International Journal of Applied Operational Research
Vol. 7, No. 1, pp. 45-61, Winter 2017

Journal homepage: ijorlu.liau.ac.ir

A New Mathematical Formulation for Multi-product Green Capacitated Inventory Routing Problem in Perishable Products Distribution Considering Dissatisfaction of Customers

M. Rabbani*, A. Ordibazar, H. Farrokhi-Asl

Received: 28 April 2016 **;** **Accepted:** 17 September 2016

Abstract In this paper, we propose a new mathematical model for Capacitated Inventory Routing Problem (CIRP), which considers fresh delivery of perishable products to the customers' location; otherwise, a reduction in products' demand may occur. Therefore, we attempt to plan delivering the process of products at the right time to avoid extra inventory, causing expiration of products. This leads to the dissatisfaction of customers which causes decrease in their demands. Moreover, we address global concerns about environmental issues and impacts of transportation on environment, so we aim to minimize fuel costs to reduce costs and pollution related to the CO_2 emission. CIRPs is an NP-hard problem which needs a long run time for large-scale problems. To tackle this type of problem in an acceptable time, we introduce a hybrid algorithm combining Genetic Algorithm (GA) and Simulated Annealing (SA) Algorithm. Some experiments are conducted and the results show the efficiency of our proposed algorithm.

Keywords: Vehicle Routing Problem, Environmental Impacts, Perishable Goods, Dissatisfaction of Customers, Metaheuristic Algorithms.

1 Introduction

One of the most popular problems in supply chain management is Vehicle Routing Problem (VRP), which considers a supply chain network in which some nodes require some products, and their demand should be satisfied by products via suppliers. In fact, VRPs help consumers and suppliers to decide about the routing phase of products distribution, and fleet pf vehicles should travel in a way that demands become satisfied along with minimization of supply chain network's costs.

Additionally, in multi-period problems, another concept should be taken into account. In a multi-period problem, retailers can hold inventory by its corresponding costs. Therefore, it is

* **Corresponding Author. (✉)**
E-mail: mrabani@ut.ac.ir (M.Rabbani)

M. Rabbani
Department of Industrial & Systems Engineering, College of Engineering, University of Tehran, Tehran, Iran, P.O.Box: 11155-4563.

A. Ordibazar
Department of Industrial & Systems Engineering, College of Engineering, University of Tehran, Tehran, Iran

H. Farrokhi-Asl
Department of Industrial Engineering, Iran University of Science & Technology, Tehran, Iran

[Downloaded from mail.ijaor.ir on 2026-11-28]

not necessary for the products to be delivered at each period of time, which decreases transportation costs but increases the inventory costs. This fact makes us to decide about a trade-off between distinct components of our network cost.

Moreover, demands inherently are not deterministic, so they may be different from their predicted amounts. As such, we face with another problem, named lost sale and back order. For some products, which have efficient age like dairy products, extra-unsold inventory are not useable forever because the aged products start to spoil gradually. This makes suppliers to discard these products which have its own costs. Therefore, we should try to minimize lost sale and inventory costs. Also decreasing of demands shows a kind of dissatisfaction of customers. In other words, trying to reduce the age of perishable products will avoid satisfaction of customers.

Another discussion is concerned recently a lot is environmental considerations. Because of global conditions, different parts of industries and countries are trying to minimize environmental impacts of productions and transportation systems. Many environmental damages that have occurred are hazardous and dangerous for humans. So, recently many researchers focus on environment to engage green concepts in their decision mathematical models. As a result, we should try to minimize environmental costs of the presented network simultaneously.

In this paper, we propose a new mathematical model for multi-period, multi vehicle inventory routing problems for delivering different products with different demands to retailers. As aforementioned, we try to prepare a timetable and routing for each vehicle at each period in such a way that demands are met and lost sales cost be minimized. We consider perishable products whose age affects their demands. In addition, we address the environmental factors and try to minimize the cost of fuel consumption, which leads to minimize the use of fuel and has attractiveness for suppliers to decrease their total costs.

The rest of the paper consists of literature review in section 2. Then, we define the problem in section 3. Modeling approach is discussed in section 4. We propose some meta-heuristic approaches to solve the NP-hard model in section 5. Numerical results are represented in section 6. Finally, conclusions are stated in section 7.

2 Literature Review

Vehicle Routing Problem (VRP) is to schedule delivering products of consumers, which are stored in a central depot or depots where basic constraints should be satisfied. For instance, each retailer should be visited by a vehicle and no vehicle can remain in any retailer node [1]. Moreover, sub-tour elimination constraints should be noted; that is, travel starts at a depot and ends in the same depot. Sub-tour elimination constraint is proposed by [2]. Another kind of VRP is the open loop VRP where a vehicle starts at a depot, and there is not any necessity to end in a depot. Different mathematical models are proposed for different purposes to determine the optimal route and different applications of the problem [3]. Some researchers have considered open VRP for multi-period problem like [4].

VRP is proved to be in NP-hard class of problems that causes a long time and a remarkable memory to solve them [5]. Recently, to solve VRPs, many meta-heuristic algorithms have been proposed [6]. Some researchers have used the taboo search algorithm for the multi-depot VRP for capacitated retailers' inventory [7]. Some others have used the ant colony algorithm for VRPs by considering wastes [8]. Others have applied other well-

[Downloaded from mail.ijaor.ir on 2026-11-28]

known meta-heuristic algorithms like Genetic Algorithms, Particle Swarm Optimization, etc. to solve VRPs [6,9-11].

Attention to perishable products has been recently increased, and many researchers have proposed models to cover disadvantageous of wasting products. Some VRP models have been proposed to cover the age of products constraints [12]. Also, some others proposed models and investigates their applications in case studies [13-15].

Green logistics is one of the newest fields of supply chain management. Many researchers study it and it seems that in the near future it become more popular. Some researchers studied about trends and future of the field [16]. Some other focused on transportation as key parts of green logistics [17]. In addition, some studies have considered fuel consumption. More use of fuel leads to more costs and more air pollution. So by reducing fuel consumption both costs and environmental impact decreases [18].

