Design of a Mathematical Model for Logistic Network in a Multi-Stage Multi-Product Supply Chain Network and Developing a Metaheuristic Algorithm

Esmaeil Mehdizadeh[a,*], Fariborz Afrabandpei[a,b]

[a] *Assistant Professor, Faculty of Industrial and Mechanical Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran*

[b] *MSc, Ilam Gas Treating Company, National Iranian Gas Company, Ilam, Iran*

Received 10 August, 2011; Revised 19January, 2012; Accepted 20 February, 2012

Abstract

Logistic network design is one of the most important strategic decisions in supply chain management that has recently attracted the attention of many researchers. Transportation network design is then one of the most important fields of logistic network. This study is concerned with designing a multi-stage and multi-product logistic network. At first, a mixed integer nonlinear programming model (MINLP) is formulated that minimizes transportation and holding costs. Then, a hybrid priority-based Genetic Algorithm (pb-GA) and simulated annealing algorithm (SA) is developed in two phases to find the optimal solution. The solution is represented by a matrix and a vector. Response Surface Methodology (RSM) is also used to adjust the significant parameters of the algorithm. Finally, several test problems are generated which show that the proposed metaheuristic algorithm can find good solutions in reasonable time spans.

*Key words:*Transportation network; Supply chain management; metaheuristic algorithms; Priority-based Genetic Algorithm.

1. Introduction

Logistics network design is one of the most important problems in supply chain management. Transportation network design was proposed by Hitchcock (1941). The objective was to find the way of transporting products from several sources to several destinations to minimize the total cost. Tilanus (1997) defined the logistic as the art of bringing the right amount of right products to the right place at the right time so that efficiency of the supply chain could be considerable.

Many researchers worked on logistics network design. Park (2005) proposed the solutions for integrated production and distribution planning and investigated the effectiveness of their integration in a multi-plant, multi-retailer logistic environment where the objective was to maximize the total net profit. In another study, formulating the problem by a mixed integer programming (MIP) model, Lejeune (2006) minimized costs in a three-stage supply chain including supplier, production and distribution centers. Liang (2008) developed a linear programming model for integrated production-transportation planning problems in a supply chain that minimized total production and transportation costs and the total number of return products. Fahimnia et al. (2011) considered Aggregate Production and Distribution Plans and developed a MINLP formulation for a two-echelon supply chain network.

Ryu et al. (2004) focused on integrating production and distribution problems in a supply chain where resources were common between the plants for minimizing the production, transportation and warehousing costs. Jo et al. (2007) employed a genetic algorithm (GA) for solving the nonlinear fixed charge transportation problem in two-stage supply chain networks. Karabuk (2007) described a transportation problem in a textile manufacture, which involved scheduling of pickup and delivery of daily inventory movement between plants. Farahani and Elahipanah (2008) 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 used a hybrid non-dominate sorting genetic algorithm (NSGA) to solve the problem. Zegordi and Beheshtinia (2009) considered production and transportation scheduling in a two-stage supply chain environment composed of m suppliers in the first stage and l vehicles in the second stage. The goal was to minimize the total tardiness and total deviations of assigned work loads of suppliers from their quotas. The researchers formulated the problem as a MIP problem and proposed an algorithm, namely, the multi-society genetic algorithm.

Since the majority of the problems mentioned can be categorized as NP-hard, many exact, heuristic and

[*] Corresponding author E-mail address: emehdi@qiau.ac.ir

metaheuristic methods have been developed for solving them. Recently, GAs have received considerable attention as an approach to optimization problems. They are greatly used for optimizing logistic network problems, and different ways of chromosome representation are proposed in each one. Michalewicz et al. (1991) developed a nonlinear transportation problem and solved it by a non-standard genetic algorithm approach. They used matrix representation to construct a chromosome and developed the matrix-based crossover and mutation. Similarly, Li et al. (1998) considered a multi-objective solid transportation problem and solved it using GA by using a three-dimensional matrix to represent the chromosome. Gen et al. (2006) considered the characteristics of ts TP, pb-GA with new decoding and encoding procedures. In their approach, solutions are encoded as arrays in which the position of each cell represents the sources and depots, and the values in cells show the priorities. Moreover ,the researchers proposed a new crossover operator called Weight Mapping Crossover (WMX) and carried out an experimental study in two stages. The pb-GA was also used in a study by Pishvaee et al. (2010). They used segment-based crossovers. In another study, an extended pb-GA named Ep-GA was used by Lin et al. (2009) who represented the chromosome with two sections. In the first section, priorities are shown and in the second section, guiding information about assigning retailers and costumers are provided. The researchers proposed a hybrid evolutionary algorithm based on Ep-GA, combined a local search (LS) technique and proposed a new fuzzy logic control (FLC) to enhance the search ability of EA.

