IJE TRANSACTIONS A: Basics Vol. 27, No. 4, (April 2014) 533-544

International Journal of Engineering

Journal Homepage: www.ije.ir

Optimal Design of the Cross-docking in Distribution Networks: Heuristic Solution Approach

S. M. Mousavi *[a,b] , R. Tavakkoli-Moghaddam[a], A. Siadat[b]

[a] *School of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran*
[b] *LCFC, Arts et Métier Paris Tech, Metz, France*

PAPER INFO

Paper history:
Received 30 March 2013
Received in revised form 17 May 2013
Accepted 22 August 2013

Keywords:
Logistics Management
Cross-Docking Distribution Networks
Mixed-Integer Programming (MIP) Model
Heuristics
Simulated Annealing

ABSTRACT

Cross-docking plays an importation role in distribution networks. In the recent years, a cross-docking design network problem is addressed as a new research area in logistics management. This paper presents a new mathematical model for the location of cross-docking facilities and vehicle routing scheduling problems in the distribution networks. For this purpose, a two-phase mixed-integer programming (MIP) is formulated. Then, a new heuristic-based simulated annealing (SA) is developed for solving the proposed MIP model. Finally, the presented heuristic algorithm is subsequently tested on a number of small and large-scale instances. The computational results for different-sized instances illustrate that the proposed algorithm performs effectively in a reasonable time.

doi: *10.5829/idosi.ije.2014.27.04a.04*

1. INTRODUCTION

A cross-docking distribution network can be defined as the delivery of enhanced customer and economic value through synchronized management of the flow of physical products and related information from suppliers to customers in the limited time available (e.g., two days). A network of autonomous business cross-docking facilities is considered, in which the arriving goods are transferred by vehicles (e.g., trucks) and are delivered to the retailers or customers as rapidly as possible. The performance of any node in the distribution network depends on the performance of other nodes, and their ability to coordinate activities within the network. Thus, designing this distribution network with the cross-docking can be considered as an important issue in logistics management [1-4].

To design the cross-docking distribution network, in this paper location of cross-docking facilities and vehicle routing scheduling are considered. This problem is known as NP-hard problem [1, 5]; by increasing sizes of this problem, there is a need to utilize heuristic and meta-heuristic algorithms as effective solving approaches. The exact algorithms need exponential CPU time by considering the size of the problems.

In the related literature, there exist some studies focusing on distribution planning problems with the cross-docking. For instance, Jayaraman and Ross [1] presented a practical approach for solving a multiple product, multi-echelon problem for distribution network design by the simulated annealing (SA) algorithm. Li et al. [6] addressed a cross-docking facility operation in order to eliminate or minimize storage and order picking activity in the cross-docking using just-in-time (JIT) scheduling. Lee et al. [5] presented an integration model of cross-docking facilities with vehicle routing scheduling for the distribution network design problem. Ross and Jayaraman [7] addressed an evaluation of heuristics for the location of cross-docking facilities in the supply chain. Musa et al. [8] considered the transportation problem of a cross-docking network, in which loads were transferred from suppliers to retailers through cross-docking facilities. Belle et al. [9] provided a review of the existing literature about cross-docking. The papers were classified according to the problem type ranging from more strategic or tactical to more operational problems. Liao [10] considered the

* Corresponding Author Email: sm.mousavi@ut.ac.ir (S. M. Mousavi)

simultaneous dock assignment and sequencing of inbound trucks for a multi-door cross-docking operation with the objective to minimize total weighted tardiness, under a fixed outbound truck departure schedule.

Regarding the recent developments of the cross-docking systems, Boysen and Fliedner [11] proposed a classification of truck scheduling problems, and then future research needs were identified. Acar et al. [12] presented a mixed integer quadratic model with the objective of generating trailer-to-door assignments which equally distribute idle times at doors to accommodate operational level uncertainty by considering truck arrival times. Then, a heuristic was introduced for the door assignment. Shakeri et al. [13] considered truck scheduling in a resource-constrained cross-docking, and the sequence of incoming and outgoing trucks at the dock doors of the cross-docking terminal by regarding the availability of cross-dock resources. Then, an algorithmic approach was extended which was capable of establishing solution feasibility for the problem. Hu et al. [14] formulated the optimal route selection problems in the fashion supply chain from the suppliers to the cross-docking center and from the cross-docking center to the customers as the respective vehicle routing problem. Konur and Golias [15] addressed the cross-dock operator's problem and proposed a cost-stable scheduling strategy while minimizing the average of total service costs. Also, a bi-objective bi-level optimization problem was formulated and a genetic algorithm was developed.

This paper introduces a new two-phase mixed-integer programming (MIP) model for designing cross-docking distribution networks. Then, a new heuristic-based SA is presented that characterizes a special solution representation scheme for the location of cross-docking and routing scheduling in the distribution networks. The computational results indicate that the proposed solving approach performs well on small and large-scale problems within a reasonable amount of time.

The structure of this paper is organized in six sections. In the next section, the problem definition is defined, and then a new two-phase MIP model is proposed in Section 3. The problem-solving approach is described in Section 4. Computational results are discussed in Section 5. Finally, conclusions are provided in Section 6.

2. PROBLEM DEFINITION

Cross-docking is a distribution network to decrease inventory while satisfying customers' requirements. Through streamlining the flow between the suppliers and manufacturers, this distribution can help to reduce or eliminate inventory storage.

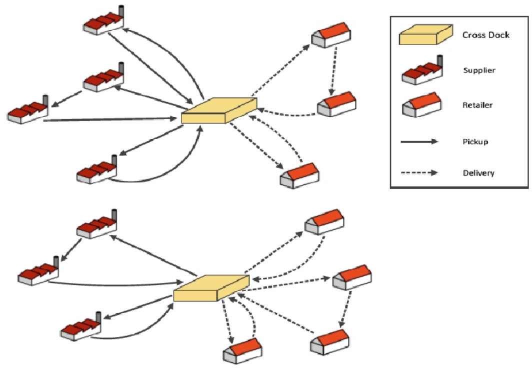

Figure 1. Concept of the proposed cross-docking distribution network

The cross-docking copes with the movement of goods directly from the receiving dock to the shipping dock, in which the goods are stored in cross-docking facilities for a short time or directly transformed to the customers [2]. In fact, the inventory holding function of a traditional warehouse can be eliminated by the cross-docking while still goods need to be classified and loaded to the delivery vehicles through a consolidation process [2, 5]. The concept of proposed distribution network with the cross-docking is illustrated in Figure 1, in which two main nodes are simultaneous arrival and consolidation.

