Scientia Iranica E (2013) 20(6), 2188–2200

SCIENTIA IRANICA

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

Design of a multi-stage transportation network in a supply chain system: Formulation and efficient solution procedure

E. Mehdizadeh[a,*], F. Afrabandpei[a,b], S. Mohaselafshar[c] and B. Afshar-Nadjafi[a]

a. *Faculty of Industrial and Mechanical Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran.*
b. *Ilam Gas Treating Company, National Iranian Gas Company, Ilam, Iran.*
c. *Department of Industrial Engineering, Mazandaran University of Science and Technology, Babol, Iran.*

Received 29 April 2012; received in revised form 30 November 2012; accepted 19 February 2013

KEYWORDS
Multi-stage transportation problem;
Supply chain management;
Priority-based genetic algorithm;
Simulated annealing;
Response surface methodology.

Abstract. Nowadays, Supply Chain Management (SCM) is an interesting problem that has attracted the attention of many researchers. Transportation network design is one of the most important fields of SCM. In this paper, an integrated multi-stage and multi-product logistic network design including forward and reverse logistic is considered. At first, a Mixed Integer Nonlinear Programming model (MINLP) is formulated in such a way as to minimize purchasing and transportation costs. Then, a hybrid priority-based Genetic Algorithm (pb-GA), and Simulated Annealing algorithm (SA) are developed in two phases to find the proper solutions. The solution is represented by a matrix and a vector. Response Surface Methodology (RSM) is used in order to tune the significant parameters of the algorithm. Several test problems are generated in order to examine the proposed meta-heuristic algorithm performance.

1. Introduction

Supply Chain Management (SCM) is often described as the optimal delivery of products from supplier to customer. Typical SCM goals include transportation network design, facility location, production scheduling and efforts to improve costs and network responsiveness. Transportation network design is one of these goals which were proposed by Hitchcock in 1941 [1]. The objective is to find the way of transporting products from several sources to several destinations, so that the total cost can be minimized. Logistics is often defined as the art of bringing the right amount of right products to the right place [2]. So, efficiency of the supply chain could be considerable.

In the literature of logistics network design, many researchers have worked on forward logistics network design, and a few have researched on reverse logistics network design. In recent years, some papers have related to integrated logistics network design. In integration of forward and reverse logistic, Fleischmann et al. [3] studied the impact of product recovery on logistics network design, and showed that integration of the forward and reverse network leads to significant cost savings. Lee and Dong [4] proposed dynamic location and allocation models in forward and reverse logistic network. A two-stage multi-period stochastic programming model was developed for reverse logistics network design to account for the uncertainties. Pishvaee et al. [5] proposed a model for integrated logistics network design to avoid the sub-optimality caused by a separate, sequential design of forward and reverse logistics network. They developed a bi-objective MIP model to minimize the total costs and responsiveness of

*. *Corresponding author. Tel/Fax: +98 281 3675784*
E-mail addresses: emehdi@qiau.ac.ir, emqiau@yahoo.com (E. Mehdizadeh)

the network. They solved the problem by a memetic algorithm based on GA and three different local searches to find the Pareto solutions.

Farahani and Elahipanah [6] developed a bi-objective MILP model for Just In Time (JIT) distribution in a multi-period, multi-product and multi-channel network to minimize the costs and the sum of backorders and surpluses of products in all periods. They applied a hybrid Non-dominate Sorting Genetic Algorithm (NSGA) to solve the problem. Zegordi and Beheshti Nia [7] considered production and transportation scheduling in a two stage supply chain environment that is composed of m suppliers in the first stage and l vehicles at the second stage. The objective function was to minimize the total tardiness and total deviations of assigned work loads of suppliers from their quotas. They formulated the problem as an MIP problem, and proposed an algorithm, namely the multi-society genetic algorithm, to solve the problem.

Many factors affect the efficiency of the logistic networks. One of them is to determine vehicles to be used to carry products. The kind of vehicles that is used for moving products can play a key role in cost reduction. Vehicles should be selected in such a way that the retailers' demand can be satisfied with the minimum transportation cost, considering capacity and the limited number of vehicles. So in this research, in addition to unit transportation cost, based on the transportation distances, the cost of using vehicles is considered, too. In this case, the capacity of vehicles and the limited number of vehicles are considered. Also, we extend the multi-stage transportation problem to multi-product case.

The multi-stage logistic network, considered in this paper, consists of five stages; supplier, wholesaler, retailer, collection/inspection and potential disposal locations. In the reverse logistic, a quota of returned products in collection/inspection centers, which are useable, are returned to the supplier centers. Others that are not useable are sent to disposal centers. The problem intends to determine the optimal forward and reverse transportation network to satisfy the retailer demands of several products, and organize the returning products by using several kinds of vehicles with minimum cost. It is assumed that there are m vehicle types for transportations with limited budget for purchasing or hiring them. The capacity of vehicles and fixed travel cost of the vehicles are considered. The aim is to satisfy the demands of retailers for p products and to organize the return products with minimum costs.

Since the majority of logistics network design problems can be categorized as NP-hard [5], many heuristics and meta-heuristics methods have been developed for solving these problems. Recently, GAs have received considerable attention as an approach to optimization problems. This approach is greatly used for optimizing logistic network problems. The different ways of chromosome representation have been proposed in the literature. Michalewicz et al. [8] developed a non-linear transportation problem, and solved it by a non-standard genetic algorithm approach. They used the matrix representation to construct a chromosome and developed the matrix-based crossover and mutation. Li et al. [9] considered a multi-objective solid transportation problem and solved it using GA. They used the three-dimensional matrix to represent the chromosome. Gen and Cheng [10] developed a spanning tree method for solution representation. In this method, solutions are represented by arrays. As this method may result in infeasible solutions, repair mechanisms should be used. Considering the characteristics of a multi-stage transportation problem, Pb-GA with new decoding and encoding procedures has been developed by Gen et al. [11]. In this approach, solutions are encoded as arrays, in which the position of each cell represents the sources and depots, and also the value in cells represents the priorities. Also, they proposed a new crossover operator called Weight Mapping Crossover (WMX), and carried out an experimental study into two stages. The pb-GA and WMX crossover was used by Lee et al. [12] for designing a reverse logistic network. The pb-GA was also used by Pishvaee et al. [5] who used segment-based crossover as well. Lin et al. [13] used an extended pb-GA, named ep-GA, and represented the chromosome with two sections; in the first section, priorities are shown, and in the second section the information is guided about how to assign retailers and costumers. They also proposed a hybrid evolutionary algorithm based on ep-GA, combined a Local Search (LS) technique and proposed a new fuzzy logic control to enhance the search ability of EA.