Many factors affect the efficiency of logistic networks. One of them is determining vehicles to be used for carrying products. The kind of vehicles that we use for moving products can play a key role in cost reduction. Vehicles should be selected in such a way that retailers' demands can be satisfied with the minimum transportation cost considering capacity and a limited number of vehicles. Thus, in the present study, in addition to the unit transportation cost based on transportation distances, the cost of using vehicles is considered. In this regard, the capacity of the vehicles and their limited number are also taken into account. Furthermore, we extend the multi-stage Transportation Problem (msTP) that includes multi-product cases.

The multi-stage logistic network considered in this paper consists of three stages: supplier, wholesaler and retailer locations. The problem deals with determining the optimal transportation network in order to satisfy the retailer demands of several products by using several kinds of vehicles with the minimum cost. It's assumed that there are m vehicle types for transportation with a limited budget for purchasing or hiring them. The capacity of vehicles and their fixed travel cost are taken into consideration too. The aim is to satisfy the demands of retailers for p products with minimum costs.

To this end, firstly the problem is defined with a mixed integer non-linear programming model (MINLP) 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 cases and is combined with the simulated annealing algorithm (SA) to solve the problem.

The remainder of this article is 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 suggests some areas for future research.

2. Problem Description and Formulation

The logistics network discussed in this paper is a three-stage logistics network including supplier, wholesaler and retailer. In the network (Figure 1), new products are shipped from the supplier centers to the retailers through direct (shipped directly from the supplier centers to the retailers) and indirect (shipped from the supplier centers to the wholesalers and then to the retailers) routes to meet the demand of each retailer.

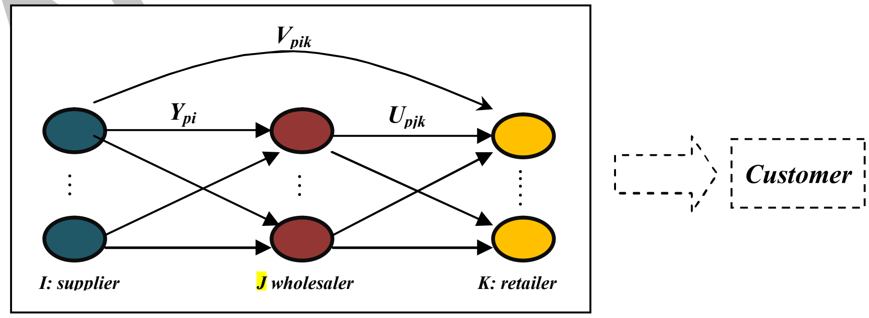


Fig. 1. Structure of the logistic network

In this section, a mathematical formulation for the problem is presented. The model has an objective function that minimizes the total transportation and holding costs. Transportation costs are based on the transportation cost of the products and using vehicles to carry products. The capacity of the sources and depots, the capacity of the vehicles, and the limited number of the vehicles are considered in this network. In terms of the above-mentioned notations, the logistic network design problem can be formulated as follows:

$$Minz = \sum_{i}\sum_{j}\sum_{p}\left(Y_{pij}c_{p}g_{1ij}\right) + \sum_{m}(pu_{m} + c_{1mij}(Y_{pij}|a_{mp}))B_{1mpij})$$

$$\sum_{i}\sum_{k}\sum_{p}\left(Y_{pik}c_{p}g_{2ik}\right) + \sum_{m}(pu_{m} + c_{2mik}(Y_{pik}|a_{mp}))B_{2mpik})$$

$$\sum_{j}\sum_{k}\sum_{p}\left(Y_{pjk}c_{p}g_{3jk}\right) + \sum_{m}(pu_{m} + c_{3mjk}(Y_{pjk}|a_{mp}))B_{3mpjk})$$

