Iranian Journal of Operations Research
Vol. 8, No. 1, 2017, pp. 02-14

A Bi-Objective Green Truck Routing and Scheduling Problem in a Cross Dock with the Learning Effect

M.M. Musavi[1], R. Tavakkoli-Moghaddam[2,3,*], F. Rayat[4]

We present a bi-objective model for a green truck scheduling and routing problem at a cross-docking system. This model determines three key decisions at the cross dock: (1) defining a sequence and schedule of inbound trucks at the receiving door, (2) specifying a sequence and a schedule of outbound trucks at the shipping door, and (3) determining the routes of the outbound truck while serving customers. The first objective function is related to responsiveness of the network that minimizes time window violations and the second objective function minimizes total fuel consumption of trucks in order to consider the environmental factor of the network. Also, a learning effect is considered in loading and unloading process times. To solve the bi-objective model, an archived multi-objective simulated annealing (AMOSA) is used and modified. Finally, a number of test problems are solved and the efficiency of the proposed AMOSA is compared with the ε-constraint method.

***Keywords**: Green truck routing and scheduling, Cross docking, Learning effect, Meta-heuristic algorithm.*

Manuscript was received on 11/05/2017 revised on 28/11/2017 and accepted for publication on 30/11/2017.

1. Introduction

A cross dock (CD) is a more efficient technique as compared to traditional warehouses by implementing five basic activities: receiving, sorting, storing, picking and shipping [24]. It consolidates and carries the orders with fully loaded trucks. So, an advantage of a CD is cost saving because of reducing the amount of inventory items due to a fast distribution system. Holding the stocks more than 24 hours is forbidden in a CD system. Actually a CD is a temporary storage until finished fulfillment of distribution. At a traditional CD center, scheduling and planning of the trucks' route are implemented separately. However, here, we consider these two problems simultaneously. It is obvious that CD centers lead to minimization of transportation cost because of avoiding the additional shipment.

Among environmental, social and economic factors, the environmental factor sometimes becomes more significant in real circumstances. So, we consider the environmental factor in our model by considering fuel consumption for trucks.

* Corresponding author.
[1] School of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran, Email: mohammad.musavi@ut.ac.ir.
[2] School of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran, Email: tavakoli@ut.ac.ir.
[3] LCFC, Arts et Métier Paris Tech, Metz, France, E-mail: reza.tavakkolimoghaddam@ensam.eu.
[4] School of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran, Email: farnazrayat@ut.ac.ir.

[Downloaded from iors.ir on 2026-11-10]

[DOI: 10.29252/iors.8.1.2]

The first study of a learning effect was presented by Wright [22]. We understand that the learning curve is an S-curve including three phases, namely incipient, learning and maturity, and that the learning progress happens rapidly compared to other phases [9]. The processing time of inbound and outbound trucks decreases due to gained experience and the number of repletion of the activities made by humans. This concept is called improvement by doing.

We consider a capacitated truck scheduling and routing problem at a cross-docking center, in which there is a learning effect for unloading/loading processes in a receiving/shipping door. Delivery time window is determined by one of the customers and minimizing the penalty of earliness/tardiness is the first objective. The second objective is to minimize fuel consumption of all trucks for obtaining a green model. Consequently, we develop a mathematical model by these assumptions. The structure of the rest of the paper is as follows. The next section provides a literature review of related papers. In Section 3, the problem is described and a mathematical model is presented. A solution procedure is discussed in Section 4 and the computational experiments are performed in the Section 5. Finally, conclusion is provided in the Section 6.

2. Literature Review

Many studies have been made on CD scheduling problem with different assumptions. We refer to some of them that are more related to our work. Soltani and Sadjadi [20] presented a truck scheduling model in a CD system. Their objective function was to minimize the total flow time. Because of NP-hardness of the problem, they developed two hybrid meta-heuristics, namely simulated annealing (SA) and variable neighborhood search (VNS), to obtain the best sequence of trucks. Also, they generated large-sized instances to illustrate the effectiveness of the proposed algorithms. Boloori Arabani et al. [2] considered a CD approach in their model for reducing inventories, lead time and customer response time. Minimizing the total operation time was the main objective of their work. They proposed some meta-heuristics for solving large-sized problems and compared their results for achieving the best sequence of trucks.