In this paper, first, the problem is defined and a Mixed Integer Non-Linear Programming model (MINLP) is developed for integrated transportation and production in a supply chain. Then, a modified priority-based Genetic Algorithm (pb-GA) with a special chromosome structure is expanded to include multi-product case. It is combined with a Simulated Annealing algorithm (SA) to solve the problem.

The remainder of this article was organized as follows: In Section 2, the problem is described and a mathematical model is presented. The proposed algorithm is presented in Section 3. Parameters setting and computational results are given in Section 4. Section 5 concludes the paper and gives some areas for future research.

2. Problem description and formulation

The Integrated Forward/Reverse Logistics Network (IFRLN) that is discussed in this paper is a multi-

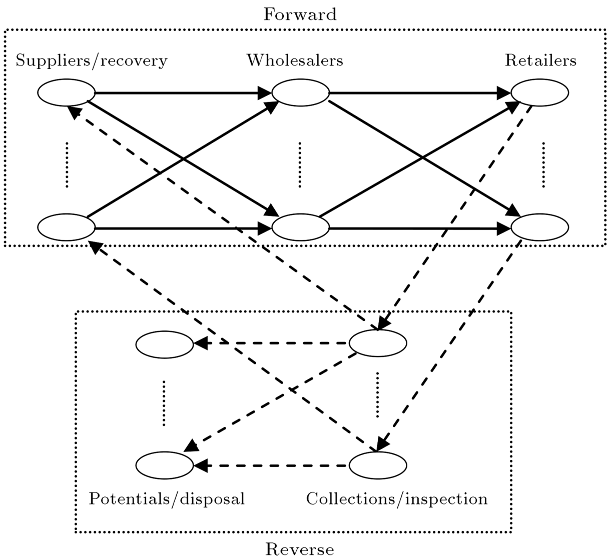

Figure 1. The schema of integrated forward/reverse logistics network.

stage logistics network including supplier/recovery, wholesaler, retailer, collection/inspection and disposal centers. The structure of the IFRLN can be presented in Figure 1. In the forward flow, new products are shipped from supplier/recovery centers to retailer centers directly (from the supplier/recovery centers to the retailer centers) and indirectly (from the supplier/recovery centers to the wholesaler centers and then to the retailer centers) to meet the demand of each retailer. In the reverse flow, returned products are collected in collection/inspection centers, and after testing, the recoverable products are shipped to supplier/recovery centers, and scrapped products are shipped to disposal centers.

In this section, a mathematical formulation for the problem is presented. The model has an objective function that minimizes the total purchasing costs of products, transportation costs of products based on the distances, purchasing or hiring costs of vehicles and travel costs of vehicles. The capacity of the sources and depots, and the capacity of the vehicles and limited number of the vehicles are considered in this network.

Minimizing costs: Purchasing cost of products, transportation cost of products, purchasing or hiring cost of vehicles, and travel cost of vehicles.

Subject to:

- Satisfying demands of all customers;
- Balancing of flow between nodes;
- Capacity constraints;
- Limited budget for purchasing vehicles;
- Assigning only one kind of vehicles for transporting each kind of products;
- Non-negativity and binary constraints.

2.1. Indices and sets

i Supplier/recovery center index (i = 1,..., I);

j Wholesaler center index (j = 1,..., J);

k Retailer center index (k = 1,..., K);

s Collection/inspection center index (s = 1,..., S);

n Disposal center index (n = 1,..., N);

p Product index (p = 1,..., P);

m Vehicle index (m = 1,..., M).

2.2. Decision variables

Y_{pij} Amount of product p transported from supplier/recovery center i to wholesaler center j;

V_{pik} Amount of product p transported from supplier/recovery center i to retailer center k;

U_{pjk} Amount of product p transported from wholesaler center j to retailer center k;

CI_{pks} Amount of return product p transported from retailer center k to collection/inspection center s;

PR_{psi} Amount of return product p transported from collection/inspection center s to supplier/recovery center i;

DC_{psn} Amount of return product p transported from collection/inspection center s to disposal center n;

B_{1mpij} 1, if vehicle m is used to carry product p between supplier/recovery center i to wholesaler center j, 0 otherwise;

B_{2mpik} 1, if vehicle m is used to carry product p between supplier/recovery center i to retailer center k, 0 otherwise;

B_{3mpjk} 1, if vehicle m is used to carry product p between wholesaler center j to retailer center k, 0 otherwise;

B_{4mpks} 1, if vehicle m is used to carry product p between retailer center k to collection/inspection center s, 0 otherwise;

B_{5mpsi} 1, if vehicle m is used to carry product p between collection/inspection center s to supplier/recovery center i, 0 otherwise;

B_{6mpsn} 1, if vehicle m is used to carry product p between collection/inspection center s to disposal center n, 0 otherwise;

2.3. Model parameters

d_{pk} Amount of demand for product p by retailer center k;

b_m Maximum budget for purchasing or hiring vehicle m;

a_{mp} Capacity of vehicle m for transporting product p;

pu_{1pi} Purchasing cost of product p from supplier/recovery center i;

pu_{2m} Purchasing or hiring cost of vehicle m;

c_p Unit transportation cost of product p along unit distance;

r_{pk} Rate of return of product p of retailer center k;

ca_{1pi} Supply capacity of supplier/recovery center i for product p;

ca_{2pj} Delivery capacity of wholesaler center j for product p;

ca_{3s} Delivery capacity of collection/inspection center s;

ca_{4n} Delivery capacity of disposal center n;

ca_{5i} Recovery capacity of supplier/recovery center i;

ρ Average disposal fraction $0 \leq \rho \leq 1$;

γ, M A large number $\gamma \geq \rho \sum_k r_{pk}.d_{pk}\ \forall p$;

g_{1ij} Distance between supplier/recovery center i to wholesaler center j;

g_{2ik} Distance between supplier/recovery center i to retailer center k;

g_{3jk} Distance between wholesaler center j to retailer center k;

g_{4ks} Distance between retailer center k to collection/inspection center s;

g_{5si} Distance between collection/inspection center s to supplier/recovery center i;

g_{6sn} Distance between collection/inspection center s to disposal center n;

c_{1mij} Fixed cost of using vehicle m to carry products between supplier/recoverycenter i to wholesaler center j;

c_{2mik} Fixed cost of using vehicle m to carry products between supplier/recovery center i to retailer center k;

c_{3mjk} Fixed cost of using vehicle m to carry products between wholesaler center j to retailer center k;

c_{4mks} Fixed cost of using vehicle m to carry products between retailer center k to collection/inspection center s;

c_{5msi} Fixed cost of using vehicle m to carry products between collection/inspection center s to supplier/recovery center i;

c_{6msn} Fixed cost of using vehicle m to carry products between collection/inspection center s to disposal center n.