$$\sum_{p}\sum_{j}h_{pj}(\sum_{i}Y_{pij} - \sum_{k}U_{pjk}) \tag{1}$$

$$S.t. \quad \sum_{j}U_{pjk} + \sum_{i}V_{pik} = d_{pk}\ \forall k,p \tag{2}$$

$$\sum_{i}Y_{pij} \geq \sum_{k}U_{pjk}\ \forall p,j \tag{3}$$

$$\sum_{j}Y_{pij} + \sum_{k}U_{pik} \leq ca_{1ip}\ \forall i,p \tag{4}$$

$$\sum_{k}Y_{pjk} \leq ca_{2jp}\ \forall j,p \tag{5}$$

$$\sum_{i}Y_{pij} - \sum_{k}U_{pjk} \leq q_{jp}\ \forall p,j \tag{6}$$

$$pu_{m}\sum_{p}(\sum_{i}(\sum_{j}B_{1mpij} + \sum_{k}B_{2mpik}) + \sum_{j}\sum_{k}B_{3mpjk}) \leq b_{m}\ \forall m \tag{7}$$

$$Y_{pij} \leq \gamma\sum_{m}B_{1mpij}\ \forall i,j,p \tag{8}$$

$$V_{pik} \leq \gamma\sum_{m}B_{2mpik}\ \forall i,k,p \tag{9}$$

$$U_{pjk} \leq \gamma\sum_{m}B_{3mpjk}\ \forall j,k,p \tag{10}$$

$$\sum_{m}B_{1mpij} \leq 1 \qquad \forall i,j,p \tag{11}$$

$$\sum_{m}B_{2mpik} \leq 1 \qquad \forall i,k,p \tag{12}$$

$$\sum_{m}B_{3mpjk} \leq 1 \qquad \forall j,k,p \tag{13}$$

$$B_{1mpij}, B_{2mpik}, B_{3mpjk} \in \{0,1\} \qquad \forall i,j,k,m,p \tag{14}$$

$$Y_{pij}, U_{pjk}, V_{pik} \geq 0 \qquad \forall i,j,k,p \tag{15}$$

Where *I*: the number of suppliers (*i*= 1,…, *I*), *J*: the number of wholesalers (*j*= 1,…, *J*), *K*: the number of retailers (*k*= 1,…, *K*), *P*: the number of products (*p*= 1,…, *P*), *M*: the number of vehicles (*m*= 1,…, *M*), d_{pk}: amount of demand for product *p* by retailer *k*, b_m: maximum budget for purchasing or hiring a vehicle *m*, a_{mp}: capacity of vehicle *m* for transporting product *p*, pu_m: purchasing or hiring cost of vehicle *m*, h_{pj}: unit holding cost of product *p* in wholesaler *j*, c_p: unit transportation cost of product *p* along unit distance, ca_{1pi}: supply capacity of supplier *i* for product *p*, ca_{2pj}: delivery capacity of wholesaler *j* for product *p*, q_{pj}: holding capacity of whole sale r*j* for product *p*, g_{1ij}, g_{2ik}, g_{3jk}: distances between related sources and depots, c_{1mij}, c_{2mik}, c_{3mjk}: fixed cost of using vehicle *m* to carry products between the related sources and depots, Y_{pij}: amount of product *p* transported from supplier *i* to wholesaler *j*, U_{pjk}: amount of product *p* transported from wholesaler *j* to retailer *k*, V_{pik}: amount of product *p* transported from supplier *i* to retailer *k*, B_{1mpij}, B_{2mpik}, B_{3mpjk}: 1, if vehicle *m* is used to carry products between the related sources and depots, 0 otherwise and γ is a large number.

In the objective function (1), the first, second and third terms represent the transportation cost of products, the purchasing or hiring cost of vehicles and the travel cost of vehicles to carry products between the related sources and

depots. The forth term shows the holding cost in the wholesaler centers. Constraint (2) denotes the total amount of products that are sent to the retailer should be equal to their total demands. Constraint (3) assures the amount of products which are sent by each wholesaler to the retailers does not exceed the inventory of the warehouse. Constraints (4) and (5) are capacity constraints. Constraint (6) guarantees that the difference between incoming and outgoing products is confined to the holding capacity of the warehouse. Constraint (7) represents the budget constraints for purchasing or hiring vehicles. Constraints (8)−(10) enforce that there should be at least one kind of vehicle to carry products. Constraints (11)−(13) require that for each path and each product, only one kind of vehicle should be used. Constraints (14)−(15) place a binary and non-negativity restriction on the corresponding decision variables.
Defining new variables and adding some extra constraints as follows, non-linearity of the model can be eliminated.