The location of cross-docking facilities and vehicle routing scheduling problems can be stated as follows: given a set of retailers with known demand and a set of potential cross-docking facilities, the location of the cross-docking facilities is determined in the first phase. Then, the vehicle routing scheduling from the cross-docking facilities is obtained in the pickup and delivery processes to minimize the sum of the costs associated with the location of cross-docking facilities and distribution from suppliers / to the retailers. In the first phase, three types of costs are minimized. The first type of costs is fixed costs associated with operating open cross-docking facilities. The second type is costs to transport units of the product from suppliers to cross-docking facilities and from cross-docking facilities to retailers. The third type is the cost of holding inventories at cross-docking facilities. In addition, in the second phase distribution costs are considered associated with the routing of vehicles, containing operational costs of vehicles and transportation costs.

The proposed model is to obtain the minimum number of cross-docking facilities among a discrete set of location sites in the first phase, and then to obtain the number of vehicles and the best route as well as the arrival time of each vehicle in the second phase. Generally, the distribution and routing plan can be designed so that the demand of each customer can be satisfied. Each customer is served by only one vehicle. The total demand on each route is less than or equal to the capacity of the vehicle assigned to that route. Each route starts and ends at the same cross-docking facility. Also, the total quantity of pickup should equal the quantity to be delivered. It is assumed that all vehicles are located in the multiple cross-docking facilities, and split pickups and deliveries are not allowed.

3. PROPOSED MODEL FORMULATION

The following notations are used in formulation of the proposed MIP model for the cross-docking location problem in the first phase, and vehicle routing scheduling problem with multiple cross-docking facilities in the second phase.

3. 1. Sets and Input Parameters

P: Set of suppliers (i =1, 2, ..., n)

D: Set of retailers (i' =1, 2, ..., m)

O: Set of cross-docking facilities (o= 1, 2, …, c)

T: Set of time ($t = t_{min}$, …, t_{max})

Am_i: Amount of product in pickup node i

$Am_{i'}$: Amount of product in delivery node i'

$Dist_{i,o}$: Distance of pickup node i from cross-docking facility o

$Dist_{i',o}$:Distance of delivery node i' from cross-docking facility o

Cap_o : Capacity of cross-docking facility o

HC_o : Holding cost per unit product in a unit of time at cross-docking facility o

T_{min} , T_{max} : Minimum and maximum of time horizon

F_o: Fixed cost to open cross-docking facility o

R: Maximum number of cross-docking facilities to be opened

K: Number of available vehicles in the pickup process

K' : Number of available vehicles in the delivery process

$d_{i'}$: Unloaded amount of product in node i' in the delivery process

Q: Maximum capacity of each vehicle

c_{ij}: Transportation cost from node i to node j in the pickup process

$c_{i'j'}$: Transportation cost from node i' to node j' in the delivery process

c_k: Operational cost of vehicle k

$c_{k'}$: Operational cost of vehicle k'

d_{ij}: Distance from node i to node j in the pickup process

$d_{i'j'}$: Distance from node i to node j' in the delivery process

t_i^k : Length of a visit for vehicle k in node i in the pickup process

$t_{i'}^{k'}$: Length of a visit for vehicle k' in node i' in the delivery process

et_{ij}: Time for the vehicle to move from node i to node j in the pickup process

$et_{i'j'}$: Time for the vehicle to move from node i' to node j' in the delivery process

3. 2. Decision Variables

$X_{o,t}^i$: 1 if product in pickup i goes to cross-docking facility o at time t, and 0 otherwise

$X_{o,t}^{i'}$: 1 if product in delivery i' is bound for cross-docking facility o at time t, and 0 otherwise

$S_{o,t}$: Amount of product at cross-docking facility o at time t

x_o: 1 if cross-docking facility o is open, and 0 otherwise.

X_{ij}^k: 1 if vehicle k transports product from node i to node j in the pickup process, and 0 otherwise

$X_{i'j'}^{k'}$: 1 if vehicle k transports product from i' to node j' in the delivery process, and 0 otherwise

y_{ij} : Transported amount of product from node i to node j in the pickup process

$z_{i'j'}$: Transported amount of product from node i' to node j' delivery process

DT_i^k : Departure time of vehicle k from node i in the pickup process

$DT_{i'}^{k'}$: Departure time of vehicle k' from node i' in the delivery process

DT_j^k : Departure time of vehicle k from node j in the pickup process

$DT_{j'}^{k'}$: Departure time of vehicle k' from node j' in the delivery process

AT_j^k : Arrival time of vehicle k from node j in the pickup process

$AT_{j'}^{k'}$: Arrival time of vehicle k' from node j' in delivery process

AT_o^k: Arrival time of vehicle k at cross-docking facility o in the pickup process

$AT_{o'}^{k'}$: Arrival time of vehicle k' at cross-docking facility o' delivery process

3. 3. Cross-docking Facilities Location (Phase 1)

The location problem of cross-docking facilities can be formulated as below:

$$MinZ_1 = \sum_{o=1}^{c} F_o x_o + \sum_{o=1}^{c}\sum_{t=T_{\min}}^{T_{\max}} HC_o S_{o,t} + \sum_{i'=1}^{m}\sum_{o=1}^{c}\sum_{t=T_{\min}}^{T_{\max}} X_{o,t}^{i'} Dist_{i',o} C_{i',o} + \sum_{i=1}^{n}\sum_{o=1}^{c}\sum_{t=T_{\min}}^{T_{\max}} X_{o,t}^{i} Dist_{i,o} C_{i,o} \tag{1}$$

s.t.