In this paper, we consider a capacitated inventory Routing Problem for perishable products by considering fuel consumption and at last propose some meta-heuristic approach to solve the models.

3 Problem Definition

In this paper, we consider a Capacitated Inventory Routing Problem (CIRP) for perishable products by considering environmental aspects where a set of heterogeneous vehicles with different levels of technologies is used. As more a vehicle's level of technology, it has more transportation costs, which could be related to the kind of maintenance, should be done for it and less environmental costs which makes using of them reasonable. The network consists of a single depot and a set of different retailers. Each retailer has demands for some different products, which are predictable through historical data. As such, we assume that demands are deterministic which reasonably does not affect on generality and realism of the problem. As an aforementioned, products that are planning to be transferred are perishable. It means that as product's age increases, its usefulness decreases. Therefore, by passing time, demand of a product reduces from nominal demand. It shows dissatisfaction of customers. We propose a model to schedule an optimal timetable for a multi-period time horizon. All nominal demands should be met. Nevertheless, by passing time, demands may be considered as a lost sale that has a cost for unsold and useless products. Retailers can hold inventory for some days up to their volume capacity of inventory, which makes products aged and reduce their demands. Demands of products may decrease linearly based on [13]. In our proposed model, constraints like sub-tour elimination, i.e., each vehicle should start at a depot and finished its route in the same depot is representative for a closed loop supply chain. Each retailer should be serviced only by one vehicle and one time at each period. Our aim is preparing a timetable for all vehicles to satisfy all retailers' needs and in addition to transportation and inventory costs trying to reduce harmful environmental aspects of transportation and try to increase customer satisfaction by providing appropriate products that should utilities fresh.

4 Modeling Approach

To schedule the optimal timetable, we propose a mixed integer non-linear problem that satisfied all constraints and try to reduce monetary and environmental costs. The following

[Downloaded from mail.ijaor.ir on 2026-11-28]

model consists of some non-linear constraints which effects on solving time. Therefore, to overcome this problem we linearize them in the following subsection.

4.1 Nomenclature

To model this problem, we define the following notation contains sets, parameters and decision variables:

Index Sets:

I=(0,1, 2 …*i*)	Set of Retailers (I=0 shows the central Depot)
L=(1, 2 …*l*)	Set of Products
K= (1, 2… *k*)	Set of Vehicles
T= (1, 2… *t*)	Set of time periods

Parameters:

Q_k	Capacity of vehicle *k*
$DN_{i,l,t}$	Nominal Demand of retailer *i* for product l in period *t*
β	Rate of demand decreasing caused by age of product
CAP_i	Maximum Capacity of customer *i* to hold inventory
S_i	Time of serving for retailer *i*
$t_{i,j}$	Time to transport from node *i* to node *j*
LT	Duration of a period
$CT_{i,j,k}$	Cost of transportation from node *i* to node *j* by vehicle with level of technology *k*
$h_{i,l}$	Cost of holding inventory for product *l* in retailer *i*
$dis_{i,j}$	Distance from node *i* to node *j*
CF_{k}	Cost of fuel consumption per unit of distance for vehicle with level of technology *k*
ρ_0	Rate of loss for Back orders
α	Variable cost of fuel consumption
$CL_{\mathrm{i,l,k}}$	Cost of loading product *l* in retailer *i* which is served by vehicle with level of technology *k*
W_{l}	Weight of product *l*
$Price_l$	Price of product l
NI	Number of retailers
NL	Number of Products
M	A Big Number

[Downloaded from mail.ijaor.ir on 2026-11-28]

Decision Variables

X_{ijlkt} $\begin{cases} 1 \text{ if product } l \text{ is transfered from node } i \text{ to node } j \text{ by vehicle } k \text{ in time period } t \\ 0 \text{ Otherwise} \end{cases}$

XY_{ijkt} $\begin{cases} 1 \text{ if vehicle } k \text{ is transfered from node } i \text{ to node } j \text{ in time period } t \\ 0 \text{ Otherwise} \end{cases}$

Y_{ilkt} $\begin{cases} 1 \text{ if node } i \text{ is served by vehicle } k \text{ with product } l \text{ in time period } t \\ 0 \text{ Otherwise} \end{cases}$

V_{ikt} $\begin{cases} 1 \text{ if node } i \text{ is served by vehicle } k \text{ in time period } t \\ 0 \text{ Otherwise} \end{cases}$

W_{ilkt} Integer amount of product l that delivered to node *i* by vehicle *k* in time period *t*

U_{ilt} The dummy variable for sub tour elimination

ap_{ilt} Age of a product l for retailer *i* in period *t*

I^{+}_{ilt} Inventory of products *l* for retailer i in period *t*

I^{-}_{ilt} Lost sale of products *l* for retailer i in period *t*

Z_{ijlkt} Amount of product *l* which is transported from node *i* to node *j* in time period *t* by vehicle *k*

DE_{ilt} Effective demand of product *l* for retailer *i* in period *t*

4.2 Mathematical Model

Constraints of the model are as below:

$$\sum_{i,\, i\neq j} XY_{ijkt} = V_{jkt} \qquad \forall j,k,t \qquad (1)$$

$$\sum_{j,\, j\neq i} XY_{ijkt} = V_{ikt} \qquad \forall j,k,t \qquad (2)$$

$$\sum_{k} V_{ikt} \leq 1 \qquad \forall i,t,\ i\neq 0 \qquad (3)$$

$$\sum_{i\neq 0} V_{ikt} / NI \leq V_{0kt} \leq \mathrm{M}^{*} \sum_{i\neq 0} V_{ikt} \qquad \forall i,k,t \qquad (4)$$

$$\sum_{l} X_{ijlkt} \leq \mathrm{NL}^{*} XY_{ijkt} \qquad \forall i,j,k,t \qquad (5)$$