Mousavi and Tavakkoli-Moghaddam [17] considered location and routing scheduling decisions in a CD system and presented a two-stage mixed-integer programming (MIP) model to minimize total costs. They proposed a new algorithm based on a two-stage hybrid SA algorithm for solving small and large-scaled problems efficiently in a reasonable time. Shahin Moghadam et al. [16] utilized a cross-docking concept in their model for making a more efficient flow time between suppliers and customers and considered a set of homogeneous vehicles with limited capacity for transferring products. The customers specified time windows for satisfying their demand in an appropriate time. They obtained SA and a hybrid meta-heuristic algorithm (i.e., ant colony system and SA) as solution methods for solving a mixed integer nonlinear problem. A truck scheduling problem in a multi-door CD system was introduced by Ye et al. [1]. The objective of their work was determining the door assignment and the appropriate sequence for inbound and outbound trucks to minimize the makespan and to meet the requirements. They developed a scatter search (SS) method to solve the generated instances and evaluate the performance of SS in comparison with the genetic algorithm (GA). SS obtained better solutions in a reasonable time.

Dondo and Cerdá [7] developed a mixed-integer linear programming (MILP) model for a vehicle routing problem (VRP) with a CD problem. The model specified the door assignment, scheduling and routing of the vehicles, desired sequence of the heterogeneous vehicles and the required travel times for delivering the products to the assigned door. They proposed a branch-and-cut method for solving the mentioned problem in an acceptable time. Just-in-time (JIT) philosophy was utilized in

[Downloaded from iors.ir on 2026-11-10]

[DOI: 10.29252/iors.8.1.2]

truck scheduling problem to deliver the customers' demands on-time by Assadi and Bagheri [3], who considered ready time for trucks and transshipment time between receiving and shipping doors. A mixed integer programming (MIP) model was presented to minimize total earliness and tardiness of outbound trucks. They solved small-scale problems with CPLEX solver and developed two meta-heuristics, namely differential evolution (DE) and population-based SA to solve medium to large-sized instances. At last, the results of meta-heuristics were compared to the CPLEX solver for evaluating their performance. Keshtzari et al. [11] introduced an improved mathematical model to solve scheduling of inbound and outbound truck problems in CD systems having small sizes by using commercial optimization solvers. Minimizing the makespan was the objective of the proposed model. They developed a particle swarm optimization (PSO) hybridized with SA for solving large-sized problems.

Biskup [5] examined the learning effect in a single-machine scheduling problem as the first attempt. The different types of the learning effect in scheduling problems was presented for minimizing the deviation from a common due date and minimizing the sum of flow time. Actually, the main reason for using this approach was to decrease the job processing time by repeating a process due to gaining experience. Qian and Steiner [19] considered a single machine scheduling problem with the learning/deterioration effect, time-dependent processing times and due date for each job. The objective of their model was to minimize the number of tardy jobs. They used polynomial-time algorithms for all instances.

Li et al. [13] presented a single-machine scheduling problem based on a common flow allowance. All jobs had slack due windows and there was learning effect concept for processing time and the resource were non-renewable. The main goal was to specify appropriate due window, resource allocation and the sequence of processing for minimizing total costs (i.e., earliness, tardiness, the window location, window size, makespan and resource consumption). They utilized a polynomial time algorithm for a linear or a non-linear function of allocation. A single-machine scheduling problem was introduced by Niu et al. [18] involving the learning and deterioration effect. Actually, the processing time depended on a function of learning effect, job's starting time and the specific control parameter. They solved the problem in a polynomial time. Minimizing the makespan, the total completion time, the total waiting time and their deviations, the earliness and the tardiness penalty were the main terms of the objectives. Liu et al. [14] proposed a single machine scheduling problem with resource-dependent processing times and assignment of the due window. They specified the limit for earliness, tardiness, window location, window size and makespan as model's constraints in order to minimize the total resource consumption cost. They considered two methods for the due window assignment, namely the common flow allowance due window assignment and the common due window assignment. They solved the expressed problem in a polynomial time.

Solving multi-objective scheduling problems with meta-heuristics have attracted many researchers in recent years. Darvish et al. [6] proposed a bi-objective open shop scheduling problem. They considered processing times and due dates as fuzzy parameters and proposed the MOPSO meta-heuristic as the solution method. Tavakkoli-Moghaddam and Amin-Tahmasbi [21] developed a bi-objective permutation flowshop problem with sequence-dependent setup time. A multi-objective immune system was proposed for the solution procedure. Xu et al. [23] designed a multi-objective iterative local search for a multi-objective permutation flowshop scheduling problem with sequence-dependent setup times. Khalili and Tavakkoli-Moghaddam [10] proposed a new electromagnetic-like method based multi-objective meta-heuristics for a flexible scheduling problem. Fakhrzad and Emami [8] presented a fuzzy multi-objective linear model for solving a new job shop scheduling problem and developed an approach with respect to the overall acceptable degree of the decision maker satisfaction.