$$\sum_{o=1}^{c}\sum_{t=T_{\min}}^{Ts_d-1} X_{o,t}^{i'} = 0 \qquad \forall i' \tag{2}$$

$$\sum_{o=1}^{c}\sum_{t=TS_d}^{TE_d} X_{o,t}^{i'} \le 1 \qquad \forall i' \tag{3}$$

$$\sum_{o=1}^{c}\sum_{t=TE_d+1}^{T_{\max}} X_{o,t}^{i'} = 0 \qquad \forall i' \tag{4}$$

$$\sum_{o=1}^{c}\sum_{t=T_{\min}}^{Ts_p-1} X_{o,t}^{i} = 0 \qquad \forall i \tag{5}$$

$$\sum_{o=1}^{c}\sum_{t=TS_p}^{TE_p} X_{o,t}^{i} = 1 \qquad \forall i \tag{6}$$

$$\sum_{o=1}^{c}\sum_{t=TE_p+1}^{T_{\max}} X_{o,t}^{i} = 0 \qquad \forall i \tag{7}$$

$$\sum_{o=1}^{C}\sum_{i=1}^{n} X_{o,t}^{i} Am_i \ge \sum_{o=1}^{C}\sum_{i'=1}^{m} X_{o,t}^{i'} Am_{i'} \qquad \forall T_{\min} \le t \le T_{\max} \tag{8}$$

$$s_{o,T_{\min-1}} = 0 \qquad \forall o \tag{9}$$

$$s_{o,t} = s_{o,t-1} - \sum_{i'=1}^{m} X_{o,t}^{i'} Am_{i'} + \sum_{i=1}^{n} X_{o,t}^{i} Am_i \qquad \forall o \text{ and } T_{\min} \le t \le T_{\max} \tag{10}$$

$$S_{o,t} \le cap_o \qquad \forall o \text{ and } T_{\min} \le t \le T_{\max} \tag{11}$$

$$X_{o,t}^{i} \le x_o \qquad \forall i,o \text{ and } T_{\min} \le t \le T_{\max} \tag{12}$$

$$X_{o,t}^{i'} \le x_o \qquad \forall i',o \text{ and } T_{\min} \le t \le T_{\max} \tag{13}$$

$$\sum_{o=1}^{c} x_o \le R \tag{14}$$

$$X_{o,t}^{i}, X_{o,t}^{i'}, x_o \in \{0,1\} \tag{15}$$

$$S_{o,t} \ge 0 \tag{16}$$

Objective function (1) minimizes the total costs including costs of holding inventories at cross-docking facilities and costs of transportation, namely costs of transportation from suppliers to cross-docking facilities and then from cross-docking facilities to retailers. Constraints (2), (3) and (4) ensure that each delivery, if necessary, is fulfilled within its specified time window and beyond that range it takes the value of zero. Constraints (5), (6) and (7) guarantee the time window restriction for pickups. Constraint (8) ensures the sufficient inventory of product to meet all demands. Constraint (9) sets a zero initial inventory for product at each cross-docking facility. Changes in the inventory level of each cross-docking facility at each time are indicated by constraint (10). Constraint (11) describes the potential capacity of cross-docking facilities. Constraints (12) and (13) ensure that transporting product from suppliers to cross-docking facility and from cross-docking facility to retailers in the pickup and delivery processes can be performed only when the corresponding cross-docking facility is open. Constraint (14) limits the number of cross-docking facilities that can be located. Constraints (15) and (16) define decision variables of the model.

3. 4. Vehicle Route Scheduling (Phase 2):

The vehicle route scheduling problem with multiple cross-docking facilities can be formulated by:

$$Min\ Z_2 = \sum_{i=1}^{n}\sum_{j\in(P\cup O)}\sum_{k=1}^{K} (c_{ij} d_{ij}) x_{ij}^{k} + \sum_{i\in(P\cup O)}\sum_{j=1}^{n}\sum_{k=1}^{K} \left[(c_{ij} d_{ij}) + c_k\right] x_{ij}^{k} + \sum_{i'=1}^{m}\sum_{j'\in(D\cup O)}\sum_{k'=1}^{K'} (c_{i'j'} d_{i'j'}) x_{i'j'}^{k'} + \sum_{i'\in(D\cup O)}\sum_{j'=1}^{m}\sum_{k'=1}^{K'} \left[(c_{i'j'} d_{i'j'}) + c_{k'}\right] x_{i'j'}^{k'} \tag{17}$$

s.t.

$$\sum_{i\in(P\cup O)}\sum_{k=1}^{K} x_{ij}^{k} = 1 \qquad \forall j \tag{18}$$

$$\sum_{j\in(P\cup O)}\sum_{k=1}^{K} x_{ij}^{k} = 1 \qquad \forall i \tag{19}$$

$$\sum_{i'\in(D\cup O)}\sum_{k'=1}^{K'} x_{i'j'}^{k'} = 1 \qquad \forall j' \tag{20}$$

$$\sum_{j'\in(D\cup O)}\sum_{k'=1}^{K'} x_{i'j'}^{k'} = 1 \qquad \forall i' \tag{21}$$

$$\sum_{i\in(P\cup O)}\sum_{k=1}^{K} x_{ij}^{k} \ge 1 \qquad \forall j \in O \tag{22}$$

$$\sum_{i'\in(D\cup O)}\sum_{k'=1}^{K'} x_{i'j'}^{k'} \geq 1 \qquad \forall j' \in O \tag{23}$$

$$\sum_{i\in(P\cup O)} x_{ir}^{k} = \sum_{j\in(P\cup O)} x_{rj}^{k} \qquad \forall k, r \in (P\cup O) \tag{24}$$

$$\sum_{i'\in(D\cup O)} x_{i'r'}^{k'} = \sum_{j'\in(D\cup O)} x_{r'j'}^{k'} \qquad \forall k', r \in (D\cup O) \tag{25}$$

$$\sum_{i\in(P\cup O)}\sum_{j\in(P\cup O)} x_{ij}^{k} \leq 1 \qquad \forall k \tag{26}$$

$$\sum_{i'\in(D\cup O)}\sum_{j'\in(D\cup O)} x_{i'j'}^{k'} \leq 1 \qquad \forall k' \tag{27}$$

$$\sum_{i\in(P\cup O)}\sum_{j\in(P\cup O)}\sum_{k=1}^{K} x_{ij}^{k} \leq K \tag{28}$$

$$\sum_{i'\in(D\cup O)}\sum_{j'\in(D\cup O)}\sum_{k'=1}^{K'} x_{i'j'}^{k'} \leq K' \tag{29}$$

$$y_{ij} \leq Q \qquad \forall i, j \in (P\cup O) \tag{30}$$

$$z_{i'j'} \leq Q \qquad \forall i', j' \in (D\cup O) \tag{31}$$

$$\sum_{i=1}^{n} p_i = \sum_{i'=1}^{m} d_{i'} \tag{32}$$

$$y_{jr} = y_{ij} - p_j \qquad if\ i \in P, \forall j, r \tag{33}$$

$$y_{jr} = y_{ij} - \sum_{i=1}^{n} p_j \qquad if\ i \in O, \forall j, r \tag{34}$$