$$\sum_{i,\, i\neq j} X_{i,j,l,k,t} = Y_{j,l,k,t} \qquad \forall j,l,k,t \qquad (6)$$

$$\sum_{j,\, j\neq i} X_{i,j,l,k,t} = Y_{i,l,k,t} \qquad \forall i,l,k,t \qquad (7)$$

$$\sum_{l} Y_{i,l,k,t} / NL \leq V_{ikt} \leq \mathrm{M}^{*} \sum_{l} Y_{i,l,k,t} \qquad \forall i,k,t,\ i\neq 0 \qquad (8)$$

$$U_{i,l,t} - U_{j,l,t} + Q_k {}^{*} x_{i,j,l,k,t} \leq Q_k - D_{j,l,t} \qquad \forall i,j,l,k,t,\ i\neq j,\ i\neq 0,\ j\neq 0 \qquad (9)$$

[Downloaded from mail.ijaor.ir on 2026-11-28]

$$D_{i,l,t} \le U_{i,l,t} \le \sum_k Q_k \qquad \forall\, i,l,t\ ,i \neq 0 \qquad (10)$$

$$W_{ilkt} \le Q_k * Y_{i,l,k,t} \qquad \forall\, i,l,k,t \qquad (11)$$

$$\sum_{i\neq 0} Y_{i,l,k,t} / NI \le Y_{0,l,k,t} \le \mathrm{M}^* \sum_{i\neq 0} Y_{i,l,k,t} \qquad \forall\, l,k,t \qquad (12)$$

$$\sum_{i,l} W_{i,l,k,t} \le Q_k \qquad \forall k,t \qquad (13)$$

$$ap_{i,l,t} = (ap_{i,l,t-1} + 1).(1 - \sum_k Y_{i,l,k,t}) \qquad \forall\, i,l,t\ ,i \neq 0 \qquad (14)$$

$$DE_{ilt} = [DN_{i,l,t} - \beta * ap_{i,l,t}] \qquad \forall\, i,l,t\ ,i \neq 0 \qquad (15)$$

$$I^+_{i,l,t} - I^-_{i,l,t} = I^+_{i,l,t-1} - I^-_{i,l,t-1} + \sum_k W_{i,l,k,t} - DE_{ilt} \qquad \forall\, i,l,t \qquad (16)$$

$$\sum_l I_{i,l,t} \le CAP_i \qquad \forall\, i,t \qquad (17)$$

$$\sum_{i,j} S_i * x_{i,j,l,k,t} + \sum_{i,j} t_{i,j} * x_{i,j,l,k,t} \le LT \qquad \forall\, l,k,t \qquad (18)$$

$$\sum_{j\neq i} Z_{jilkt} - \sum_{j\neq i} Z_{ijlkt} = W_{ilkt} \qquad \forall\, i,l,k,t\ ,i \neq 0 \qquad (19)$$

$$Z_{ijlkt} \le Q_k * x_{i,j,l,k,t} \qquad \forall\, i,j,l,k,t \qquad (20)$$

$$\sum_{i,l} I^+_{i,l,T} + I^-_{i,l,T} = 0 \qquad T \text{ is the last period} \qquad (21)$$

As described before, XY_{ijkt} shows vehicle k moves from node i to node j or not. Equations 1 and 2 are basic constraints of classic VRP, which consider assignment of nodes to a tour traveled by a vehicle. These equations do not consider which products should be transferred. They only decide which vehicle should travel each route. Equation 3 helps the model to guarantee that in each solutions each retailer be visited only by one vehicle. Equation 4 ensures that vehicle k moves from depot if it should visits even one retailer. Equation 5 obliges vehicle k move from node i to node j if even one product should transport to node j from node i. Equations 6 and 7 make variable $Y_{i,l,k,t}$ to get value. Equation 8 decides that vehicle k visits retailer i and all products should be delivered to retailer i by vehicle k. Equations 9 and 10 consider sub-tour elimination to avoid travelling a vehicle in a network of retailers without central depot. Equation 11 also makes variable $Y_{i,l,k,t}$ to get value based on amount of products should be delivered to retailers. Equation 12 decides usage of vehicle k in depot. Equation 13 considers capacity of vehicles to avoid over loading a vehicle in the schedule. Equation 14 calculates age of products, which are holds in the stock. Equation 15 represents reduction of demands because of age of the products, which is named effective demand that may have differences from nominal demand that included as parameter to the model. Equation 16 is equation of inventory, which considers back order. Capacity of holding inventory for each retailer is calculated in Equation 17. Time service of each period is addressed in Equation 18. Equations 19 and 20 show weight of carried load from each node to another which will be used to calculate amount of fuel consumption with respect to [21]. Fuel consumption affects on transportation costs and air pollution and relates to amount of carried load for each vehicle. Equation 21 controls inventory level at the last period.

[Downloaded from mail.ijaor.ir on 2026-11-28]

4.3 Model Linearization

As mentioned, proposed model have nonlinear constraints. Equations 14 and 15 are non-linear. To linearize Equation 14 we use from an approach based on [12] as bellow:

In Equation 14, we define an integer variable $SC_{i,l,t}$ as bellow and a big number M:

$$SC_{i,l,t} = ap_{i,l,t-1} * \sum_{k} Y_{i,l,k,t} \qquad \forall\, i,l,t \qquad (22)$$

$$ap_{i,l,t} = ap_{i,l,t-1} - SC_{i,l,t} - \sum_{k} Y_{i,l,k,t} + 1 \qquad \forall\, i,l,t \qquad (23)$$

$$SC_{i,l,t} \le ap_{i,l,t-1} \qquad \forall\, i,l,t \qquad (24)$$

$$SC_{i,l,t} \le M * \sum_{k} Y_{i,l,k,t} \qquad \forall\, i,l,t \qquad (25)$$

$$SC_{i,l,t} \ge ap_{i,l,t-1} + M * (\sum_{k} Y_{i,l,k,t} - 1) \qquad \forall\, i,l,t \qquad (26)$$