2.4. Mathematical formulation

In terms of the above-mentioned notations, the IFRLN design problem can be formulated as follows:

Min

$$\begin{aligned} z = &\sum_i \sum_j \sum_p (Y_{pij}(pu_{1pi} + c_p g_{1ij}) \\ &+ \sum_m (pu_{2m} + c_{1mij}(Y_{pij}/a_{mp}))B_{1mpij}) \\ &+ \sum_i \sum_k \sum_p (V_{pik}(pu_{1pi} + c_p g_{2ik}) \\ &+ \sum_m (pu_{2m} + c_{2mik}(V_{pik}/a_{mp}))B_{2mpik}) \\ &+ \sum_j \sum_k \sum_p (U_{pjk} c_p g_{3jk} + \sum_m (pu_{2m} \\ &+ c_{3mjk}(U_{pjk}/a_{mp}))B_{3mpjk}) \\ &+ \sum_k \sum_s \sum_p (CI_{pks} c_p g_{4ks} + \sum_m (pu_{2m} \\ &+ c_{4mks}(CI_{pks}/a_{mp}))B_{4mpks}) \\ &+ \sum_s \sum_i \sum_p (PR_{psi} c_p g_{5si} + \sum_m (pu_{2m} \\ &+ c_{5msi}(PR_{psi}/a_{mp}))B_{5mpsi}) \\ &+ \sum_s \sum_n \sum_p (DC_{psn} c_p g_{6sn} + \sum_m (pu_{2m} \\ &+ c_{6msn}(DC_{psn}/a_{mp}))B_{6mpsn}), \end{aligned} \tag{1}$$

S.t.

$$\sum_j U_{pjk} + \sum_j V_{pik} = d_{pk} \qquad \forall k, p, \tag{2}$$

$$\sum_i Y_{pij} \geq \sum_k U_{pjk} \qquad \forall p, j, \tag{3}$$

$$\sum_j Y_{pij} + \sum_k V_{pik} \leq ca_{1ip} \qquad \forall i, p, \tag{4}$$

$$\sum_i Y_{pij} \leq ca_{2jp} \qquad \forall j, p, \tag{5}$$

$$\sum_k U_{pjk} \leq ca_{2jp} \qquad \forall j, p, \tag{6}$$

$$\sum_p \sum_k CI_{pks} \leq ca_{3s} \qquad \forall s, \tag{7}$$

$$\sum_p \sum_s DC_{psn} \leq ca_{4n} \qquad \forall n, \tag{8}$$

$$\sum_p \sum_s PR_{psi} \leq ca_{5i} \qquad \forall i, \tag{9}$$

$$\sum_s PR_{psi} \leq \gamma . \sum_j Y_{pij} \qquad \forall i, p, \tag{10}$$

$$\sum_s CI_{pks} = r_{pk} . d_{pk} \qquad \forall p, k, \tag{11}$$

$$\sum_n DC_{psn} = \rho . \sum_k CI_{pks} \qquad \forall p, s, \tag{12}$$

$$\sum_i PR_{psi} = (1-\rho) . \sum_k CI_{pks} \qquad \forall p, s, \tag{13}$$

$$pu_{2m} \sum_p \left(\sum_i \left(\sum_j B_{1mpij} + \sum_k B_{2mpik} \right) + \sum_j \sum_k B_{3mpjk} + \sum_k \sum_s B_{4mpks} + \sum_s \left(\sum_i B_{5mpsi} + \sum_n B_{6mpsn} \right) \right) \leq b_m \qquad \forall m, \tag{14}$$

$$Y_{pij} \leq M . \sum_m B_{1mpij} \qquad \forall i, j, p, \tag{15}$$

$$V_{pik} \leq M . \sum_m B_{2mpik} \qquad \forall i, k, p, \tag{16}$$

$$U_{pjk} \leq M . \sum_m B_{3mpjk} \qquad \forall j, k, p, \tag{17}$$

$$CI_{pks} \leq M . \sum_m B_{4mpks} \qquad \forall k, s, p, \tag{18}$$

$$PR_{psi} \leq M . \sum_m B_{5mpsi} \qquad \forall s, i, p, \tag{19}$$

$$DC_{psn} \leq M . \sum_m B_{6mpsn} \qquad \forall s, n, p, \tag{20}$$

$$\sum_m B_{1mpij} \leq 1 \qquad \forall i, j, p, \tag{21}$$

$$\sum_m B_{2mpik} \leq 1 \qquad \forall i, k, p, \tag{22}$$

$$\sum_m B_{3mpjk} \leq 1 \qquad \forall j, k, p, \tag{23}$$

$$\sum_m B_{4mpks} \leq 1 \qquad \forall k, s, p, \tag{24}$$

$$\sum_m B_{5mpsi} \leq 1 \qquad \forall s, i, p, \tag{25}$$

$$\sum_m B_{6mpsn} \leq 1 \qquad \forall s, n, p, \tag{26}$$

$$B_{1mpij},\ B_{2mpik},\ B_{3mpjk},\ B_{4mpks},\ B_{5mpsi},\ B_{6mpsn} \in \{0,1\} \quad \forall i, j, k, s, n, p, \tag{27}$$

$$Y_{pij},\ U_{pjk},\ V_{pik},\ CI_{pks},\ DC_{psn},\ PR_{psi} \geq 0 \quad \forall i, j, k, s, n, p. \tag{28}$$

In the objective function (1), first and second terms represent the purchasing cost and transportation cost of products based on the distances, purchasing or hiring cost of vehicles and travel cost of vehicles to carry products from supplier/recovery centers to wholesaler and retailer centers, respectively. 3rd, 4th, 5th and 6th terms represent transportation cost of products based on the distances, purchasing or hiring cost of vehicles and travel cost of vehicles to carry goods and use products between related sources and depots.