$$z_{i'j'} = z_{i'r'} + d_{j'} \qquad if\ i' \in D, \forall j', r' \tag{35}$$

$$z_{i'j'} = z_{j'r'} + \sum_{i'=1}^{m} d_{i'} \qquad if\ i' \in O, \forall j', r' \tag{36}$$

$$u_i - u_j + n\sum_{k=1}^{K} x_{ij}^{k} \leq (n-1) \qquad \forall i, j \tag{37}$$

$$u_{i'} - u_{j'} + m\sum_{k'=1}^{K'} x_{i'j'}^{k'} \leq (m-1) \qquad \forall i', j' \tag{38}$$

$$DT_j^{T} \geq \left(et_{ij} + DT_i^{k} + t_i^{k}\right)x_{ij}^{k} \qquad \forall k, i, j \tag{39}$$

$$DT_{j'}^{T'} \geq \left(et_{i'j'} + DT_{i'}^{k'} + t_{i'}^{k'}\right)x_{i'j'}^{k'} \qquad \forall k', i', j' \tag{40}$$

$$AT_j^{k} \geq \left(DT_i^{k} + et_{ij}\right)x_{ij}^{k} \qquad \forall k, i, \forall j \in O \tag{41}$$

$$AT_{j'}^{k'} \geq \left(DT_{i'}^{k'} + et_{i'j'}\right)x_{i'j'}^{k'} \qquad \forall k', i', \forall j' \in O \tag{42}$$

$$AT_o^{k} = AT_{o'}^{k''} \qquad \forall k \neq k'', \forall o \neq o' \tag{43}$$

$$AT_o^{k'} = AT_{o'}^{k'''} \qquad \forall k' \neq k''', \forall o \neq o' \tag{44}$$

$$u_i \leq n \qquad \forall i \tag{45}$$

$$u_j \leq n \qquad \forall j \tag{46}$$

$$u_{i'} \leq m \qquad \forall i' \tag{47}$$

$$u_{j'} \leq m \qquad \forall j' \tag{48}$$

$$x_{ij}^{k}, x_{i'j'}^{k'} \in \{0,1\} \qquad \forall i, j, i', j' \tag{49}$$

$$\begin{aligned} &y_{ij}, z_{i'j'}, DT_i^{k}, DT_{i'}^{k'}, DT_j^{k}, DT_{j'}^{k'}, AT_o^{k}, AT_{o'}^{k'}, \\ &u_i, u_j, u_{i'}, u_{j'} \geq 0 \qquad \forall i, j, i', j', o, o', k, k' \end{aligned} \tag{50}$$

The objective function (17) minimizes total transportation costs associated with moving product in the pickup and deliver processes as well as operational cost of each vehicle in these processes separately. Constraints (18) and (19) show that one vehicle has to arrive at and leave one node in the pickup process. Constraints (20) and (21) show that one vehicle has to arrive at and leave one node in the delivery process. Constraints (22) and (23) specify that every supplier or retailer belongs to one and only one route, but cross-docking facilities may belong to more than one route. Constraints (24) and (25) express the consecutive movement of vehicles. Whether or not a vehicle arrives at and leaves a cross-docking facility in the pickup and delivery processes is shown in constraints (26) and (27). Constraints (28) and (29) ensure that the numbers of vehicles that arrive or leave a cross-docking facility in the pickup or delivery processes must be less than the number of available vehicles. Constraints (30) and (31) express that the quantity of loaded product in a vehicle cannot exceed the maximum capacity of the vehicle. The flow conservation for product is manifested in constraint (32). The quantity of products between nodes in the pickup and delivery processes is shown in constraints (33) – (36). Constraints (37) and (38) ensure that every retailer is on a route connected to the set of cross-docking facilities. Constraints (39) and (40) express that the departure time of a vehicle from a node is determined by the sum of the arrival time at a node, the length of a visit, and time to move in the pickup and delivery processes. The arrival time at a cross-docking facility is represented in constraints (41) and (42) for the pickup and delivery processes. The constraints for simultaneous arrival to a cross-docking facility are given in Equations (43) and (44). Constraints (45) – (50) enforce the integrality restrictions on the decision variables.

4. PROPOSED HEURISTIC

The proposed heuristic for designing cross-docking distribution networks is based on the SA algorithm. The SA can be regarded as a local search-based algorithm, and can be able to escape from being trapped into a local optimum by accepting with small probability worse solutions during its iterations. The concept of the algorithm is based on the annealing process applied to the metallurgical industry [1]. This algorithm has been successfully employed to numerous complicated combinatorial optimization problems as well as a wide range of real-world problems [e.g., 1, 7].

4. 1. Proposed Heuristic for Cross-docking Facilities Location (Phase 1) This sub-section explains the algorithmic steps of heuristic-based SA.

The search for least-cost solutions is guided by a control parameter, known as temperature (T), and this temperature determines the acceptance of inferior solutions. Proposed algorithm begins with a randomly generated initial configuration which specifies the cross-docking facilities to be opened, the suppliers and retailers assigned to the cross-docking facilities. The total cost is computed by the objective function of the proposed model.

Step 1: Initialization Initial and final values of the control parameter temperature are considered, known as T_0 and T_f respectively. An initial cross-docking facility solution is randomly obtained by assigning supply of suppliers and demand flows of retailers between pickup, cross-docking facilities and delivery nodes in the cross-docking distribution network. This results in an initial feasible solution by producing product flows. The objective function value of this solution can be regarded as the objective function value for the best configuration found best solution (BS), current configuration $C(\Phi)$, and the newest configuration $C(\Phi')$. All counters are set to 1.

Step 2: Check Feasibilities The algorithm now assesses product flow assignments for cross-docking facilities to make sure that the capacity of each cross-docking facility, investment opening cost and numbers of potential cross-docking facilities are satisfied. Furthermore, we check that quantity of product and demand of customer should be satisfied. If the configuration is not feasible, we return to step 1.

Step 3: Provide a Feasible Neighboring Solution Once the network design problem has been initialized, an objective function value is calculated, and feasibility ensured, the current feasible cross-docking system configuration is then updated by choosing a supplier and reassigning the amount of product between a cross-docking facility and supplier. On the other hand, this procedure can be utilized for retailers. They are accomplished by randomly choosing a supplier and a customer to perturb. Its flow is randomly allocated to another combination of pickup / cross-docking facility / delivery nodes. All feasibilities are investigated once again. Finally, the objective function value of the neighboring solution $C(\Phi')$ is determined.