$$Y_{pij}.B_{1mpij} = Z_{1mpij} \tag{16}$$

$$Z_{1mpij} \geq Y_{pij} - \gamma.\left(1 - B_{1mpij}\right) \tag{17}$$

$$Z_{1mpij} \leq Y_{pij} + \gamma.\left(1 - B_{1mpij}\right) \tag{18}$$

$$V_{pik}.B_{2mpik} = Z_{2mpik} \tag{19}$$

$$Z_{2mpik} \geq V_{pik} - \gamma.(1 - B_{2mpik}) \tag{20}$$

$$Z_{2mpik} \leq V_{pik} + \gamma.(1 - B_{2mpik}) \tag{21}$$

$$U_{pjk}.B_{3mpjk} = Z_{3mpjk} \tag{22}$$

$$Z_{3mpjk} \geq U_{pjk} - \gamma.(1 - B_{3mpjk}) \tag{23}$$

$$Z_{3mpjk} \leq U_{pjk} + \gamma.(1 - B_{3mpjk}) \tag{24}$$

3. Solution Approach

Although the exact algorithms find the optimal solution, the problems with real size are time-consuming. The metaheuristic algorithms are, therefore, used to find the near optimal solution in a reasonable time span. In this section, first the chromosome representation is described, and then a metaheuristic algorithm is proposed based on GA and SA to find the optimal solution in two phases. The evolutionary algorithms cause a lot of diversification in the solutions. Since the optimal routes are determined in the first phase, there should be a lot of diversification in the solutions. Consequently, it is better to use an evolutionary algorithm like GA in this phase. The neighborhood search algorithms can be used to improve the convergence speed to the optimal solution. In the second phase of the proposed algorithm, we prefer to use a neighborhood search algorithm like SA for its convergence speed.

3.1 Chromosome representation

In our problem, the solution is represented by a matrix and a vector. In the first one, the priority-based encoding method proposed by Gen et al. (2006) 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 in cells show the priorities. In the second one, 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 two segments. A typical example of the matrix is depicted in Figure 2, and the modified priority based decoding algorithm of a segment is shown in Figure 3.

To decode the chromosome, the second segment should be decoded before the first one. This involves determining the shipment from the supplier to the retailer or shipment from the wholesaler to the retailer. The demand of a retailer would be satisfied by a supplier or a wholesaler with the minimum cost. So the U_{pjk} and V_{pik} could be determined. Then the first segment should be decoded to determine Y_{pij}. In this segment, the demand of a wholesaler j for product p should be sent to the retailers

$$(b_{p,j} = \sum_{k} U_{pjk})$$

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

	First segment				***Second segment***						
	i		***j***		***i***		***j***		***k***		
	1	2	1	2	1	2	1	2	1	2	3
p=1	3	1	2	4	2	3	7	5	6	1	4
p=2	1	3	4	2	6	2	1	4	7	3	5

Fig. 2. An illustration of the chromosome with the modified priority-based encoding method

3.2 Operators

Crossover operator: in the matrix of priorities, segment-based crossover is used. In each row of parents, the first segments or the second segments swapped to generate offspring (Figure 5).

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

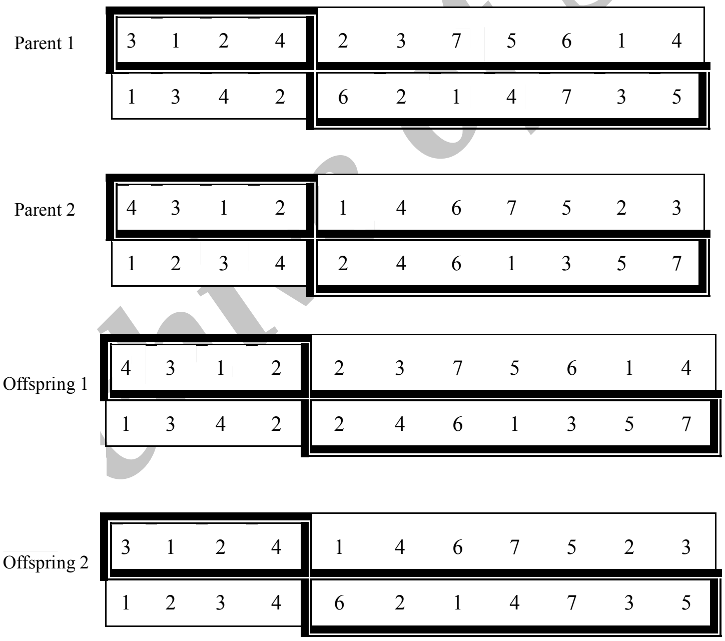
Neighborhood search: in the assignment vector, to find the neighboring solution of the current solution, some alleles are selected randomly from the assignment vector and their numbers are added to the available vehicles set. Then some numbers from this set are selected randomly and assigned to the routes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

```
Repeat
   P ←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 ←the source   k ←the depot
   $b_{p,k}$ ←demand of depot k for product p
   $ca_{p,j}$ ←capacity of source j for product p
   The amount of product p to be shipped between source j and depot k   ←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 all the priorities of product p were examined, then omit product p from available products.
   If all the products were examined, then exit.
End
```