[Downloaded from iors.ir on 2026-11-10]

[DOI: 10.29252/iors.8.1.2]

As we analyze the gap of models in CD scheduling and routing, to the best of our knowledge, there is no paper consideration of the environmental factor in CD problems. Also, there is lack of solution procedures for multi-objective problems. Furthermore, we consider the responsiveness, environmental factor, learning effect and routing of trucks in our model simultaneously. Providing a fast and effective meta-heuristic algorithm for the problem is another motivation for leading proper Pareto solutions.

3. Problem Definition

This section presents a mathematical model of the multi-objective CD routing and scheduling problem with the learning effect and environmental factor of outbound trucks while serving customers. There are several customers each one having a pre-known demand for each product type and a time window to receive the products. Assume a cross-dock with only one receiving door and one shipping door that consolidates the incoming products, sorts them and distributes them to the outgoing trucks. Each incoming (inbound) truck that docks at the receiving door contains quantities of each product type. Regarding capacities of the outgoing (outbound) truck and the demands and time windows of customers, products are distributed to the outbound trucks and each truck delivers the products to customers via routes.

Two objectives are considered as follows: (1) responsiveness and (2) environmental factor of the network. Therefore, the first objective is to minimize any earliness or tardiness of the outbound trucks, and the second objective is to minimize the total fuel consumption of the outbound trucks. Fig. 1 illustrates our considered CD network.

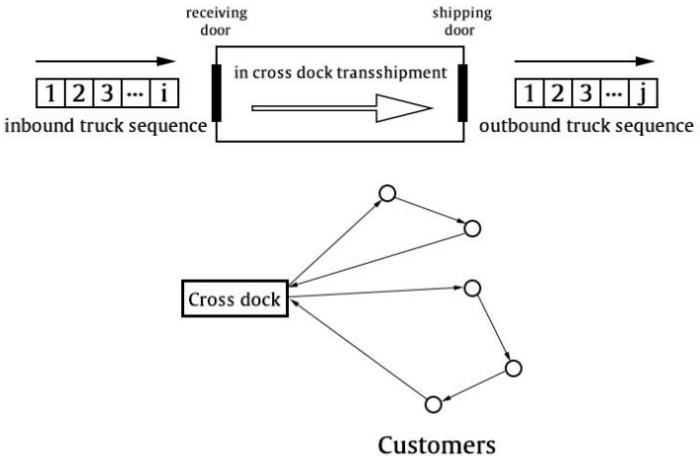


Figure 1. Cross-dock sequences and network of the problem

The notations of the mathematical model are as follows:

Sets:

R	Number of inbound trucks indexed by i and ii
S	Number of outbound trucks indexed by j and jj
N	Number of customers indexed by a and b, where $a = 1$ and $a = n$ determine the CD
K	Number of item types indexed by k.

[Downloaded from iors.ir on 2026-11-10]

[DOI: 10.29252/iors.8.1.2]

Parameters:

r_{ik}	Number of units item type k in inbound truck i
μ_s	Earliness penalty coefficient
μ_f	Tardiness penalty coefficient
$speed$	Average truck speed
h_{ab}	Travel time between customer a and b
φ_0	Fuel consumption rate of truck when empty loaded
φ^*	Fuel consumption rate of truck when fully loaded
cap_j	Capacity of outbound truck j
s_{ak}	Demand of customer a for item type k
M	A significantly big number
D	Changeover time of truck
t_k^{in}	Unloading time of item type k
t_k^{out}	Loading time of item type k
α	Learning rate
V	Operational time at CD
S_a	Start of time window for customer a
F_a	End of time window for customer a.