Step 4: Assess Current Solution with Neighboring Solution If the objective function value of the neighboring solution is greater than that of the current solution ($C(\Phi') > C(\Phi)$), process to Step 5. Otherwise, if the objective functions value of the newest configuration improves over the current solution ($C(\Phi') < C(\Phi)$), the neighboring solution can be regarded as the current solution. Then, this solution is compared to BS (i.e., the best solution obtained thus far). If the objective function value of the newest configuration is less than that of the best one found so far $(C(\Phi') < BS)$, then replace the best solution with that of the neighboring solution process to step 6.

Step 5: Investigate Metropolis Condition The difference (Δcost) between the neighboring solution and the current solution is calculated, as $(\Delta\text{cost} = C(\Phi') > C(\Phi))$. Then, the Metropolis criterion is applied to obtain the probability at which the relatively inferior neighboring solution can be accepted, $P(A)$. This probability is calculated by [1]:

$$P(A) = \exp(\Delta\text{cost}/T_i) \tag{51}$$

where T_i is the current temperature. A random number is then generated from the interval (0,1). If this random number is less than $P(A)$, then the neighboring solution replaces the current solution. Proceed to Step 6.

Step 6: Increase counters. Memory and status variables are updated. The counters are incremented by one. If the iteration counter value is less than or equal to the maximum iterations for the temperature level, then return to Step 3. Otherwise go to Step 7.

Step 7: Adjust Temperature Temperature is adjusted in iteration i by the cooling rate. Mathematically, we have:

$$T_i = T_0 - i^{\frac{\ln(T_o - T_f)}{\ln(N)}} \tag{52}$$

If the new value of T_i is greater than or equal to the stopping value (T_f), then reset iteration counters to one and return to Step 3. Otherwise, stop.

4. 2. Proposed Heuristic for Vehicle Routing Scheduling (Phase 2)

This sub-section explains the algorithmic steps of the heuristic-based SA for the routing scheduling problem, providing insight into the progress of the search.

Step 1: Initialization The path representation is applied to encode the solution of the vehicle routing scheduling problem with multiple cross-docking facilities in the distribution networks. The idea of the path representation is that the suppliers and retailers are listed in the order, in which they are visited in pickup and delivery processes through the cross-docking facilities. For instance, suppose that there are seven suppliers numbered 1 to 7. If the path representation is [02350610470], then three routes are needed to serve all these seven suppliers in the pickup process. In the first route, a vehicle starts from the cross-docking facility, which is illustrated as 0, travels to suppliers 2, 3 and finally supplier 5. After that, the vehicle returns back to the cross-docking facility. In the second route, the vehicle starts with supplier 6 and then supplier 1. Similarly, the vehicle travels back to the cross-docking facility after serving the suppliers. In the third route, the vehicle starts with supplier 4 and then supplier 7. Similarly, the vehicle travels back to the cross-docking facility after serving the suppliers. In the same way, this procedure will be utilized for delivery processes.

It is pointed out that each solution contains O links if there are O cross-docking facilities in the vehicle routing scheduling problem. For this problem in the step of initialization, there are three sub-steps to generate a feasible initial solution. The first sub-step is to assign suppliers / retailers to each of the O links, that is, the grouping problem. There are a number of cross-docking facilities, suppliers and retailers, and each supplier / retailer should be allocated to one cross-docking facility or link. Because the objective function is to minimize the total distribution costs, suppliers and retailers are assigned to the cross-docking facility which is minimum distribution cost. The second sub-step is to assign suppliers / retailers in the same link to several routes by using the saving method in [16]. The method constructs a saving matrix for every two suppliers / retailers in the same link. Then, the suppliers / retailers with large saving value are grouped in the same route while not violating the vehicle capacity constraint and arrival time constraint. The third sub-step is to solve the scheduling problem by the NNH in [17]. The principle of the NNH is to randomly begin with the first supplier and retailer. Then, the next customer / retailer is chosen as minimum cost to the previous one from those unselected suppliers/retailers to build the pickup and delivery sequence until all suppliers and retailers are chosen.

Step 2: Improvement This procedure is based on the SA to improve upon the best solution obtained at any step of the algorithm. The algorithm is described as follows:

For $i = 1\text{–}n$ do
  For $i' = 1\text{–}m$ do
  (a) Initialize max-iterations, initial temperature.
    Set count = 1, T_0= temp-start.
  (b) Let the best solution obtaining in the initialization step be called the current solution, x_c.
    Compute the objective function for current solution, $OBF(x_c)$.
    Randomly generate a neighbouring solution using either the interchange neighbourhood; forward insertion neighbourhood or backward insertion neighbourhood. Let neighbouring solution called the adjacent solution, x_a. Compute objective function for adjacent solution, $OBF(x_a)$
  (c) If $OBF(x_a) < OBF(x_c)$
    Then set $x_c = x_a$;
    Else
    Set $\Delta = OBF(x_a) - OBF(x_c)$;
    Set T = temp-start/log(1+count);
    With probability $e^{-\Delta/T}$ set $x_c = x_a$.
    Increment count by 1.
  (d) If count < max-iteration, go to Step (b).

The output of the current solution is applied as the final solution. The annealing schedule used in Step (c) of the above algorithm is based on [18]. The interchange neighborhood, by far the most popular scheme, is simple: swap two randomly chosen supplier / retailers in the pickup and delivery sequence. In forward insertion neighborhood a supplier / retailer is relocated further forward in the sequence, and in backward insertion neighborhood a supplier / retailer is relocated further backward in the sequence. Figures 2 and 3 show the solution representations of an example and visual illustrations by the proposed heuristic-based SA algorithm for the second phase of the MIP model.