Constraint (2) denotes that the total amount of products sent to the retailer centers should be equal to their total demands. Constraint (3) assures that the amount of products sent by each wholesaler center to the retailer center does not exceed the inventory of the warehouse. Constraints (4)-(10) are capacity constraints on facilities. Constraint (11) assures that the ratio of demands as return products are collected in the collection centers. Constraints (12) and (13) assure that in the reverse logistic, a quota of returned products in collection/inspection centers, which are useable, are returned to the supplier centers and others that are unuseable are sent to disposal centers. Constraint (14) represents the constraint of budget for purchasing or hiring vehicles. Constraints (15)-(20) enforce that there should be at least one vehicle to carry products. Constraints (21)-(26) enforce that for each path and each product, only one kind of vehicle should be used. Constraint (27) denotes the binary variables. Constraint (28) represents the non-negativity restriction of the decision variables.

Table 1. Parameters value in a small example.

	Parameters value																	
p	2																	
m	3																	
i	2																	
j	2																	
k	3																	
s	2																	
n	2																	
d_{pk}	100	217	169	123	220	195												
b_m	150000	250000	200000															
a_{mp}	23	12	26	18	16	29												
pu_{1pi}	3459	4113	1888	2198														
pu_{2m}	45000	25200	30792															
c_p	80	60																
r_{pk}	0.05	0.1	0.09	0.06	0.04	0.1												
ca_{1pi}	1200	900	800	1000														
ca_{2pj}	550	650	500	610														
ca_{3s}	100	86																
ca_{4n}	70	55																
ca_{5i}	120	90																
ρ	0.2																	
γ	10000																	
M	10000																	
g_{1ij}	132	165	197	107														
g_{2ik}	300	180	158	350	232	262												
g_{3jk}	131	105	124	75	90	147												
g_{4ks}	59	66	73	55	95	80												
g_{5si}	100	80	140	180														
g_{6sn}	200	840	640	400														
c_{1mij}	84	65	97	51	71	69	88	89	59	74	72	82						
c_{2mik}	145	150	93	141	138	79	74	119	175	100	130	86	150	90	120	143	166	175
c_{3mjk}	144	100	133	132	106	119	85	83	117	134	145	89	119	112	80	103	91	135
c_{4mks}	109	186	100	169	173	178	107	135	123	172	138	181	116	123	113	112	178	152
c_{5msi}	71	55	80	74	64	70	66	53	59	54	57	59						
c_{6msn}	106	78	81	154	156	121	75	91	101	143	71							

In order to validate the performance of the model, we present a small example and solve it with LINGO software. The parameters are presented in Table 1 and the variables and objective function are presented in Table 2. It seems that the obtained solution is reasonable.

3. Solution approach

Although the exact algorithms find the optimal solution, the problems with real size are time consuming. So, the meta-heuristic algorithms are used to find the near optimal solution in a reasonable time span. Since the majority of logistics network design problems can be categorized as NP-hard [5], many heuristics and meta-heuristics methods have been developed for solving these problems. In this section, first, the chromosome representation is described, and then a meta-heuristic algorithm is proposed based on GA and SA to find the optimal solution in two phases.

3.1. Chromosome representation

In our problem, the solution is represented by a matrix and a vector. In the matrix, the priority-

Table 2. Variables value in a small example.

Variables and objective function value													
Variable	Y_{122}	Y_{222}	B_{11122}	B_{13222}	V_{112}	V_{113}	V_{212}	V_{213}	B_{21113}	B_{22112}	B_{23212}	B_{23213}	U_{121}
Value	100	123	1	1	217	169	220	195	1	1	1	1	100
Variable	U_{221}	B_{32121}	B_{33221}	CI_{111}	CI_{121}	CI_{131}	CI_{211}	CI_{221}	CI_{231}	B_{42111}	B_{42121}	B_{42131}	B_{42211}
Value	123	1	1	5	21.7	15.2	7.38	8.8	19.5	1	1	1	1
Variable	B_{42221}	B_{42231}	PR_{112}	PR_{212}	B_{52112}	B_{53212}	DC_{111}	DC_{211}	B_{61111}	B_{63211}			
Value	1	1	33.53	28.55	1	1	8.38	7.14	1	1			
Objective value 0.1665004E+08													

based encoding method, proposed by Gen et al. [11], is used. In this approach, solutions are encoded as arrays in which the position of each cell represents the sources and depots, and the value of cells represent the priorities. In the vector, the assigned vehicles to carry the products between the sources and depots are represented in the vector.

To apply the priority-based encoding method to the problem, for each product type, the priorities are represented in different rows. The chromosome consists of five segments, each of which is related to one echelon of the IFRLN. A typical example of the matrix is shown in Figure 2. The modified priority-based decoding algorithm of a segment is shown in Figure 3.

To decode an IFRLN chromosome, the second segment should be decoded before the first segment. Decoding of the second segment contains determining the shipment from supplier/recovery centers to the retailer centers or shipment from wholesaler centers to the retailer centers. The demand of a retailer center would be satisfied with a supplier/recovery center or a wholesaler center for the minimum cost. Therefore, the U_{pjk} and V_{pik} could be calculated. After that, the first segment should be decoded to determine Y_{pij}. In this segment, the demand of a wholesaler center, j, for product, p, is what should be sent to the retailer centers ($b_{p,j} = \sum_k U_{pjk}$). Then, the third, fourth and fifth segments are decoded, respectively, and CI_{pks}, DC_{psn} and PR_{psi} are calculated. Note that in the third segment, $b_{p,k} = r_{pk}.d_{pk}$, in the fourth segment, $b_{p,s} = \rho.\sum_k CI_{pks}$ and in the fifth segment, $b_{p,s} = (1-\rho).\sum_k CI_{pks}$.

	First segment				Second segment							Third segment					Fourth segment				Fifth segment			
	i		j		i		j		k			k			s		s		n		s		i	
	1	2	1	2	1	2	1	2	1	2	3	1	2	3	1	2	1	2	1	2	1	2	1	2
p=1	3	1	2	4	2	3	7	5	6	1	4	1	4	2	3	5	2	4	3	1	2	1	3	4
p=2	1	3	4	2	6	2	1	4	7	3	5	5	3	2	1	4	3	1	4	2	4	2	3	1

Figure 2. The solution representation with the modified priority-based encoding method.