Variables:

c_i	Entering time of inbound truck i at receiving door
f_i	Leaving time of inbound truck i at receiving door
d_j	Entering time of outbound truck j at shipping door
l_j	Leaving time of outbound truck j at receiving door
x_{ijk}	Number of items of type k transferred from inbound truck i to outbound truck j
g_{aj}	Leaving time of outbound truck j after visiting customer a
ss_{jk}	Number of units of item type k in outbound truck j
E_a	Amount of earliness time for customer a
T_a	Amount of tardiness time for customer a
UL_{jak}	Number of units of item type k in outbound truck j after visiting customer a
u_{ij}	A binary variable that is 1 if any item is transfered from inbound truck i to outbound truck j, and 0, otherwise
$p_{i,ii}$	A binary variable that is 1 if inbound truck i precedes inbound truck ii, and 0, otherwise
$q_{j,jj}$	A binary variable that is 1 if outbound truck j precedes outbound truck jj, and 0, otherwise
z_{abj}	A binary variable that is 1 if outbound tuck j travels from customer a to b, and 0, otherwise
y_{aj}	A binary variable that is 1 if outbound truck j services customer a, and 0, otherwise.

The proposed mathematical model is

$$\text{Min } Obj1 = \mu_s \sum_{a=2}^{n-1} E_a + \mu_f \sum_{a=2}^{n-1} T_a \quad (1)$$

[Downloaded from iors.ir on 2026-11-10]

[DOI: 10.29252/iors.8.1.2]

$$\text{Min } Obj2 = speed \times \sum_{a=1}^{n}\sum_{b=1}^{n}\sum_{j=1}^{J} h_{ab}\left[\varphi_0 z_{abj} + \left(\frac{\varphi^* - \varphi_0}{cap_j}\right)\left(\sum_{k=1}^{K} UL_{jak}\right)\right] \quad (2)$$

s.t.

$$\sum_{j=1}^{J} x_{ijk} = r_{ik}, \qquad \forall i,k \quad (3)$$

$$\sum_{i=1}^{I} x_{ijk} = ss_{jk}, \qquad \forall j,k \quad (4)$$

$$x_{ijk} \le M u_{ij}, \qquad \forall i,j,k \quad (5)$$

$$f_i \ge c_i + \sum_{k=1}^{K}\left[r_{ik} t_k^{in}\left(1 + \sum_{ii=1}^{I}\left(1 - p_{i,ii}\right)\right)^{\alpha}\right], \qquad \forall i \quad (6)$$

$$c_{ii} \ge f_i + D - M\left(1 - p_{i,ii}\right), \qquad \forall i, ii\ ;\ i \ne ii \quad (7)$$

$$c_i \ge f_{ii} + D - M\left(1 - p_{i,ii}\right), \qquad \forall i, ii\ ;\ i \ne ii \quad (8)$$

$$p_{ii} = 0 \qquad \forall i \quad (9)$$

$$l_j \ge d_j + \sum_{k=1}^{K}\left[ss_{jk} t_k^{out}\left(1 + \sum_{jj=1}^{J}\left(1 - q_{j,jj}\right)\right)^{\alpha}\right], \qquad \forall j \quad (10)$$

$$d_{jj} \ge l_j + D - M\left(1 - q_{j,jj}\right), \qquad \forall j, jj\ ;\ j \ne jj \quad (11)$$

$$d_j \ge l_{jj} + D - M\left(1 - q_{j,jj}\right), \qquad \forall j, jj\ ;\ j \ne jj \quad (12)$$

$$q_{jj} = 0, \qquad \forall j \quad (13)$$

$$l_j \ge c_i + V + \sum_{k=1}^{K}\left[x_{ijk}\left(t_k^{in}\left(1 + \sum_{ii=1}^{I}\left(1 - p_{i,ii}\right)\right)^{\alpha} + t_k^{out}\left(1 + \sum_{jj=1}^{J}\left(1 - q_{j,jj}\right)\right)^{\alpha}\right)\right] - M\left(1 - u_{ij}\right), \qquad \forall i,j \quad (14)$$

$$E_a \ge S_a - g_{aj} - M\left(1 - y_{aj}\right), \qquad a = 2,\dots,n-1; \forall j \quad (15)$$

$$T_a \ge g_{aj} - F_a - M\left(1 - y_{aj}\right), \qquad a = 2,\dots,n-1; \forall j \quad (16)$$

$$\sum_{k=1}^{K}\sum_{a=1}^{n} s_{ak} y_{aj} \le cap_j, \qquad \forall j \quad (17)$$

[Downloaded from iors.ir on 2026-11-10]