5. COMPUTATIONAL RESULTS

In this section, computational results are illustrated in small and large-scale cases for the proposed two-phase MIP model verification as well as the proposed heuristic results, respectively. For this purpose, eight test problems are solved in small sizes by GAMS optimization software for two phases of the presented model, including multiple cross-docking facilities locating (phase one) and vehicle route scheduling with multiple cross-docking facilities (phase two). Sizes of the test problems are given in Table 1. All parameters for the first and second phases of the proposed model are given in Tables 2 and 3. Some parameters are generated randomly in uniform distributions. It is pointed out that the problem-solving approach is coded in MATLAB. All small and large-sized test problems are run by using the Intel Dual Core, 2.8 GHz compiler and 2 GB of RAM. The comparison of GAMS with the proposed heuristic illustrates that the proposed algorithm can approximately obtain an optimal solution in less time than GAMS as provided in Tables 4 and 5. The average gaps between the optimal and the heuristic solutions for the first and second phases are 3.22% and 3.70% indicating the efficiency of the proposed heuristic-based SA algorithm in the distribution network. Moreover, increasing the size of the two-phase cross-docking distribution network problem increases the solution time of GAMS exponentially while it does not have significant impacts on the solution time of the proposed algorithm.

Pickup process

Crossdock 1	0	1	4	0	2	3	5
Crossdock 2	0	6	7	8	0	9	10

Delivery process

0	5	3	0	1	9	11	0	8	12	0	Crossdock 1
0	4	10	0	7	6	0	14	2	13	0	Crossdock 2

Figure 2. Example of solution representations

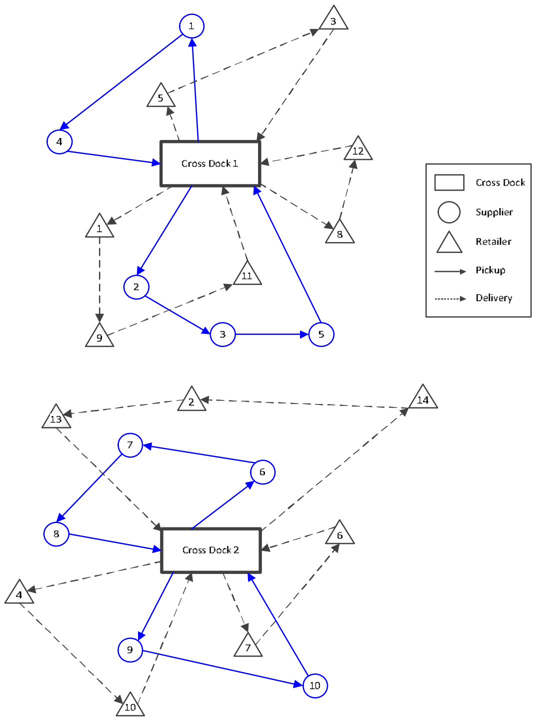

Figure 3. Visual illustrations of the example solution

TABLE 1. Sizes of small test problems

Problem no.	No. of suppliers (n)	No. of potential cross-docking facilities (c)	No. of retailers (m)
1	3	2	4
2	4	3	5
3	5	3	6
4	7	4	8
5	8	5	9
6	9	6	10
7	10	7	11
8	11	8	12

TABLE 2. Sources of random generations for the first phase of proposed MIP model

Parameters	Problems 1 & 2	Problems 3 & 4	Problems 5 & 6	Problems 7 & 8
HC_o	~Uniform (100, 200)	~Uniform (120, 220)	~Uniform (180, 280)	~Uniform (180, 300)
$Dist_{i,o}$, $Dist_{i',o}$	~Uniform (20, 40)	~Uniform (25, 45)	~Uniform (30, 50)	~Uniform (20, 45)
TC	~Uniform (5000, 6000)	~Uniform (9500, 12500)	~Uniform (15000, 18000)	~Uniform (12500, 18000)
R	2	3	4	6
Cap_o	~Uniform (200, 500)	~Uniform (350, 700)	~Uniform (500, 800)	~Uniform (400, 600)
F_o	~Uniform (2000, 5000)	~Uniform (2500, 5500)	~Uniform (3000, 6000)	~Uniform (2000, 6000)

TABLE 3. Sources of random generations for the second phase of the proposed MIP model

Parameters	Problems 1 & 2	Problems 3 & 4	Problems 5 & 6	Problems 7 & 8
k	5	6	8	8
K'	4	5	6	7
Q	~Uniform (200, 1000)	~Uniform (200,1100)	~Uniform (200,1200)	~Uniform (100, 1400)
p_i	~Uniform (20, 30)	~Uniform (10, 40)	~Uniform (5, 40)	~Uniform (5, 45)
$d_{i'}$	~Uniform (15, 30)	~Uniform (10, 35)	~Uniform (5, 35)	~Uniform (5, 40)
c_{ij}, $c_{i'j'}$	~Uniform (300, 500)	~Uniform (200, 500)	~Uniform (100,500)	~Uniform (200, 500)
d_{ij}, $d_{i'j'}$	~Uniform (20, 30)	~Uniform (20, 40)	~Uniform (20, 50)	~Uniform (15, 55)
c_k, $c_{k'}$	~Uniform (150, 250)	~Uniform (250, 450)	~Uniform (200,500)	~Uniform (200,600)
t_i^k , $t_{i'}^{k'}$	~Uniform (35,45)	~Uniform (30,50)	~Uniform (20,50)	~Uniform (15,55)
et_{ij}, $et_{i'j'}$	~Uniform (50,150)	~Uniform (40, 200)	~Uniform (40, 250)	~Uniform (30, 250)

TABLE 4. Results in small-sized test problems for the first phase

No. of test problems	GAMS		Proposed heuristic-based SA (400 iterations)		
	Best solution	Time (s)	Best solution	Time (s)	Gap (%)
1	5871.6	31.6	5951.4	24.6	1.36
2	10753.4	55.2	11048.8	28	2.75
3	11534.6	83.2	11858	30	2.80
4	24794	104.5	25390.4	30.5	2.41
5	32352.6	111.7	34305.6	34	6.04
6	36974	149	38207.4	36.2	3.34
7	49106.4	230.5	50831.2	36.5	3.51
8	55897	248	57909	41	3.60
Average	28410.5	126.7	29437.7	32.6	3.22

TABLE 5. Results in small-sized test problems for the second phase

No. of test problems	GAMS		Proposed heuristic-based SA (400 iterations)		
	Best solution	Time (s)	Best solution	Time (s)	Gap (%)
1	92848	41	95100.6	38.5	1.22
2	170552.2	138.2	178544.8	42	4.69
3	250507.6	152	254265.2	45	1.50
4	310597	156	317483.6	44	2.22
5	360578.4	198	373576	47	3.60
6	448172.2	205.9	480480	50.5	7.21
7	482970.6	349.5	504180.6	56.4	4.39
8	558970	370.6	579090	52	3.60
Average	271515.4	201.4	282692.5	46.9	3.70