Repeat

$p \leftarrow$ Select a random product from available products.
Select a node with maximum priority in the pth row of the segment.
If the node is a source, then select an available depot with minimum cost.
If the node is a depot, then select an available source with minimum cost.
$j \leftarrow$ the source $k \leftarrow$ the depot
$b_{p,k} \leftarrow$ demand of the depot k for product p
$ca_{p,j} \leftarrow$ capacity of the source j for product p
Amount of product p to be shipped between source j and depot $k \leftarrow$ Min ($ca_{p,j}$, $b_{p,k}$)
Reduce the $b_{p,k}$ and $ca_{p,j}$
If $ca_{p,j} = 0$, then omit the jth source for product p and omit its priority number in the priority matrix.
If $b_{p,k} = 0$, then omit the kth depot for product p and omit its priority number in the priority matrix.
If the all priorities of the product p were examined, then omit product p from available products.
If the all products were examined, then exit.

End

Figure 3. The modified priority based decoding algorithm of a segment.

2	3	1	2	3	2	1	3	1	2	1	3	2	3	2	3	2	2	3

Figure 4. The representation of assignment vector.

In assignment vector, potentially available vehicles for transporting all available products in all available routes are represented. The available vehicles for transporting Y_{pij}, U_{pjk}, V_{pik}, CI_{pks}, DC_{psn} and PR_{psi} are represented, respectively, in the vector. A typical example of the vector is shown in Figure 4.

3.2. Operators

Crossover operator: in the matrix of priorities, segment-based crossover is used. In each row of parents, the corresponding segments are selected alternately with equal probability, and are simply swapped to generate offsprings (Figure 5).

Mutation operator: in the matrix of priorities, segment-based mutation is used. In each row of selected chromosomes, some segments are selected alternately for mutation. In each selected segment, allele-based mutation is used; two alleles are selected randomly and swapped.

Neighborhood search: in assignment vector, for finding the neighboring solution of the current solution, some alleles are selected randomly from the assignment vector, and their numbers are added to the available vehicles set. Then, some numbers from this set are selected randomly and assigned to the routes indicated by the selected alleles.

3.3. The proposed algorithm

The proposed algorithm consists of two phases. In the first phase, the optimal routes and amounts of products that must be carried in the routes are determined. Then, in the second phase, the optimal vehicles for transporting the products are determined.

3.3.1. First phase

In this phase, the optimal routes and amounts of products which must be carried in the routes are determined using GA. For generating an initial population, random numbers from 1 to each segment size are generated and represented in the segments of the matrices.

In decoding procedure, the costs are determined by variable transportation costs (and purchasing costs) without vehicle costs. Also, for evaluation of chromosomes, fitness function is calculated without vehicles costs. For each solution, if the number of all its routes is more than the number of available vehicles, a very big number, as a penalty function, is added to the fitness function to avoid infeasible solutions. The number of available vehicles of kind m is determined by $[b_m/pu_{2m}]$.

The roulette wheel selection method is used to select parents. The crossover and mutation operators are used as mentioned. The termination condition is reaching to maximum generation number and a feasible solution, in which the entire routes is less than or equal to the total number of vehicles.

3.3.2. Second phase

After determining the optimal routes and amounts of products, which must be carried in the first phase, in the next phase, the kind of vehicles for transporting the products between the selected sources and depots are determined. The number of optimal routes is less than or equal to the available vehicles. The SA algorithm is used to determine the optimal assignment of vehicles to the routes. For generating an initial solution, the available vehicles for transporting Y_{pij}, U_{pjk}, V_{pik}, CI_{pks}, DC_{psn} and PR_{psi} are selected randomly from the available vehicles set, and are represented respectively in the vector.

In this phase, the fitness function is only calculated by the costs of using vehicles for transporting

Parent 1

3	1	2	4	2	3	7	5	6	1	4	1	4	2	3	5	2	4	3	1	2	1	3	4
1	3	4	2	6	2	1	4	7	3	5	5	3	2	1	4	3	1	4	2	4	2	3	1

Parent 2

4	3	1	2	1	4	6	7	5	2	3	2	3	4	1	5	4	3	2	1	4	2	1	3
1	2	3	4	2	4	6	1	3	5	7	5	4	3	2	1	1	2	3	4	1	3	4	2

Offspring 1

4	3	1	2	2	3	7	5	6	1	4	2	3	4	1	5	2	4	3	1	4	2	1	3
1	3	4	2	2	4	6	1	3	5	7	5	3	2	1	4	1	2	3	4	4	2	3	1

Offspring 1

3	1	2	4	1	4	6	7	5	2	3	1	4	2	3	5	4	3	2	1	2	1	3	4
1	2	3	4	6	2	1	4	7	3	5	5	4	3	2	1	3	1	4	2	1	3	4	2

Figure 5. The example of crossover operator.

Phase 1:

Y= [ ]; U= [ ]; V= [ ]; CI= [ ]; DC= [ ]; PR= [ ];

Step 1:
Get crossover rate (p_c), mutation rate (p_m), population size (*popsize*) and maximum iteration (*maxitr*); *itr*=1.

Step 2:
Randomly generate chromosomes as initial population.

Step 3:
Decode chromosomes and determine Y_{pij}, U_{pjk}, V_{pik}, CI_{pks}, DC_{psn}, PR_{psi} , fitness values and number of routs for each solution.
Select best chromosome based on fitness values.

Step 4:
Select a number of best chromosomes for reproduction.
Select a pair of parents by roulette wheel selection method.
Apply Crossover operator on selected parents to generate offspring.
Select a number of chromosomes for mutation randomly.
Apply mutation operator on selected chromosomes.

Step 5:
If $itr \leq maxitr$ or solution is infeasible, *itr*=*itr*+1 and go to ***step 3***.
Otherwise, terminate.

Phase 2:

Step 1:
Get t_0, a and k; t_c=t_0; *itr* =1.

Step 2:
Generate an initial solution of assignment vector and regarding it as a best solution.