[DOI: 10.29252/iors.8.1.2]

$$\sum_{j=1}^{J}\sum_{\substack{a=1\\a\neq b}}^{n} z_{abj} = 1, \qquad b = 2, \dots, n-1 \qquad (18)$$

$$\sum_{b=2}^{n-1} z_{1bj} = 1, \qquad \forall j \qquad (19)$$

$$\sum_{a=2}^{n-1} z_{a,nj} = 1, \qquad \forall j \qquad (20)$$

$$\sum_{a=1}^{n-1} z_{aej} = \sum_{b=2}^{n} z_{ebj}, \qquad e = 2, \dots, n-1; \forall j \qquad (21)$$

$$g_{bj} \geq g_{aj} - h_{ab} - M\left(1 - z_{abj}\right), \qquad \begin{matrix} a = 1, \dots, n-1; \\ b = 2, \dots, n; \forall j;\ a \neq b \end{matrix} \qquad (22)$$

$$g_{1j} = l_j, \qquad \forall j \qquad (23)$$

$$\frac{1}{M}\sum_{\substack{a=1\\a\neq b}}^{n-1} z_{abj} \leq y_{bj} \leq \sum_{\substack{a=1\\a\neq b}}^{n-1} z_{abj}, \qquad b = 2, \dots, n-1; \forall j \qquad (24)$$

$$\sum_{a=1}^{n} s_{ak} y_{aj} = ss_{jk}, \qquad \forall j, k \qquad (25)$$

$$UL_{j1k} = ss_{jk}, \qquad \forall j, k \qquad (26)$$

$$UL_{jak} - UL_{jbk} \leq s_{bk} + M\left(1 - z_{abj}\right), \qquad \forall j, a, b, k \qquad (27)$$

$$c_i, f_i, d_j, l_j, x_{ijk}, g_{aj}, ss_{jk}, E_a, T_a, UL_{jak} \geq 0,$$

$$u_{ij}, p_{i,ii}, q_{j,jj}, z_{abj}, y_{aj} \in \{0,1\} \qquad \forall i, j, a, b, k. \qquad (28)$$

The first objective (1) is to minimize the total earliness or tardiness penalties of each customer. The second objective (2) is to minimize the total fuel consumption of the outbound trucks. The first term of this objective calculates fuel consumption when the truck is empty. The second term is related to fuel consumption of truck when they are loaded. Constraints (3) and (4) calculate the number of each product type to be transferred from inbound trucks to outbound trucks. Constraint (5) guarantees that there is transshipment between trucks if any product is transferred between them. Constraint (6) calculates the leaving time of inbound trucks. Constraints (7) and (8) determine entering time of each inbound truck. Constraint (9) guarantees that there is no sequence between a truck and itself. Constraints (10) to (13) are similar to constraints (7) to (9); however, they are for outbound trucks. Constraint (14) explains the relation of entering time of inbound trucks and leaving time of outbound trucks. Constraints (15) and (16) relate to time windows of the customers. Constraint (17) assures that the capacity of each outbound vehicle is not violated. Constraints (18) to (23) relate to vehicle routing of outbound trucks. Constraints (18) to (21) are well-known VRP constraints and constraints (22) and (23) calculate leaving time of each outbound truck after visiting customers. Constraint (24) guarantees that if there is a route between two customers by a specific vehicle, then those customers

[Downloaded from iors.ir on 2026-11-10]

[DOI: 10.29252/iors.8.1.2]

are visited by the vehicle. Constraint (25) determines the number of products in each outbound vehicle regarding the demand of customers that this vehicle serves. Constraints (26) and (27) calculate load of each vehicle for each item type after visiting customers. Finally, constraint (28) presents the positive variables and integer variables.

The learning effect is considered in loading and unloading times for trucks. The operating time for humans to manipulate the products decreases when they repeat the same process. This effect is implemented by

$$p_{l[k]} = p_l \times k^{\alpha}, \qquad -1 < \alpha < 0, \qquad \forall j, a, b, k, \tag{29}$$

where $p_{l[k]}$ is the processing time of job l in position k and α is rate of learning. This is incorporated in the model by relations (6), (10) and (14) to calculate processing of loading and unloading times more precisely.

4. Solution Approach

In the previous section, we proposed a multi-objective CD scheduling and routing problem consisting of three sub-problems of scheduling inbound trucks, scheduling outbound trucks and routing outbound trucks. A CD scheduling and routing problem was proved to be an NP-hard problem [12], and so an effective heuristic is needed to solve the problem in reasonable time. Also, our problem is a bi-objective one. Therefore, multi-objective solution methods should be proposed to obtain non-dominated solutions of the problem. In this section, we present the augmented ε-constraint and a modified archived multi-objective simulated annealing (AMOSA) algorithm for the problem.

