain Manag.: Int. J.*, **8**(4), pp. 335-342 (2003).

21. Xia, W. and Wu, Z.H. "Supplier selection with multiple criteria in volume discount environments", *Omega*, **35**, pp. 494-504 (2007).

22. Ebrahim, R.M., Razmi, J. and Haleh, H. "Scatter search algorithm for supplier selection and order lot sizing under multiple price discount environment", *Adv. Eng. Soft.*, **40**, pp. 766-776 (2009).

23. Herer, Y.T., Rosenblatt, M.J. and Hefter, I. "Fast algorithms for single-sink fixed charge transportation problems with applications to manufacturing and transportation", *Transport. Sci.*, **30**(4), pp. 276-290 (1996).

24. Kim, T. and Goyal, S.K. "A consolidated delivery policy of multiple suppliers for a single buyer", *Int. J. Proc. Manag.*, **2**, pp. 267-287 (2009).

25. Kamali, A., Fatemi-Ghomi, S.M.T. and Jolai, F. "A multi-objective quantity discount and joint optimization model for coordination of a single-buyer multi-vendor supply chain", *Comput. Math. Appl.*, **62**, pp. 3251-3269 (2011).

26. Gheidar-Kjeljani, J., Ghodsypour, S.H. and Fatemi-Ghomi, S.M.T. "Supply chain optimization policy for a supplier selection problem: a mathematical programming approach", *Iran. J. Oper. Res.*, **2**(1), pp. 17-31 (2010).

27. Eberhart, R.C. and Kennedy, J. "A new optimizer using particle swarm theory", In *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, Nagoya, Japan, pp. 39-43 (1995).

28. Kennedy, J. and Eberhart, R.C. "Particle swarm optimization", In *IEEE International Conference on Neural Networks*, Perth, Australia, pp. 1942-1948 (1995).

29. Glover, F. "Heuristics for integer programming using surrogate constraints", *Dec. Sci.*, **8**, pp. 156-166 (1977).

30. Laguna, M. and Marti, R., *Scatter Search: Methodology and Implementations*, in C. Kluwer Academic Publishers, Boston, MA (2003).

31. Marti, R., Laguna, M. and Glover, F. "Principles of scatter search", *Eur. J. Oper. Res.*, **169**(2), pp. 359-372 (2006).

32. Geem, Z.W., Kim, J.H. and Loganathan, G. "A new heuristic optimization algorithm: Harmony search", *Simulation*, **76**(2), pp. 60-68 (2001).

33. Mahdavi, M., Fesanghary, M. and Damangir, E. "An improved harmony search algorithm for solving optimization problems", *Appl. Math. Comput.*, **188**(2), pp. 1567-1579 (2007).

34. Reynolds, R.G. "An introduction to cultural algorithms", In *Proceedings of the 3rd Annual Conference on Evolutionary Programming*, World Scientific Publishing, San Diego, Calif, USA, pp. 131-139 (1994).

35. Srinivasan, S. and Ramakrishnan, S. "Cultural algorithm toolkit for interactive knowledge discovery", *Int.*

J. Data Mining & Knowledge Manage. Process, **2**(5), pp. 53-70 (2012).

36. Sternberg, M. and Reynolds, R.G. "Using cultural algorithms to support re-engineering of rule based expert systems in dynamic environments: A case study in fraud detection", *IEEE Trans. Evol. Comput.*, **1**(4), pp. 225-243 (1997).

37. Gao, X.Z., Wang, X., Jokinen, T., Ovaska, S.J., Arkkio, A. and Zenger, K. "A hybrid optimization method for wind generator design", *Int. J. Innov. Comput. Inf. Control*, **8**(6), pp. 4347-4373 (2012).

Biographies

Saeed Alaei obtained BS and MS degrees in Industrial Engineering from Amirkabir University of Technology, Tehran, Iran, in 2009, and Sharif University of Technology, Tehran, Iran, in 2011, respectively. He is currently a PhD degree candidate in Industrial Engineering at K.N. Toosi University of Technology, Tehran, Iran. His research interests are mainly focused on supply chain coordination, multi-echelon inventory management and game theory.

Farid Khoshalhan received MS and PhD degrees in Industrial Engineering, in 1997 and 2001, respectively, from Tarbiat Modares University, Tehran, Iran. He is currently Assistant Professor in the Faculty of Industrial Engineering at K.N. Toosi University of Technology, Tehran, Iran. His research interests include e-commerce and e-business, evolutionary algorithms and metaheuristics, multiple criteria decision making and game theory.