Step 3:
Evaluate the solution using the second part of the fitness function.
If the fitness value of new solution (F_n) is less than the best solution (F_o), accept the new solution as the best solution.
Otherwise, accept the new solution as the best solution with probability of $e^{\frac{F0-Fn}{tc}}$.
MA= the best solution

Step 4:
M=all vehicles set.
Find the neighborhood solution of the current solution.

Step 5:
If $itr < k$, *itr*=*itr*+1 and go to ***step 3*** of this phase.

Step 6:
$t_c = a * t_c$.
If $t_c > 0$, *itr*=1 and go to ***step 3*** of this phase.
Otherwise, terminate.

n=1
For p, i and j, if $Y_{pij} \neq 0$, then m=$MA(n)$, B_{1mpij}=1 and n=n+1.
For p, i and k, if $V_{pik} \neq 0$, then m=$MA(n)$, B_{2mpik}=1 and n=n+1.
For p, j and k, if $U_{pjk} \neq 0$, then m=$MA(n)$, B_{3mpjk}=1 and n=n+1.
For p, k and s, if $CI_{pks} \neq 0$, then m=$MA(n)$, B_{4mpks}=1 and n=n+1.
For p, s and i, if $PR_{psi} \neq 0$, then m=$MA(n)$, B_{5mpsi}=1 and n=n+1.
For p, s and n, if $DC_{psn} \neq 0$, then m=$MA(n)$, B_{6mpsn}=1 and n=n+1.

At the end, summarize the fitness value of ***phase 1*** and ***phase 2***.

Figure 6. Proposed algorithm for the IFRLN.

amounts of products that has been determined in the first phase.

After determining the optimal fitness function values in the first and second phases, the results are summarized to show the true fitness function. The proposed algorithm for the IFRLN is summarized in Figure 6.

4. Computational results

In order to validate the performance of the algorithm, we generate several instances. The mathematical model of IFRLN is coded in LINGO optimization software, and the proposed meta-heuristic algorithm is coded in MATLAB on a computer with 4.0 GB Ram and 2.66 GHz processor. The Response Surface Methodology (RSM) is used to determine the optimal parameters of the algorithm. The model solutions and the meta-heuristic solutions are compared on the problem instances.

4.1. Data generation

Here, fiftheen instances are defined that can be characterized by the number of products (n_p), between 2 and 7, vehicles (n_m) between 2 and 6, supplier/recovery centers (n_i) between 2 and 9, wholesaler centers (n_j) between 2 and 11, retailer centers (n_k) between 2 and 18, collection/inspection centers (n_s) between 2 and 8

Table 3. Parameters range in the test problems.

Parameters	Small instances	Big instances	Parameters	Small instances	Big instances
d_{pk}	[100 - 220]	[100 - 220]	Ca_{1ip}	[800 - 1200]	[800 - 1200]
b_m	[150,000 - 700,000]	[700,000 - 1,400,000]	Ca_{2jp}	[500 - 700]	[500 - 700]
pu_{1pi}	[1000 - 5000]	[1000 - 5000]	Ca_{3s}	[80 -100]	[200 -300]
pu_{2m}	[10,000 - 45,000]	[10,000 - 45,000]	Ca_{4n}	[50 - 80]	[80 -100]
a_{mp}	[10 - 30]	[10 - 30]	Ca_{5i}	[80 - 150]	[180 - 250]
c_p	[50 - 100]	[50 - 100]	ρ	0.2	0.2
g_{1ij}	[100 - 200]	[100 - 200]	C_{1mij}	[50 - 100]	[50 - 100]
g_{2ik}	[150 - 400]	[150 - 400]	C_{2mik}	[60 - 180]	[60 - 180]
g_{3jk}	[70 - 150]	[70 - 150]	C_{3mjk}	[80 - 150]	[80 - 150]
g_{4ks}	[50 - 100]	[50 - 100]	C_{4mks}	[100 - 190]	[100 - 190]
g_{5si}	[80 - 200]	[80 - 200]	C_{5msi}	[50 - 80]	[50 - 80]
g_{6sn}	[200 - 1000]	[200 - 1000]	C_{6msn}	[70 - 160]	[70 - 160]
r_{pk}	[0.0 - 0 .1]	[0.0 - 0 .1]			

and disposal centers (n_n) between 2 and 7. The data required in the IFRLN problem are generated randomly as shown in Table 3.

4.2. Parameters tuning of the proposed algorithm

The parameters employed in algorithms should be selected properly to obtain a satisfactory solution quality in an acceptable time span. The RSM method is used to determine the optimal parameters of the algorithm. This is a technique for determining and representing the cause-and-effect relationship between true mean responses and input control variables influencing the responses as a multi-dimensional hyper surface [14]. This method has four stages. In the first stage, the independent parameters and their levels are determined. Some points (scenarios) are selected using these levels. In the second stage, the proposed algorithm is applied to several test problems, using these points. The results are normalized with relative percentage deviation (RPD) criteria (Eq. (29)). After collecting the data, the third stage is the prediction of the model equation and obtaining the response surface as a function of the independent variables (parameters and their interactions). Significant variables and coefficient of each variable are found, so the regression equation is determined. The fourth stage is determination of the optimum points of the equation.

$$RPD = \frac{\mathrm{Alg}_{sol} - \min_{sol}}{\min_{sol}}. \tag{29}$$

The crossover rate (p_c), mutation rate (p_m), initial temperature (t_0), iterations at a specific temperature (k) and cooling rate (a) are the five important factors affecting the proposed algorithm. So, effects of these parameters and their interactions are studied as input variables in the optimization procedure. For each parameter, the levels of parameters are defined as shown in Table 4.

Table 4. Levels of proposed algorithm parameters.

Parameter	Lower level	Middle value	Upper level
p_c	0.5	0.6	0.7
p_m	0.1	0.15	0.2
t_0	30	45	60
k	60	90	120
a	0.94	0.96	0.98

The 2^{5-2} points, using two-level factorial design, 4 central points and 2×5 axial points are selected. For each point, each problem instance is carried out 5 times, and average value of the objective function values and the processing times are recorded. These results are normalized by RPD criteria, and the average values for each point are calculated. The two regression models of the objective function and the processing time are determined using minitab14 software. The two regression models are optimized as a bi-objective problem, using a bi-objective technique and lingo9 software. The first equation is optimized, then, its solution is considered as a constraint in optimizing the second equation.