Finally, Equation 15which contains a floor, because of that $d_{i,l,t}$ is integer, linearizes as below and ϵ is a very small number to avoid mistakes.

$$d_{i,l,t} \le D_{i,l,t} - \beta * ap_{i,l,t} \qquad \forall\, i,l,t \qquad (27)$$

$$d_{i,l,t} \ge D_{i,l,t} - \beta * ap_{i,l,t} - 1 + \epsilon \qquad \forall\, i,l,t \qquad (28)$$

4.4 Objective Functions

In this paper, we try to optimize four objective functions, simultaneously. We want to minimize all of them:

$$Z_1 = \sum_{i,j,l,k,t} C_{i,j} * x_{i,j,l,k,t} + \sum_{i,l,k,t} CL_{\mathrm{i,l,k}} * Y_{i,l,k,t} \qquad (29)$$

$$Z_2 = \sum_{i \ne 0,l,t} h_{i,l} * I_{ilt}^{+} + \rho_0 * Price_l * I_{ilt}^{-} \qquad (30)$$

$$Z_3 = \sum_{i \ne 0,l,t} Price_l * (DN_{i,l,t} - DE_{i,l,t}) \qquad (31)$$

$$Z_4 = \sum_{i,j,l,k,t} \alpha * CF_{\mathrm{k}} * d_{i,j} * W_1 * Z_{ijlkt} \qquad (32)$$

Equation 29 tries to minimize transportation costs between all nodes that vehicle fleet traveled and loading cost of each product in each retailer. Equation 30 calculates cost of holding inventories and damage of back orders gained by considering a Coefficient multiply to price of products. Coefficient ρ_0 is used because the customer, which is lost, may be come back. Equation 31 calculates reduction of effective demand from nominal demand and it has damage as same as its price which is not come back and represents cost of dissatisfaction of customers. Equation 32 tries to calculate and minimize cost of fuel consumption, which is based on amount of transportation between two nodes based on [18]. By minimizing cost of fuel consumption, which has attractiveness for suppliers, consumption of fuel is decreased too.

[Downloaded from mail.ijaor.ir on 2026-11-28]

4.5 Multi-objective Optimization

To transform multi-objective optimization problem into single objective one, we use Torabi-Hassini method [19]. Objective function of this method has two components. Before describing two components, we should notice that to normalize objective functions we use fuzzy-normalization approach that normalizes objective functions as Equation (33). To find PIS (Positive Ideal Solution) of Z_t we optimized problem just for Z_t without considering other objective functions. This value is minimum possible amount for this objective function. While calculating PIS for each objective function, we calculate amounts of other objective functions that have near optimum amount. NIS (Negative Ideal Solution) is the best of near optimum amounts.

$$\mathrm{N_t} = \frac{PIS_t - \mathrm{Z_t}}{PIS_t - NIS_t} \tag{33}$$

The first component calculates maximum amount of normalized objective functions as lambda in Equation (34). Minimizing Lambda gives a solution for the problem as gained by Min-Max method.

$$Lambda = Max \tag{34}$$

The second component calculates weighted summation of normalized objective functions. It should be noted that summation of weights of objective functions should be one as Equation (35). How much weight of an objective function is greater it considered more important.

$$\sum_{t=1}^{4} w_t = 1 \tag{35}$$

Equation (36) shows Torabi-Hassini multi objective method. Parameter V is an amount between 0 and 1. If V=1 the approach leads to Min-Max method and If V=0 it changes to weighted-sum approach. Therefore, an amount between 0 and 1 leads to a good solution which is calculated something between weighted sum and Min-Max method. In addition, objective function should be subject to system constraints as Equation (37).

$$Min\ F = V * Lambda + (1 - V) * \sum_t w_t * N_t(x) \tag{36}$$

s.t.

$$\mathrm{x} \in \mathrm{X} \tag{37}$$

$$V \in [0,1] \tag{38}$$

5 Solution Approach

The proposed model, as discussed is a kind of VRP problems. VRP problems are NP-hard [5, 20,21]. Therefore, by increasing in number of problem parameters, especially number of periods, computational complexity increases exponentially. So time of solving remarkably is affected by size of the problem. In fact, it is not solvable in large-scale size by exact algorithms. To overcome this problem, we proposed a meta-heuristic algorithm that can solve the model in an accepted time even in large-scale size of the problem.

[Downloaded from mail.ijaor.ir on 2026-11-28]

As described, our aim in this paper is preparing a schedule for vehicle fleet to transfer specific amount of products among different retailers (W_{ilkt}). Therefore, the important decision variable in our model is amount of each product, their destination and the vehicles, which should transfer them. Other decision variables and concepts of the model can be retrieved from this variable. The binary variable, which shows that if a vehicle visits a retailer for a product in a time period can be (Y_{ilkt}) understood from W_{ilkt}. If W_{ilkt} is not zero then Y_{ilkt} is either. Other variables like amount of inventory, age of products, effective demands, lost sale (because of decreasing demand for age of product) etc. can be easily calculated from W_{ilkt}. Each meta-heuristic algorithm has two general aspects, First solution and fitness function. Here we describe about our first solution and fitness function. Then in the next part, we will discuss about our algorithm.