Fig. 3. Decoding procedure for the modified priority based algorithm.

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

Fig. 4. The representation of the assignment vector



Parent 1

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

Parent 2

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

Offspring 1

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

Offspring 2

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

Fig. 5. An example of crossover operator

3.3 The proposed algorithm

The proposed algorithm consists of two phases. In the first phase, the optimal routes and amounts of products to satisfy demands 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 the GA. To generate an initial population, random numbers from 1 to each segment size are generated and represented in the segments of the matrices.

In the decoding procedure, the costs are determined by variable transportation costs without the vehicle costs. Besides, to evaluate the chromosomes, the fitness function is calculated without the vehicles costs. For each solution, if the number of 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_m].

The roulette wheel selection method is used to select parents. The crossover and mutation operators are used as mentioned. The termination condition reaches the maximum generation number and a feasible solution. Feasible solution is the solution that the number of its 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 next phase the kinds 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 is used to determine the optimal assignment of vehicles to the routes. To generate an initial solution, the available vehicles for transporting Y_{pij}, U_{pjk} and V_{pik} 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 the amounts of products determined in the first phase.

After determining the optimal fitness function in the first phase and the second phase, they are summarized to show the true fitness function. The proposed algorithm for the problem is summarized in Figure 6.

```
Phase 1:
  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 the initial population.
  Step 3:
  Decode chromosomes and determine $Y_{pij}$, $U_{pjk}$, $V_{pik}$, fitness values and number of routes for each chromosome.
  Select the best chromosome based on the fitness values.
  Step 4:
  Select a number of chromosomes for reproduction.
  Select a pair of parents by a pair of parents through the roulette wheel selection method.
  Apply the crossover operator tothe selected parents to generate offspring.
  Apply the mutation operator to the randomly selected chromosomes.
  Step 5:
  If $itr \leq maxitr$ or the solution is infeasible, itr=itr+1 and go to step 3.
  Otherwise, terminate.
Phase 2:
  Step 1:
  Get $T_0$, a andk; $T_c=T_0$; itr=1.
  Step 2:
  Generate an initial solution of the assignment vector and regard it as the 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 the probability of $e^{\frac{Fo-Fn}{Tc}}$.
  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:
  $Tc=a*Tc$.
  If $Tc > 0$, itr=1 and go to step 3 of this phase.
  Otherwise, terminate.
  At the end, summarize the fitness value of phase 1 and phase 2.
```

Fig. 6. The proposed algorithm for the logistic network

4. Computational Results

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

4.1 Data generation

We define 15 instances that can be characterized by the number of products (n_p) that are between 2 and 10, vehicles (n_m) that are between 3 and 9, suppliers (n_i) that are between 2 and 14, wholesalers (n_j) that are between 2 and 15 and retailers (n_k) that are between 3 and 30. The data required for the problem are generated randomly as shown in Table 1.

Table 1
Parameters' ranges in the test problems

Parameters	Small instances	Big instances
d_{pk}	[100 -220]	[100 -220]
b_m	[150,000 -700,000]	[700,000 -1,400,000]
pu_m	[10,000 - 45,000]	[10,000 - 45,000]
h_{pj}	[10,000 - 45,000]	[10,000 - 45,000]
q_{pj}	[500 - 700]	[500 - 700]
c_p	[50 - 100]	[50 - 100]
g_{1ij}	[100 - 200]	[100 - 200]
g_{2ik}	[150 - 400]	[150 - 400]
g_{3jk}	[70 - 150]	[70 - 150]
Ca_{1ip}	[800 - 1200]	[800 - 1200]
Ca_{2jp}	[500 - 700]	[500 - 700]
a_{mp}	[10 - 30]	[10 - 30]
C_{1mij}	[50 - 100]	[50 - 100]
C_{2mik}	[60 - 180]	[60 - 180]
C_{3mjk}	[80 - 150]	[80 - 150]