The results show that the models were highly significant with $p-$values $= 0.000$. The optimal values are $p_c = 0.58$, $p_m = 0.17$, $t_0 = 25$, $k = 144$ and $a = 0.97$.

Table 5. Results of LINGO and the proposed algorithm solutions.

Problem	$(n_p, n_m, n_i, n_j, n_k, n_s, n_n)$	LINGO			Proposed meta-heuristic		GAP
		Best objective (A)	Objective bound	CPU Time (sec)	Average value (B)	Average CPU time (sec)	
1	(2,3,2,2,3,2,2)	1.66 E+07	1.66 E+07	2	1.66 E+07	6.30	0.0000
2	(3,2,3,3,4,3,2)	3.57E+07	3.57E+07	53	3.57E+07	30	0.0000
3	(4,3,3,3,4,3,2)	5.16 E+07	5.16 E+07	138	5.19 E+07	41	0.0058
4	(2,4,3,4,6,3,2)	3.36E+07	3.36E+07	3×3600	3.40 E+07	27	0.0119
5	(3,3,4,6,8,4,2)	7.06 E+07	6.88 E+07	9×3600	7.35 E+07	89	0.0411
6	(3,3,4,6,9,4,4)	7.63 E+07	7.62E+07	3×3600	8.01 E+07	102	0.0498
7	(4,4,5,7,9,2,4)	1.167E+08	1.064E+08	3×3600	1.161 E+08	122	-0.0051
8	(3,4,5,7,10,5,3)	9.13E+07	8.90E+07	5×3600	9.35 E+07	150	0.0241
9	(5,4,6,7,10,5,3)	1.51E+08	1.39E+08	8×3600	1.498 E+08	301	-0.0079
10	(4,5,6,8,12,6,3)	1.49E+08	1.44E+08	6×3600	1.506 E+08	275	0.0107
11	(6,5,6,9,12,6,4)	-	1.858 E+08	-	2.001 E+08	407	-
12	(4,6,7,9,14,7,4)	1.53 E+08	1.42 E+08	8×3600	1.518 E+08	322	-0.0078
13	(5,5,8,10,15,7,5)	-	1.74 E+08	-	1.861 E+08	430	-
14	(7,5,8,10,15,7,7)	-	2.82 E+08	-	3.061 E+08	678	-
15	(5,6,9,11,18,8,5)	-	2.27 E+08	-	2.355 E+08	790	-

Also in the first phase, the population size is selected 100. The maximum generation is selected 100 for small instances and 200 for large instances.

4.3. Numerical results

The proposed algorithm is executed five times for each problem. The average values and LINGO results are shown in Table 5. A quality criterion, GAP, is defined to show the relative difference between the LINGO and proposed algorithm solutions. Let A and B denote the best objective value, using the LINGO and the average objective values of the proposed meta-heuristic algorithm, respectively. Now define the GAP as:

$$\text{GAP} = \frac{B - A}{R}. \tag{30}$$

The lower the value of this metric, the better the solution quality.

As shown in Table 5, the proposed algorithm finds the near optimal solutions in less computational time. In small size problems, the proposed algorithm has found optimal solution similar to LINGO (problems 1 and 2). But when the problem size increases, LINGO cannot find proper solution in reasonable time, while the proposed algorithm finds near-the-objective-bound solutions in less computational time. For the problems 5-10 and 12, LINGO cannot find the optimal solutions within 600 second, and instead of optimal solutions, the best feasible solution is given as comparison. In the large scale problems, LINGO cannot find any solution in an acceptable time span, but the proposed algorithm finds the solution near the objective bound of the problem in a reasonable time span (problems 11 and 13-15). The proposed algorithm in comparison with LINGO only finds slightly worse solutions in a few problems. GAP values do not exceed 5% for these problems. In the problems 7, 9 and 12, the proposed algorithm in comparison with LINGO finds better solution in shorter computational time (GAP < 0). In the large size problems, LINGO cannot find any solution in an acceptable time span, but the proposed algorithm finds the solution near the objective bound of the problem, in a reasonable time span.

For verification of the algorithm, appropriate statistical tests can be used to test the significant difference between two sets A and B, as follows:

$$\begin{cases} H_0: & \mu_A = \mu_B \\ H_1: & \mu_A \neq \mu_B \end{cases} \tag{31}$$

Nonparametric tests should be used due to the non-parametric characteristics of the data. Wilcoxon signed rank test is one of these tests that is used for paired comparisons. This test is employed using SPSS 16 software. The result is shown in Table 6. Significant level, 0.213, shows that there are no reasons for rejecting zero hypotheses with a 95% confidence level. So, there is no significant difference between the performance of the algorithm and using LINGO for solving the model.

The convergence speed of the first phase of the proposed algorithm is depicted in Figure 7. It is obvious that fitness function value decreases steeply and the proposed algorithm reaches the optimal value of the first phase after 25 generations.

Table 6. Result of Wilcoxon test on responses.

Ranks				Test statistics	
B − A	**N**	**Mean rank**	**Sum of ranks**		***B − A***
Negative ranks	3	4.00	12.00	Z	-1.245 [a]
Positive ranks	6	5.50	33.00		
Ties	2			Asymp. Sig. (2-tailed)	0.213
Total	11				

[a]: Based on negative ranks.

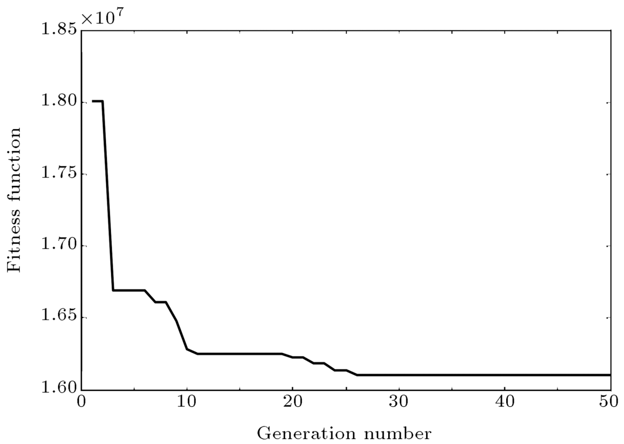

Figure 7. Convergence curve of the proposed algorithm.