5.1 Construction Algorithm

Our initial solutions for this model is a two dimensional table contains amount of W_{ilkt} for all consumers in all periods. The first dimension is $I \times L$ length that considers delivery of all products for each consumer. The second dimension is $K \times T$ length, which considers use of each vehicle for all periods. This type of table can help us to calculate other variables and constraints easier in the fitness function. Because each column of horizontal axis presents schedule of each vehicle in a specific period and routing can be easily retrieved by specifying consumers that the vehicle visits them in the period, which their W_{ilkt} is not zero. In addition, each row of vertical axis presents delivery of each product for each consumer and satisfying demands of each consumer for products can be easily calculated. First Solution table is shown in Figure 1:

		Day 1			Day 2				Day T		
		Vehicle 1	. . .	Vehicle K	Vehicle 1	. . . .	Vehicle K		Vehicle 1	. . . .	Vehicle K
Consumer 1	Product 1	$W_{1,1,1,1}$	. . .	$W_{1,1,K,1}$	.	. . . .	.		$W_{1,1,1,T}$	. . . .	$W_{1,1,K,T}$
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	Product L	$W_{1,l,1,1}$	. . .	$W_{1,l,K,1}$	.	. . . .	.		$W_{1,l,1,T}$	. . . .	$W_{1,l,K,T}$
Consumer 2	Product 1	$W_{2,1,1,1}$	. . .	$W_{2,1,K,1}$	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	Product L	$W_{2,l,1,1}$	. . .	$W_{2,l,K,1}$	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
Consumer I	Product 1	$W_{I,1,1,1}$	. . .	.		. . . .			$W_{I,1,1,T}$	. . . .	$W_{I,1,K,T}$
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	.	.	. . .	.	.	. . . .	.		.	. . . .	.
	Product L	$W_{I,l,1,1}$	. . .	.		. . . .			$W_{I,l,1,T}$	. . . .	$W_{I,L,K,T}$

Fig. 1 A solution representation for the problem

[Downloaded from mail.ijaor.ir on 2026-11-28]

The first table is created by generating random numbers, which will be refined by algorithm's operators will be presented in the following sections. To achieve a quality solution near optimal, we choose a random numbers from {0, co* $D_{i,l,t}$} for eachW_{ilkt} , also co is a coefficient in which random numbers can get amount more than $D_{i,l,t}$ and it is about 2 or 3 based on the problem so sometimes nominal demands will be satisfied and probable extra products will remain in the inventory warehouses and sometimes nothing deliver to the retailer and consumer should use its inventory from previous periods. In addition, every period only one vehicle gets amount for each retailer to satisfy the corresponding constraint that each consumer gets all its services once a period from only one vehicle not more. Therefore, the First Solution has some amounts near optimal solution and helps that algorithm works better. Algorithm's operators will change the First Solution and generate new solutions. Feasibility and Optimality of each solution will be evaluated by fitness function that will be defined in the following sections, weak solutions will be discarded, and better solutions will be accepted.

5.2 Fitness Function

As mentioned, feasibility and optimality of the solutions should be checked in the fitness function. Design of the first solution as a two dimension table helps us to calculate different variables. One of the most important variables is $x_{i,j,l,k,t}$, which defines amount of transportation between two nodes for calculating the first objective function. Moreover, for the fourth objective function, we need to know about the routing each vehicle use each period.

The routing of each vehicle can be a permutation of retailers that the vehicle visits in a period (retailers that have amount of W_{ilkt} more than zero). We put depot at the both sides of each permutation, which easily considers sub-tour elimination constraint, because each route starts at depot and finished to the depot. Then, we calculate cost of transportation for each permutation by calculating costs between sequent nodes. At last, the best route with lowest transportation cost is chosen from different permutations by the probability of 50 percent which is a greedy answer that can lead us to the best optimal solution. To make diversity in generated solutions in other 50 percent of states, we choose one of the routes randomly. Therefore, objective functions included transportation cost and environmental cost can be calculated from chosen routing and W_{ilkt}.

Amount of Inventory (I^{+}_{ilt}) and Back order (I^{-}_{ilt}) can be easily calculated by the amounts of $W_{i,l,k,t}$ in sequent periods. Which can be easily understood by checking each row that shows amount of delivery of each retailer for each product. In addition, other variables like age of product are computed easily. Therefore, we calculate inventory cost and waste of inventory cost as second and third objective function. To calculate effective demand, age of products is calculated by means of timetable of$W_{i,l,k,t}$. And decreasing of the demand is calculated based on age of products and it will be used to calculate objective function Equation (31).

To consider constraints, we use a Lagrangian approach to give big amount of penalty to constraints that are not satisfied. Then, we compute amount of penalties and objective functions and choose from lowest amounts. In this way, algorithm leads to feasible solutions and then tries to make it better and better.

[Downloaded from mail.ijaor.ir on 2026-11-28]

Equations 13, 17, 18 and 21 get penalties for not satisfying and are added to the total cost function. Equations (1-12) are satisfied in the routing and first table as described before. Equations (14-16) are calculated for objective function 2 and 3. Equation 19 and 20 are calculated for fourth objective function.

The first Solution provides schedule of transportations for each vehicle. On the other hand, fitness function checks amounts of objective functions. In addition, Fitness Function provides routing of each vehicle for each periods, which is one of the target of the model.

5.3 Genetic Algorithm (GA)

Genetic Algorithm (GA) is one of the population-based Evolutionary algorithms used broadly in Operations Research and computer science. In addition, it is a well-known algorithm to solve the complex problems. GA initializes by a population of chromosomes and inspired from nature operators of Cross Over and Mutation to generate new generations. In each Generation, strong solutions remains and weak solutions discard. So, after some iterations solutions lead to optimal solution. Here we generate a number First Solution, which described in Section 5.1 and evaluate by Fitness Function described in Section 5.2 then better solutions by minimum amounts of Fitness Function remain as strong Solutions. Each new generation produces by operators of GA.

Each chromosome consists of some genes. In our paper, retailers are genes of our chromosome. Therefore, our chromosome consists of some timetable presenting amounts of $WI_{i,l,t}$ for each retailer for all products through all periods. Therefore, each member of our population in GA is like what is shown in Figure 1.Other parameters of GA are consists of bellow concepts:

nPop	Number of chromosomes in each population
nc	Cross over percentage of nPop
nm	Mutation percentage of nPop
iter	Number of iteration of algorithm

5.3.1 Cross Over

In each cross over, Algorithm selects two chromosome randomly or by a probability amount then combines them to produce two new solutions. There are 3 general kinds of Cross Over: Single Point Cross Over, Double Point Cross Over and Uniform Cross Over.