4.2 Parameters setting 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. It 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 (Gunaraj and Murugan, 1999). This method has four stages. In the first stage, the independent parameters and their levels are determined. Then 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 (equation 25). After collecting the data, the third stage involves predicting the model equation and obtaining the response surface as a function of the independent variables (parameters and their interactions). Then significant variables and their coefficients are found to determine the regression equation. Finally, in the fourth stage optimum points of the equation are determined.

$$RPD = \frac{Alg_{sol} - min_{sol}}{min_{sol}} \qquad (25)$$

The crossover rate (p_c), mutation rate (p_m), initial temperature(t_0), iterations in each 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 the parameters are defined as shown in Table 2.

Table 2
Levels of the proposed algorithm parameters

Parameter	Lower	Middle	Upper
p_c	0.5	0.6	0.7
p_m	0.1	0.15	0.2
t_0	10	25	30
k	110	130	150
a	0.94	0.96	0.98

The 2^{5-2} points using a two-level factorial design, 4 central points and 10 axial points are selected. For each point, each problem instance is carried out 5 times and the average value of the objective function values and the processing times are recorded. The results are normalized by the RPD criteria and the average values for each point are calculated. Two regression models of the objective function and processing time are determined. The two equations are optimized as a bi-objective problem using a bi-objective technique and LINGO9. Then the first equation is optimized and its solution is considered as a constraint on optimizing the second equation.

The results show that the models are 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.

In the first phase, 125is selected as the population size. The maximum generation is selected to be 100 for small instances and 500 for large instances.

4.3 Numerical results

The proposed algorithm is used five times for each problem. The average values and lingo results are presented in Table 3. A quality criterion, *GAP*, is defined to show the differences between the lingo and the proposed algorithm solutions. If *A* shows the best objective of the lingo and *B* shows the average values of the proposed metaheuristic algorithm, the *GAP* is defined as the following equation:

$$GAP = \frac{B-A}{B} \qquad (26)$$

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

For the problems 10−15, LINGO cannot find the optimal solutions within 4000 seconds and instead of optimal solutions, the best feasible solution is given for comparison. As shown in Table 3, the proposed algorithm finds the solutions near the optimal solutions in shorter computational times. The proposed algorithm finds only slightly worse solutions than LINGO in few problems. The *GAP* values do not exceed the 5% for these problems. In the problems 3, 7 and 9, compared with LINGO, the proposed algorithm finds the better solution in a shorter computational time (*GAP<0*). In the big 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.

Table 3
Results of the test problems of different sizes

		Lingo			Proposed metaheuristics		*GAP*
Problem	(n_p,n_m,n_i,n_j,n_k)	Best objective (*A*)	Objective bound	CPU Time (sec.)	Average value (*B*)	Average CPU Time (sec.)	
1	(2, 3, 2, 2, 3)	1.249E+07	1.249E+07	1	1.249E+07	2.3	0.0000
2	(2, 4, 3, 4, 4)	2.580E+07	2.580E+07	16	2.584E+07	10.1	0.0015
3	(3, 4, 3, 3, 6)	2.713E+07	2.713E+07	78	2.692E+07	14.4	-0.0078
4	(3, 3, 4, 6, 8)	5.087E+07	5.087E+07	248	5.403E+07	49.2	0.0584
5	(3, 3, 6, 7, 9)	6.053E+07	6.053E+07	476	6.273E+07	105.1	0.0350
6	(3, 4, 5, 7, 10)	6.892E+07	6.551E+07	792	7.092E+07	142.8	0.0282
7	(4, 4, 5, 7, 9)	8.552E+07	8.265E+07	1023	8.426 E+07	188.7	-0.0149
8	(4, 5, 6, 8, 12)	1.149E+08	1.117E+08	1892	1.170E+08	269.5	0.0179
9	(5, 4, 6, 7, 10)	1.419E+08	1.243E+08	2469	1.384E+08	354.2	-0.0253
10	(5, 5, 8, 10, 15)	-	1.584E+08	-	1.667E+08	402.3	-
11	(5, 6, 9, 11, 16)	-	1.851E+08	-	1.986 E+08	690.8	-
12	(6, 5, 9, 13, 17)	-	2.136E+08	-	2.153E+08	752.1	-
13	(7, 5, 10, 14, 19)	-	2.421E+08	-	2.512E+08	923.5	-
14	(8, 6, 11, 12, 25)	-	3.249E+08	-	3.346E+08	1357.3	-
15	(10, 9, 14, 15, 30)	-	3.852E+08	-	3.937E+08	2069.8	-

5. Conclusions

In this paper, 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. In this study, we formulated the problem as a mixed integer nonlinear programming model (MINLP) to minimize the total costs of transportation and holding products.