4.1. Augmented ε-constraint

Mavrotas et al. [15] presented an improved version of the augmented ε-constraint method for multi-objective integer programming problems. In this method, the multi-objective problem is changed to a single-objective problem by adding other objectives to the constraints.

4.2. Proposed AMOSA

This multi-objective meta-heuristic algorithm was presented by Bandyopadhyay et al. [4]. It gathers all non-dominated solutions in an archive. At first, some initial solutions are generated and saved in the archive. Then, a random individual is selected and by some neighborhood search operators, a new individual is generated. The meta-heuristic algorithm compares the domination state of this new individual and other solutions in the archive and it is accepted to be saved in the archive or not due to three different cases of the heuristic algorithm.

4.3. Solution Representation

To present a solution chromosome for the problem, we define three different arrays:

(1) ***Inbound truck sequence:*** a random permutation of numbers from 1 to R is generated, which represents the sequence of the inbound trucks.
(2) ***Outbound truck sequence:*** a random permutation of numbers from 1 to S is generated, which represents the sequence of the outbound trucks.

[Downloaded from iors.ir on 2026-11-10]

[DOI: 10.29252/iors.8.1.2]

(3) ***Customer routes:*** a random permutation of numbers from 1 to $S + N - 1$ is generated, which represents the route of the outbound trucks. The array is separated from numbers greater than N to define route of each vehicle.

Fig. 2 depicts an example of a solution chromosome with 4 inbound trucks, 3 outbound trucks and 6 customers.

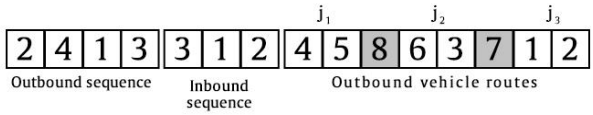


Figure 2. An example of a solution representation

5. Numerical Experiments

To perform numerical experiments on our proposed multi-objective model, we used the GAMS software environment for the augmented ϵ-constraint method. Also, used MATLAB software environment for the AMOSA meta-heuristic algorithm. All the experiments were performed on a PC with 2.5GHz Core i5 CPU and 8GB RAM.

To tune parameters of the AMOSA, we implemented the Taguchi method and the obtained results were 0.98 for temperature damping rate, 15 for HL, 30 for SL and 50 number of iterations. Also, the minimum temperature was set to 1 and maximum temperature was set to 100 for large instances and 50 for medium instances and 20 for small instances. 10 different test problems were designed. The characteristics of these test problems are listed in Table 1. To generate test problems, the parameters of the model were set to the values specified in Table 2.

Table 1. Characteristics of the test problems

Problem	$\|K\|$	$\|R\|$	$\|S\|$	$\|N-2\|$
1	2	3	2	3
2	2	3	2	5
3	3	3	2	5
4	3	4	3	6
5	3	5	3	10
6	4	8	5	15
7	4	10	10	25
8	4	10	10	35
9	5	15	20	50
10	5	15	25	60

[Downloaded from iors.ir on 2026-11-10]

[DOI: 10.29252/iors.8.1.2]

Table 2. Parameters of the test problems

Parameter	Value
r_{ik}	Uniformint(5, 15)
μ_s	0.5
μ_f	0.5
$speed$	40
h_{ab}	Uniform(50, 300)
φ_0	1
φ^*	2
cap_j	Uniform$(2\left\lceil\frac{demand}{\lvert S\rvert}\right\rceil, 4\left\lceil\frac{demand}{\lvert S\rvert}\right\rceil)$
s_{ak}	Uniformint(6, 18)
D	25
t_k^{in}	Uniform(1, 2)
t_k^{out}	Uniform(1.5, 2.5)
α	-0.5
V	10
S_a	Uniform(40, 250)
F_a	Uniform(65, 330)

Table 3 provides the results of test problems obtained by the implementation of AMOSA algorithm. For each test problem, a number of Pareto solutions obtained by the algorithm (NOS) are presented. An example of Pareto solutions for the test problem number 10 is depicted in Fig. 3. The number of non-dominated Pareto solutions achieved by an algorithm is a metric for evaluating its effectiveness. Also, in Table 3, computational CPU times for the test problems are listed. This metric shows that the algorithm computes the results in reasonable times. Fig. 4 illustrates a comparison of AMOSA and the exact method for the CPU time index. As seen, as the problem size increases, the solution time grows exponentially by the exact solution method. For the last two problems, we cannot achieve any solution in a reasonable time. However, the meta-heuristic algorithm provides solutions so rapidly. The best solutions of each objective function for every test problem are listed in Table 3.