Single Point Cross Over: In this kind of Cross Over, every selected solutions called parents which have some genes. A single random number is generated and from that single point parents combine to create new children as Figure 2 demonstrates.

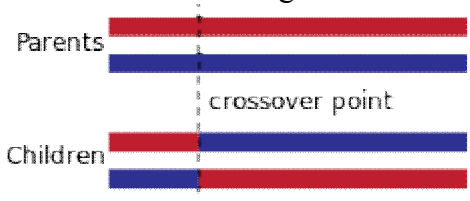

Fig. 2 Single Point Cross Over

Double Point Cross Over: In Double Point Cross Over, two random numbers are generated and from that two points parents combine to create new children as Figure 3 presents.

[Downloaded from mail.ijaor.ir on 2026-11-28]

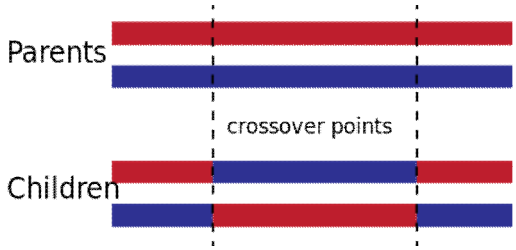

Fig. 3 Double Point Cross Over

Uniform Cross Over: In this kind of Cross Over, children can inherent about 50 percent of genes from each parents as Figure 4 shows.



Fig. 4 Uniform Cross Over

5.3.2 Mutation

In mutation, one or more than one genes are selected and changed. It is inspired from nature that sometimes happens.

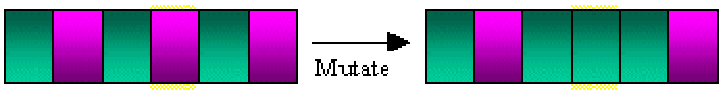

Fig. 5 Mutation Operator

5.3.3 Cross Over and Mutation in this paper

In this paper, each retailer (from vertical axis) or each day (from horizontal axis) is assumed to be a gene. Moreover, cross over operates to select some retailers and change their answers between selected parents. To achieve a more efficient answer, to choose parents we use a Roulette Wheel approach which is a well-known approach. To achieve a better combination, in each iteration one of the Single Point Cross Over, Double Point Cross Over and Uniform Cross Over is selected randomly and operates on answers. It makes reaching to optimal solution faster.

In mutation a retailer is chosen and its amounts of $WI_{i,l,t}$ changes. These two Operators create new generations and helps to reach a better solution. In mutation, those genes with more violations are chosen to change.

[Downloaded from mail.ijaor.ir on 2026-11-28]

5.4 Simulated Annealing Algorithm (SA)

Simulated Annealing Algorithm (SA) is a single solution based meta-heuristic algorithm. Which starts by a single solution like what was presented in Figure1. Then it tries to find a neighbor of the given solution. The algorithm will accept the solution if its value which is calculated by Fitness Function is better than present solution. But if it was worse than present solution it may be accepted by a probability function which decreases in each iteration and is depended on following parameters:

T_0	Initial temperature
T_f	Final temperature
T	Difference between T_f and T_0
ΔE	Difference between present solution and new solution

The probability function of accepting worse solutions is $e^{-\frac{\Delta E}{T}}$ that decreases in each iteration because *T0* reduces gradually in each iteration.

In this paper, to find a neighbor of our solution we choose some periods or customers randomly and changed their permutation to reach a neighbor of present solution. Then we evaluate new solution and decide to accept or reject it.

5.5 Hybrid of GA and SA (SAIGA)

In this paper, we propose a hybrid of Genetic Algorithm and Simulated Annealing algorithm. In this approach, GA starts with *nPop* chromosomes and tries to make solutions better. Then final solution, which is the best in GA, enters SA as its first solution. SA is able to escape from local optimum. If GA just could find a local optimum, SA can escape from local optimum and find a better solution nearer o global optimum. The pseudo code of the hybrid algorithm, which is a Simulated Annealing algorithm that is initialized by Genetic Algorithm (SAIGA), is presented in Figure 6.

6 Numerical results

In this part, we use a set of parameters that are demonstrated in Table 1. There are two key contribution consists of environmental impacts and role of age of products in consumer's demands and effect of that in satisfaction of customers. To check the model, we tested three different scenarios in small, medium and large scale of the problem. And solve them by GAMS 24.3.3 and computations are done by CPLEX Solver, and SAIGA Algorithm is done in MATLAB 2016 (a). Computations are done in a Computer by Intel® Core™ i7-4510U CPU @ 2.00 GHz. Processor Speed is 2594 MHz and 8 GB of RAM.

Number of Retailers, Products, Vehicles and time periods are represent in the Table 1 that shows size of the problem. Since VRP Problems are NP-hard as described before, by increasing in number of parameters it becomes too hard to solve. Amount of Transportation Costs, Environmental Costs of the different levels of vehicles' technology are represented. Holding Costs and demands of the retailers are shown in the Table 1. Other Parameters are generated by uniform and normal distribution.