This problem is NP-hard, so we developed a hybrid metaheuristic algorithm based on the pb-GA and SA algorithm in two phases to find the 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, the value in cells show the priorities, and each row corresponds to a product type. In the assignment vector, the assigned vehicles to carry products between the sources and depots are represented. The algorithm consists 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 algorithm. Several problems were generated and solved with the LINGO optimization software and the proposed metaheuristic algorithm. The results showed that the proposed algorithm can find near optimal solutions in reasonable time spans.

Future researches may pursue other objectives with regard to this logistic network. For example, 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 costs can be added to the objective function and minimized.

About the algorithm, for generating the initial population, the GRASP* algorithm can be used instead of generating it randomly. So the algorithm will have the 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.

6. References

[1] Fahimnia, B., Luong, L. and Marian, R., (2011). Genetic algorithm optimization of an integrated aggregate

* Greedy Randomized Adaptive Search Procedure

production–distribution plan in supply chains. International Journal of Production Research, 1–16.

[2] Farahani, R. Z. and Elahipanah, M., (2008). A genetic algorithm to optimize the total cost and service level for JIT distribution in a supply chain. International Journal of Production Economics, 111, 229–243.

[3] Gen, M., Altiparmak, F. and Lin, L., (2006). A genetic algorithm for two-stage transportation problem using priority-based encoding. ORS. 28, 337–354.

[4] Gunaraj, V. and Murugan, N., (1999). Application of response surface methodology for predicting weld bead quality in submerged arc welding of pipes. Journal of Materials Processing Technology, 88, 266–275.

[5] Hitchcock, F.L., (1941). The distribution of a product from several sources to numerous localities.Journal of Math. Phys., 20,224–230.

[6] Jo, J.B., Li, Y.Z. and Gen, M., (2007). Nonlinear fixed charge transportation problem by spanning tree-based genetic algorithm. C.I.E., 53, 290–298.

[7] Karabuk, S., (2007). Modeling and optimizing transportation decisions in a manufacturing supply chain. Transportation Research Part E, 43 (4), 321–337.

[8] Lejeune, M.A., (2006). A variable neighborhood decomposition search method for supply chain management planning problems. European Journal of Operational Research, 175 (2), 959–976.

[9] Li, Y.Z., Gen, M. and Ida, K., (1998). Improved genetic algorithm for solving multi-objective solid transportation problem with fuzzy number. International journal of Fuzzy Theory Systems, 4(3), 220–229.

[10] Liang, T.F., (2008). Integrating production–transportation planning decision with fuzzy multiple goals in supply chains. International Journal of Production Research, 46, 1477–1494.

[11] Lin, L., Gen, M. and Wang, X., (2009). Integrated multistage logistics network design by using hybrid evolutionary algorithm. Computers& Industrial Engineering, 56, 854–873.

[12] Michalewicz, Z., Vignaux, G.A. and Hobbs, M., (1991). A non-standard genetic algorithm for the nonlinear transportation problem. ORSA Journal on Computing, 3 (4), 307–316.

[13] Park, Y.B., (2005). An integrated approach for production and distribution planning in supply chain management. International Journal of Production Research, 43, 1205–1224.

[14] Pishvaee, M.S., Farahani, R.Z. and Dulaert, W., (2010).A memetic algorithm for bi-objective integrated forward/reverse logistics network design. Computer & Operation Research, 37, 1100–1112.

[15] Ryu, J., Dua, V. and Pistikopoulos, E.N., (2004). A bi-level programming framework for enterprise-wide process networks under uncertainty. Computers and Chemical Engineering, 28 (6–7), 1121–1129.

[16] Tilanus, B., (1997). Introduction to information system in logistics and transportation. Elsevier, Amsterdam, 7–16.

[17] Zegordi, S.H. and BeheshtiNia, M.A., (2009). A multi-population genetic algorithm for transportation scheduling. T. Research Part E, 45, 946–959.