5. Conclusions

In this paper, a designing and transportation planning in a multi-stage multi-product supply chain network is examined. Decision makers need to determine the optimal routes and vehicles when there is a limited budget for hiring vehicles. We considered the Integrated Forward/Reverse Logistics Network (IFRLN) problem and formulated it as a Mixed Integer Nonlinear Programming model (MINLP) to minimize the total costs of purchasing the products, hiring the vehicles and transportation.

The problem is NP-hard, so we developed a hybrid meta-heuristic algorithm based on pb-GA and SA algorithm in two phases to find optimal solution. The solution is represented by a matrix and a vector. In the matrix, the position of each cell represents the sources and depots, and the value in cells represents the priorities. Each row of the matrix is corresponding to a product type. In the assignment vector, the assigned vehicles are represented to carry products between the sources and depots. The algorithm is composed of two phases. In the first phase, the amount of products to be carried between the sources and depots are determined. Then, in the second phase, the vehicles for transporting products are determined.

Response Surface Methodology (RSM) was used to set the effective parameters of the algorithms. Several problems were generated and solved with LINGO optimization software and the proposed meta-heuristic algorithm. The results showed that the proposed algorithm can find the solution in less computational time. The results of Wilcoxon test showed that there is no significant difference between the objective function values of the algorithm and using LINGO for solving the problem.

For future research, other objectives can be used in this logistic network. Scheduling problems can be considered. Satisfying customers' demands on time will increase service level of the supply chain. The network responsiveness also can be used to satisfy the customers. Holding cost can be added to the objective function, and minimized.

GRASP (Gready Randomized Adaptive Search Procedure) algorithm can be used to generate the initial population, instead of generating it randomly. So, the algorithm will have three phases. Other metaheuristic algorithms can be developed to solve the problem, and then the algorithms can be compared from convergence to the optimal solution. In the second phase, other neighborhood search algorithms like local search and tabu search can be applied.

References

1. Hitchcock, F.L. "The distribution of a product from several sources to numerous localities", *Journal of Math. Phys.,* **20**, pp. 224-230 (1941).
2. Tilanus, B. "Introduction to information system in logistics and transportation", In: Tilanus B, Ed., *Information Systems in Logistics and Transportation,* Elsevier, Amsterdam, pp. 7-16 (1997).
3. Fleischmann, M., Beullens, P., Bloemhof-ruwaard, J.M. and Wassenhove, L. "The impact of product recovery on logistics network design", *Production and Operations Management,* **10**, pp. 156-173 (2001).
4. Lee, D.H. and Dong, M. "Dynamic network design for reverse logistics operations under uncertainty", *Transportation Research Part E,* **45**, pp. 61-71 (2009).
5. Pishvaee, M.S., Farahani, R.Z. and Dulaert, W. "A memetic algorithm for bi-objective integrated forward/ reverse logistics network design", *Computer & Operation Research*, **37**, pp. 1100-1112 (2010).
6. Farahani, R.Z. and Elahipanah, M. "A genetic algorithm to optimize the total cost and service level

for JIT distribution in a supply chain", *International Journal of Production Economics,* **111**, pp. 229-243 (2008).

7. Zegordi, S.H. and Beheshti Nia, M.A. "A multi-population genetic algorithm for transportation scheduling", *Transportation Research Part E*, **45**, pp. 946-959 (2009).
8. Michalewicz, Z., Vignaux, G.A. and Hobbs, M. "A non-standard genetic algorithm for the nonlinear transportation problem", *ORSA Journal on Computing,* **3**(4), pp. 307-316 (1991).
9. Li, Y.Z., Gen, M. and Ida, K. "Improved genetic algorithm for solving multi-objective solid transportation problem with fuzzy number", *International Journal of Fuzzy Theory Systems,* **4**(3), pp. 220-229 (1998).
10. Gen, M. and Cheng, R.W., *Genetic Algorithms and Engineering Optimization*, Wiley, New York (2000).
11. Gen, M., Altiparmak, F. and Lin, L. "A genetic algorithm for two-stage transportation problem using priority-based encoding", *OR Spectrum,* **28**, pp. 337-354 (2006).
12. Lee, J.E., Gen, M. and Rhee, K.G. "Network model and optimization of reverse logistics by hybrid genetic algorithm", *Computers & Industrial Engineering*, **56**, pp. 951-964 (2009).
13. Lin, L., Gen, M. and Wang, X. "Integrated multistage logistics network design by using hybrid evolutionary algorithm", *Computers & Industrial Engineering*, **56**, pp. 854-873 (2009).
14. Gunaraj, V. and Murugan, N. "Application of response surface methodology for predicting weld bead quality in submerged arc welding of pipes", *Journal of Materials Processing Technology*, **88**, pp. 266-275 (1999).

Biographies

Esmaeil Mehdizadeh is currently Assistant Professor at the Department of Industrial Engineering, Islamic Azad University, Qazvin Branch, Iran. He received his PhD degree in Industrial Engineering from Islamic Azad University, Science and Research Branch, Tehran, in 2009. His research interests are in the areas of operation research such as production planning, scheduling and meta-heuristic methods. He has several papers in journals and conference proceedings.

Fariborz Afrabandpei is currently Production Planner in Ilam Gas Treating Company, National Iranian Gas Company, Ilam, Iran. He received his MSc degree in Industrial Engineering from Industrial Engineering Department, Islamic Azad University, Qazvin Branch, Iran in 2012. His research interests are in the areas of operation researches such as production planning and scheduling. He has studied supply chain management and bi-objective methods.

Somayeh Mohaselafshar received her BS degree in Applied Mathematics from Tabriz University, Tabriz, Iran and her MSc degree in Industrial Engineering from Mazandaran University of Science and Technology, Babol, Iran in 2011. Her research interests are in the areas of operation research such as production planning, scheduling and meta-heuristic methods. She has studied supply chain management, manufacturing cells, job scheduling on machines and bi-objective methods. She is a member of Tabriz municipality.

Behrouz Afshar-Nadjafi is currently Assistant Professor and Graduate Programs Manager of Industrial Engineering at Islamic Azad University, Qazvin Branch, Qazvin, Iran. He received his PhD degree in Industrial Engineering at Sharif University of Technology, in 2008. His research interests are the project scheduling, Inventory and production planning. Currently, he is working on modeling and solution methods including exact procedures, meta-heuristic algorithms and artificial intelligence techniques regarding discrete optimization problems in reality.