[Downloaded from mail.ijaor.ir on 2026-11-28]

Step 1. Generate nPop first solution and evaluate them
While (stopping condition == false) do the following and n=1
Step 2. Randomly select two chromosomes and c=1
Step 3. Generate rc a random integer number between 1 to 3
3.1) if rc==1 use Single Point Cross Over function to generate 2 new Solutions
3.2) if rc==2 use Double Point Cross Over function to generate 2 new Solution s
3.3) if rc==3 use Uniform Point Cross Over function to generate 2 new Solutions
Step 4. Evaluate solutions and c=c+1
*If c=nc*nPop go to Step 5 otherwise go to Step 2*
Step 5. Randomly select one chromosomes and m=1
5.1) Use Mutation function to generate new Solution and evaluate and m=m+1
*If m=nm*nPop go to Step 6 otherwise go to Step 5*
Step 6 Sort all solutions' evaluation : truncate first nPop best Solution and n=n+1
If n=Maxiter stopping condition ==true
Step 7. Choose best solution and put W= best solution
While (stopping condition == false) do the following,
Step 8. Generate W_{nh} in the neighborhood of W using createneighbor function and evaluate
Step 9. Calculate $\Delta E = E(W_{nh}) - E(W)$, NI=0 , n=0
If $\Delta E \leq 0$, then set change=true
Else if $\Delta E \geq 0$ generate rs=random[0,1)
If $rs \leq e^{-\frac{\Delta E}{T}}$, then set change=true , NI=NI+1
Else set change =false
Step10. If(change==true) then set $W = W_{nh}$; n=n+1
If (n≤maxit2) then go to Step 8 otherwise go to Step 11
Step 11. If($E(W_{nh}) \leq E(W^*)$) set $W^* = W_{nh}$
Else if(NI≥Maxit1) then set Stopping condition=true
End.

Fig. 6 Pseudo code of the hybrid SAIGA algorithm**.**

To test different aspects of the problem. Some differences in transportation costs, environmental costs and holding costs are done.

Table 1 The problem sets and scenarios

	Number of Problem	Number of retailers	Number of periods	Number of vehicles	Number of Products	Retailers' demand	Vehicle capacity	Transportation cost per unit distance	Environmental cost of each Vehicle per unit of products	Holding Cost per unit of product
Scenario 1	1	3	3	2	2	U(20,30)	150-200	1-2	5-3	50
	2							1-2	10-6	50
	3							3-6	5-3	50
	4							1-2	5-3	100
Scenario 2	1	6	5	3	4	U(10,30)	200-250-300	1-2-3	5-3-1	50
	2							1-2-3	10-6-2	50
	3							3-6-9	5-3-1	50
	4							1-2-3	5-3-1	100
Scenario 3	1	12	10	4	5	U(10,40)	6000-7000-9000-10000	1-2-3-4	5-3-1-0.5	50
	2							1-2-3-4	10-6-2-1	50
	3							3-6-9-12	5-3-1-0.5	50
	4							1-2-3-4	5-3-1-0.5	100

[Downloaded from mail.ijaor.ir on 2026-11-28]

Additionally, different states of each scenario are solved and results are shown in Tables 2-4. We should notice that if different objective functions have contrast with each other or not.

Scenario 1: The first Scenario is small-Scale size of the problem. It has 3 Retailers which should be delivered by Supplier. Vehicles have small Capacities. State 1 is the base state of the Scenario. In Other States each Costs which are shown in Table 1 is changed and other Parameters are the same.

Table 2 Detailed costs for the first scenario problems

	GAMS Bounds- 1st Scenario								**SAIGA Approach- 1st Scenario**		
	Transportation Costs	Inventory Costs	Decreasing Demand Cost	Environmental Costs	Lower Bound	Upper Bound	Optimality gap (%)	Time (Seconds)	Objective Function	Optimality gap (%)	Time (Seconds)
1	3023	6410	150	27466.98	0.904	0.904	0	2.107	0.904	0	60.37
2	2760	9855	150	47125.62	0.550	0.550	0	1.466	0.550	0	72.38
3	8849	9955	0	23685.66	0.398	0.398	0	1.023	0.398	0	69.34
4	3033	12065	300	19868.31	0.450	0.450	0	1.002	0.450	0	71.32

Amounts of lower bound and upper bound Show the best possible of the problem and best solution we could calculate. Since the size of the problem is small these two values are the same. Also, optimality gap which shows differences between objectives is zero. This amounts are shown in GAMS report. SAIGA results show that metaheuristic algorithm can find optimal solutions. And we can use it for large-scale sizes of the problem.

Scenario 2: The Second Scenario shows medium-scale size of the problem. It has 6 retailers that should be served. Vehicles are larger and it cause more Environmental costs. Because as described before. Environmental Costs are related to the weight of the load.

Table 3 Detailed costs for the 2nd scenario problems

	GAMS Bounds- 2nd Scenario								SAIGA Approach- 2nd Scenario		
	Transportation Costs	Inventory Costs	Decreasing Demand Cost	Environmental Costs	Lower Bound	Upper Bound	Optimality gap (%)	Time (Seconds)	Objective Function	Optimality gap (%)	Time (Seconds)
1	16968	54040	350	70192.92	0.354	0.399	11.31	6201.813	0.389	10.01	271.05
2	17227	53142.5	1050	136892.7	0.348	0.425	18.08	7442.24	0.347	9.2	289.23
3	50038	59767.5	0	70315.38	0.360	0.418	13.83	5832.378	0.400	11.25	268.59
4	16697	69900	0	67830.21	0.308	0.335	7.89	1548.368	0.336	8.9	280.32

Results show that for this size. GAMS cannot reach to the optimal solution with 0% optimality gap. In addition, it has 12.5 % gap in average and to find the best solution GAMS runs up to 7443 seconds to reach an acceptable result. SAIGA in the other columns shows that meta-heuristic algorithm could find better solutions with less Optimality GAP. Optimality gap for meta-heuristic algorithm is calculated by considering Lower Bound of the GAMS as the Best Possible. Also as shown time consuming to calculate for meta-heuristic algorithm much more acceptable than GAMS runtime.

[Downloaded from mail.ijaor.ir on 2026-11-28]

Scenario 3: The third scenario shows large-scale size of the problem. It has 12 retailers that should be served. This size of the problem is much more similar to the reality. However, as it will be understood even this size is too hard to solve with deterministic approaches.