Table 3. Results of the test problems

Prob.	NOS	CPU time (sec)	OFV1	OFV2
1	5	18.43	215.26	65232.11
2	5	19.7	677.77	86848.44
3	5	19.85	452.94	100920.25
4	3	20.25	723.2	128597.31
5	7	28.89	2150.71	230471.63
6	9	38.42	4203.8	420422.53
7	10	62.95	12942.12	1284081.51
8	13	76.1	18582.8	1937860.29
9	15	178.89	46239.58	4505192.81
10	15	218.45	63035.83	6580403.13

[Downloaded from iors.ir on 2026-11-10]

[DOI: 10.29252/iors.8.1.2]

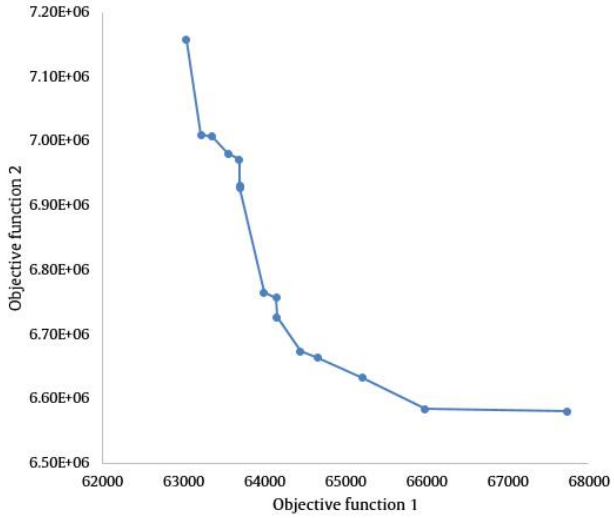


Figure 3. Pareto solutions of test problem No. 10

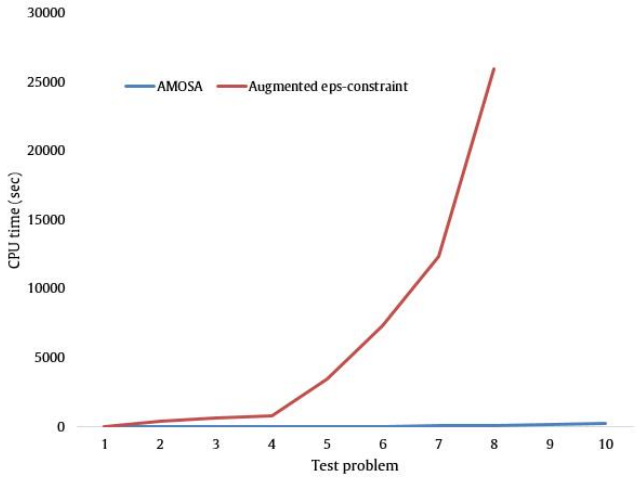


Figure 4. Comparison of CPU times of two solution algorithms

6. Conclusion

A multi-objective cross-docking (CD) scheduling and routing problem was presented. We considered the learning effect for processing times of humans when manipulating the products and the receiving and shipping doors. The CD is considered with only one receiving and shipping door and sequence of inbound and outbound trucks were determined in the model. Also, the routes of the trucks were defined in this model like a vehicle routing problem (VRP). The objectives were to minimized time windows violations and fuel consumptions of the trucks to attend the environmental factor of the network. We proposed an effective multi-objective meta-heuristic algorithm for solving the problem. The presented AMOSA algorithm provided proper solutions and appropriate Pareto solutions. Also, compared to the augmented ε-constraint solution method, the proposed meta-heuristic algorithm resulted in Pareto-optimal solutions with proper computing times. For future

[Downloaded from iors.ir on 2026-11-10]

[DOI: 10.29252/iors.8.1.2]

studies, we suggest the followings: considering routes for inbound trucks, considering social impacts for environmental factor and providing other solution procedures for the problem.

References

[1] Ahmadi, E., M. Zandieh, M. Farrokh, and S.M. Emami (2016), A multi objective optimization approach for flexible job shop scheduling problem under random machine breakdown by evolutionary algorithms. *Computers & Operations Research*, 73, 56-66.