TABLE 6. Results in large-sized test problems for the first phase

No. of test problems	No. of suppliers	No. of cross-docking facilities	No. of retailers	GAMS		Proposed heuristic-based SA (400 iterations)		Proposed heuristic-based SA (600 iterations)	
				Best solution	Time (s)	Best solution	Time (s)	Best solution	Time (s)
1	25	10	30	-	-	47136	305.5	46998	340.5
2	30	12	35	-	-	56966	335.6	54901	493.4
3	35	15	40	-	-	51171.2	374.1	50241.6	508.7
4	40	18	45	-	-	63030	412.2	62563.2	490.9
5	45	20	50	-	-	66592	327	63792	452.4
6	50	22	55	-	-	75694.4	449.7	75032	589
7	55	24	60	-	-	72683	454.3	71808	508
8	55	26	65	-	-	81265	462	80012.3	513.5
9	60	29	74	-	-	82006.2	451	80374.4	484.9
10	65	34	76	-	-	85798.9	478.8	83998	631
Average	46	21	53	-	-	68234.3	405	66972.1	501.2

TABLE 7. Results in large-sized test problems for the second phase

No. of test problems	No. of suppliers	No. of cross-docking facilities	No. of retailers	GAMS		Proposed heuristic-based SA (400 iterations)		Proposed heuristic-based SA (600 iterations)	
				Best solution	Time (s)	Best solution	Time (s)	Best solution	Time (s)
1	25	10	30	-	-	620386.8	381	552648	577.2
2	30	12	35	-	-	673271	448.4	601915.7	622.6
3	35	15	40	-	-	677230	407.7	576737.5	594.2
4	40	18	45	-	-	618782.3	474.6	561984.3	654.8
5	45	20	50	-	-	715091.7	488	611115.8	611
6	50	22	55	-	-	738085.5	475.8	655716.5	636.4
7	55	24	60	-	-	773760.9	526.3	644447	688
8	55	26	65	-	-	852403.6	605.8	832180.7	672.9
9	60	29	74	-	-	871999.4	608	833414	722.9
10	65	34	76	-	-	890100	618.5	884817	731.3
Average	46	21	53	-	-	743111.1	503.4	675497.7	651.1

For small-sized test problems, the reported gap in Tables 4 and 5 are calculated as below which denotes the gap between the optimal solutions and solutions obtained by the proposed heuristic-based SA algorithm by:

$$\frac{obj_{heuristic} - obj_{optimal\ solution}}{obj_{optimal\ solution}} \times 100. \quad (53)$$

Some parameters randomly generated in uniform distributions for the large-sized test problems similar to small-sized test problems as provided in Tables 6 and 7. The average time of the proposed heuristic-based SA for ten large-sized test problems for the first phase in 400 and 600 iterations are 405 (s) and 501.2 (s), respectively (see Table 6). In addition, the average time of the heuristic-based SA for the second phase in 400 and 600

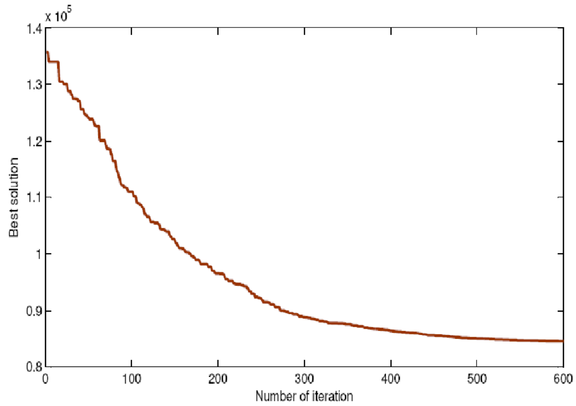

Figure 4. Convergence rate for the tenth large-sized test problem in the first phase

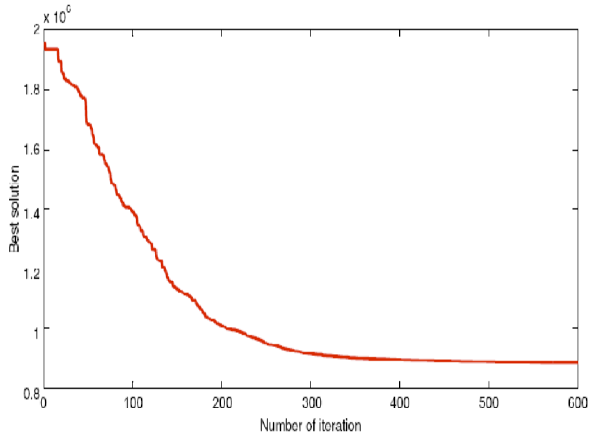

Figure 5. Convergence rate for the tenth large-sized test problem in the second phase

iterations are 503.4 (s) and 651.1 (s), respectively (See Table 7). The best results are provided for 600 iterations in all large-sized test problems. The proposed heuristic run time is acceptable for solving these test problems. For the first phase of the proposed MIP model, the maximum run time in 400 and 600 iterations are 478.8 (s) and 631 (s), respectively, and for the second phase the maximum run time are 618.5 (s) and 731.3 (s) for tenth large-sized test problem. Also, the convergence rates of the proposed heuristic are depicted in Figures 4 and 5 for the tenth test problem.

6. CONCLUSIONS

Logistics managers require making appropriate decisions regarding both qualitative and quantitative aspects to improve the design of the cross-docking distribution networks. Appropriate decisions must be particularly made concerning the location of cross-docking facilities, the structure of the fleet and the strategies to satisfy customers' requirements with their services. It can be conducted within the framework, in which the location and routing scheduling problems through the cross-docking are studied. This paper introduced a new two-phase mixed-integer programming (MIP) model for the location of cross-docking facilities and vehicle routing scheduling in the cross-docking distribution networks. Then, a new heuristic-based simulated annealing (SA) was presented for these problems by considering a special solution representation scheme. The heuristic, hybridized location and routing scheduling decisions, is needed in order to provide high quality solutions with reasonable computational time. To verify the proposed heuristic-based SA, seven small-sized test problems were solved by GAMS software. The computational results obtained by the heuristic were efficient approaching to the optimal solution. The average gap between the proposed heuristic algorithm and GAMS solutions for the first and second phases was equal to 3.22% and 3.70% illustrating the acceptable results, respectively. In addition, the proposed-solving approach was employed to solve the presented MIP model for ten large-scale instances. These solutions demonstrated that the presented MIP model was verified, and the proposed heuristic-based SA provided as an effective problem-solving approach in term of solutions quality and computational time. For the future research, the proposed model can be presented under uncertainty, particularly for the influential parameters that can be provided in fuzzy or stochastic values. It can exert considerable adverse influences on important decisions made by top managers in the cross-docking distribution networks.