Table 3 Detailed costs for the 3rd scenario problems

	GAMS Bounds- 3rd Scenario								SAIGA Approach- 3rd Scenario		
	Transportation Costs	Inventory Costs	Decreasing Demand Cost	Environmental Costs	Lower Bound	Upper Bound	Optimality gap (%)	Time (Seconds)	Objective Function	Optimality gap (%)	Time (Seconds)
1	556.161	0	484.535	142077.99	0.112	-	-	50.989	0.194338	73	345.53
2	556.166	0	484.535	284158.602	0.113	-	-	82.077	0.208482	85	357.32
3	1390.523	0	484.358	254497.128	0.315	-	-	137.592	0.575652	83	275.35
4	473.787	0	484.363	253665.345	0.257	-	-	118.609	0.42096	64	312.24

Because the problems of this scenario was not solvable in acceptable time, we could not reach to a feasible integer solution. So we solved relaxed formed of the problem and it was about 96.75 seconds. SAIGA reaches to feasible integer solutions. Because it was validated in previous scenarios, it could be trustable that reaches to near optimum solutions. Because Lower Bound of the problems are belonged to Linear Programming form of the problem, Optimality GAP is in average 76.25% that shows a near optimum solution.

7 Conclusion

In this paper, we developed a Vehicle Routing Problem to transport Perishable products to retailers. We considered fuel consumption and tried to minimize the total cost that consists of transportation, inventory, back order, dissatisfaction and fuel consumption costs to lead a solution, which is better for all stockholders. We have experimented some numerical cases and determined that engaging environmental impacts and age of products is efficient to reach a more realistic understanding of the evaluation of the solution. For future studies, we propose engaging production planning in the model and try to make demands even to achieve a more productive supply chain. In addition, uncertainty of demands can be formulated and some managerial decision can be involved in the model.

References

1. Fisher, M. L., Jaikumar, R., (1981). A generalized assignment heuristic for vehicle routing. Networks*, 11*(2), 109-124.
2. Kara, I., Laporte, G., Bektas, T., (2004). A note on the lifted Miller–Tucker–Zemlin subtour elimination constraints for the capacitated vehicle routing problem. European Journal of Operational Research, 158(3), 793-795.
3. Toth, P., Vigo, D., (2014). Vehicle routing: problems, methods, and applications: SIAM.
4. Lalla-Ruiz, E., Expósito-Izquierdo, C., Taheripour, S., Voß, S., (2016). An improved formulation for the multi-depot open vehicle routing problem. OR spectrum, 38(1), 175-187.
5. Lenstra, J. K., Kan, A., (1981). Complexity of vehicle routing and scheduling problems. Networks, 11(2), 221-227.

[Downloaded from mail.ijaor.ir on 2026-11-28]

6. Rabbani, M., Ramezankhani, M. J., Farrokhi-Asl, H., Farshbaf-Geranmayeh, A., (2015). Vehicle routing with time windows and customer selection for perishable goods. International Journal of Supply and Operations Management, 2(2), 700.
7. Escobar, J. W., Linfati, R., Toth, P., Baldoquin, M. G., (2014). A hybrid granular tabu search algorithm for the multi-depot vehicle routing problem. Journal of Heuristics, 20(5), 483-509.
8. Reed, M., Yiannakou, A., Evering, R., (2014). An ant colony algorithm for the multi-compartment vehicle routing problem. Applied Soft Computing, 15, 169-176.
9. BañOs, R., Ortega, J., Gil, C., MáRquez, A. L., De Toro, F., (2013). A hybrid meta-heuristic for multi-objective vehicle routing problems with time windows. Computers & industrial engineering, 65(2), 286-296.
10. Goksal, F. P., Karaoglan, I., Altiparmak, F., (2013). A hybrid discrete particle swarm optimization for vehicle routing problem with simultaneous pickup and delivery. Computers & industrial engineering, 65(1), 39-53.
11. Leung, S. C., Zhang, Z., Zhang, D., Hua, X., Lim, M. K., (2013). A meta-heuristic algorithm for heterogeneous fleet vehicle routing problems with two-dimensional loading constraints. *European Journal of Operational Research, 225*(2), 199-210.
12. Mirzaei, S., Seifi, A., (2015). Considering lost sale in inventory routing problems for perishable goods. Computers industrial engineering, 87, 213-227.
13. Amorim, P., Parragh, S. N., Sperandio, F., Almada-Lobo, B., (2014). A rich vehicle routing problem dealing with perishable food: a case study. Top, 22(2), 489-508.
14. Govindan, K., Jafarian, A., Khodaverdi, R., Devika, K., (2014). Two-echelon multiple-vehicle location–routing problem with time windows for optimization of sustainable supply chain network of perishable food. International Journal of Production Economics, 152, 9-28.
15. Seyedhosseini, S., Ghoreyshi, S., (2014). An integrated model for production and distribution planning of perishable products with inventory and routing considerations. Mathematical Problems in Engineering, 2014.
16. Lin, C., Choy, K. L., Ho, G. T., Chung, S., Lam, H., (2014). Survey of green vehicle routing problem: past and future trends. Expert Systems with Applications, 41(4), 1118-1138.
17. Schneider, M., Stenger, A., Goeke, D., (2014). The electric vehicle-routing problem with time windows and recharging stations. Transportation Science, 48(4), 500-520.
18. Xiao, Y., Zhao, Q., Kaku, I., Xu, Y., (2012). Development of a fuel consumption optimization model for the capacitated vehicle routing problem. Computers & operations research, 39(7), 1419-1431.
19. Torabi, S. A., Hassini, E., (2008). An interactive possibilistic programming approach for multiple objective supply chain master planning. Fuzzy Sets and Systems, 159(2), 193-214.
20. Rabbani, M., Farrokhi-Asl, H., Asgarian, B., (2017). Solving a bi-objective location routing problem by a NSGA-II combined with clustering approach: application in waste collection problem. Journal of Industrial Engineering International, 13(1), 13-27.
21. Rabbani, M., Farrokhi-asl, H., Rafiei, H., (2016). A hybrid genetic algorithm for waste collection problem by heterogeneous fleet of vehicles with multiple separated compartments. Journal of Intelligent & Fuzzy Systems, 30(3), 1817-1830.

[Downloaded from mail.ijaor.ir on 2026-11-28]