7. ACKNOWLEDGMENTS

This work has been partially supported by the Center for International Scientific Studies & Collaboration (CISSC) and the French Embassy in Tehran. The authors are also grateful for the financial support from the Égide Program in France.

8. REFERENCES

1. Jayaraman, V. and Ross, A., "A simulated annealing methodology to distribution network design and management", ***European Journal of Operational Research***, Vol. 144, No. 3, (2003), 629-645.
2. Liao, C.-J., Lin, Y. and Shih, S. C., "Vehicle routing with cross-docking in the supply chain", ***Expert Systems with Applications***, Vol. 37, No. 10, (2010), 6868-6873.

3. Mousavi, S. M., Tavakkoli-Moghaddam, R. and Jolai, F., "A possibilistic programming approach for the location problem of multiple cross-docks and vehicle routing scheduling under uncertainty", ***Engineering Optimization***, Vol. 45, No. 10, (2013), 1223-1249.
4. Mousavi, S. M., Tavakkoli-Moghaddam, R., Siadat, A. and Vahdani, B., A hybrid simulated annealing algorithm for location of cross-docking centers in a supply chain, in Hybrid metaheuristics., Springer. (2013), 12-21.
5. Lee, Y. H., Jung, J. W. and Lee, K. M., "Vehicle routing scheduling for cross-docking in the supply chain", ***Computers & Industrial Engineering***, Vol. 51, No. 2, (2006), 247-256.
6. Li, Y., Lim, A. and Rodrigues, B., "Crossdocking—jit scheduling with time windows", ***Journal of the Operational Research Society***, Vol. 55, No. 12, (2004), 1342-1351.
7. Ross, A. and Jayaraman, V., "An evaluation of new heuristics for the location of cross-docks distribution centers in supply chain network design", ***Computers & Industrial Engineering***, Vol. 55, No. 1, (2008), 64-79.
8. Musa, R., Arnaout, J.-P. and Jung, H., "Ant colony optimization algorithm to solve for the transportation problem of cross-docking network", ***Computers & Industrial Engineering***, Vol. 59, No. 1, (2010), 85-92.
9. Van Belle, J., Valckenaers, P. and Cattrysse, D., "Cross-docking: State of the art", ***Omega***, Vol. 40, No. 6, (2012), 827-846.
10. Liao, T., Egbelu, P. and Chang, P.-C., "Simultaneous dock assignment and sequencing of inbound trucks under a fixed outbound truck schedule in multi-door cross docking operations", ***International Journal of Production Economics***, Vol. 141, No. 1, (2013), 212-229.
11. Boysen, N. and Fliedner, M., "Cross dock scheduling: Classification, literature review and research agenda", ***Omega***, Vol. 38, No. 6, (2010), 413-422.
12. Acar, K., Yalcin, A. and Yankov, D., "Robust door assignment in less-than-truckload terminals", ***Computers & Industrial Engineering***, Vol. 63, No. 4, (2012), 729-738.
13. Shakeri, M., Low, M. Y. H., Turner, S. J. and Lee, E. W., "A robust two-phase heuristic algorithm for the truck scheduling problem in a resource-constrained crossdock", ***Computers & Operations Research***, Vol. 39, No. 11, (2012), 2564-2577.
14. Choi, T.-M., "Vehicle routing problem for fashion supply chains with cross-docking", ***Mathematical Problems in Engineering***, Vol. 2013, (2013).
15. Konur, D. and Golias, M. M., "Cost-stable truck scheduling at a cross-dock facility with unknown truck arrivals: A meta-heuristic approach", ***Transportation Research Part E: Logistics and Transportation Review***, Vol. 49, No. 1, (2013), 71-91.
16. Clarke, G. and Wright, J., "Scheduling of vehicles from a central depot to a number of delivery points", ***Operations Research***, Vol. 12, No. 4, (1964), 568-581.
17. Reinelt, G., "The traveling salesman: Computational solutions for tsp applications", Springer-Verlag, (1994).
18. Hajek, B., "A tutorial survey of theory and applications of simulated annealing", in Decision and Control, 24th IEEE Conference on. Vol. 24, (1985), 755-760.

Optimally Design of the Cross-docking in Distribution Networks: Heuristic Solution Approach

S. M. Mousavi [a,b], R. Tavakkoli-Moghaddam [a], A. Siadat [b]

[a] *School of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran*
[b] *LCFC, Arts et Métier Paris Tech, Metz, France*

P A P E R I N F O

Paper history:
Received 30 March 2013
Received in revised form 17 May 2013
Accepted 22 August 2013

Keywords:
Logistics Management
Cross-Docking Distribution Networks
Mixed-Integer Programming (MIP) Model
Heuristics
Simulated Annealing

چکیده

انبارهای متقاطع نقش اساسی در شبکه‌های توزیع ایفا می‌کنند. در سالهای اخیر مساله طراحی شبکه‌های توزیع انبارداری متقاطع به عنوان یک زمینه تحقیقاتی جدید معرفی می‌شود. این مقاله یک مدل ریاضی جدید برای مسائل مکان‌یابی تسهیلات انبارهای متقاطع و زمانبندی مسیریابی وسایل نقلیه در شبکه‌های توزیع ارائه می‌کند. برای این منظور یک مدل برنامه‌ریزی عدد صحیح مختلط دو فازی فرموله می‌گردد. سپس یک روش جدید ابتکاری مبتنی بر شبیه‌سازی تبرید برای حل این مدل برنامه-ریزی عدد صحیح مختلط توسعه می‌یابد. سرانجام الگوریتم ابتکاری ارائه شده برای مسائل در ابعاد پایین و بالا تست می‌شود. نتایج محاسباتی برای مسائل در اندازه‌های مختلف نشان می‌دهد که این الگوریتم پیشنهادی به طور موثر در زمان قابل قبول عمل می‌کند.

doi: *10.5829/idosi.ije.2014.27.04a.04*