[2] Arabani, A.B., Ghomi, S.F. and Zandieh M. (2011), Meta-heuristics implementation for scheduling of trucks in a cross-docking system with temporary storage. *Expert Systems with Applications*, 38(3), 1964-1979.

[3] Assadi, M.T. and M. Bagheri (2016), Differential evolution and population-based simulated annealing for truck scheduling problem in multiple door cross-docking systems. *Computers & Industrial Engineering*, 96, 149-161.

[4] Bandyopadhyay, S., Saha, S., Maulik, U. and Deb, K. (2008), A simulated annealing-based multiobjective optimization algorithm: AMOSA. *IEEE Transactions on Evolutionary Computation*, 12(3), 269-283.

[5] Biskup, D. (1999), Single-machine scheduling with learning considerations. *European Journal of Operational Research*, 115(1): 173-178.

[6] Darvish, N. (2012), A multi-objective particle swarm optimization algorithm for a possibilistic open shop problem to minimize weighted mean tardiness and weighted mean completion times. *Iranian Journal of Operations Research*, 3(1), 61-88.

[7] Dondo, R. and J. Cerdá (2015), The heterogeneous vehicle routing and truck scheduling problem in a multi-door cross-dock system. *Computers & Chemical Engineering*, 76, 42-62.

[8] Fakhrzad, M.B. and Emami, L. (2016), A fuzzy multi-objective linear programming approach for solving a new multi-objective job shop scheduling with sequence-dependent setup times. *Iranian Journal of Operations Research*, 6(1), 1-14.

[9] Jaber, M.Y., Bonney, M. and Guiffrida, A.L. (2010), Coordinating a three-level supply chain with learning-based continuous improvement. *International Journal of Production Economics*, 127(1), 27-38.

[10] Khalili, M. and Tavakkoli-Moghaddam, R. (2013), A novel adapted multi-objective meta-heuristic algorithm for a flexible bi-objective scheduling problem based on physics theory. *Iranian Journal of Operations Research*, 4(1), 25-38.

[11] Keshtzari, M., B. Naderi, and E. Mehdizadeh (2016), An improved mathematical model and a hybrid metaheuristic for truck scheduling in cross-dock problems. *Computers & Industrial Engineering*, 91, 197-204.

[12] Lee, Y.H., Jung, J.W. and Lee K.M. (2006), Vehicle routing scheduling for cross-docking in the supply chain. *Computers & Industrial Engineering*, 51(2), 247-256.

[13] Li, G., Luo, M.L., Zhang, W.J. and Wang, X.Y. (2015), Single-machine due-window assignment scheduling based on common flow allowance, learning effect and resource allocation. *International Journal of Production Research*, 53(4), 1228-1241.

[14] Liu, L., Wang, J.J. and Wang, X.Y., (2016), Single machine due-window assignment scheduling with resource-dependent processing times to minimise total resource consumption cost. *International Journal of Production Research*, 54(4), 1186-1195.

[15] Mavrotas, G. and Florios, K. (2013), An improved version of the augmented ε-constraint method (AUGMECON2) for finding the exact pareto set in multi-objective integer programming problems. *Applied Mathematics and Computation*, 219(18), 9652-9669.

[Downloaded from iors.ir on 2026-11-10]

[DOI: 10.29252/iors.8.1.2]

[16] Moghadam, S.S., S.F. Ghomi, and B. Karimi (2014), Vehicle routing scheduling problem with cross docking and split deliveries. *Computers & Chemical Engineering*, 69, 98-107.
[17] Mousavi, S.M. and R. Tavakkoli-Moghaddam (2013), A hybrid simulated annealing algorithm for location and routing scheduling problems with cross-docking in the supply chain. *Journal of Manufacturing Systems*. 32(2), 335-347.
[18] Niu, Y.P., Wang, J. and Yin, N. (2015), Scheduling problems with effects of deterioration and truncated job-dependent learning. *Journal of Applied Mathematics and Computing*, 47(1-2), 315-325.
[19] Qian, J. and Steiner, G. (2013), Scheduling with learning effects and/or time-dependent processing times to minimize the weighted number of tardy jobs on a single machine. Mathematical Problems in Engineering.
[20] Soltani, R. and Sadjadi S.J. (2010), Scheduling trucks in cross-docking systems: A robust meta-heuristics approach. *Transportation Research Part E: Logistics and Transportation Review*, 46(5), 650-666.
[21] Tavakkoli-Moghaddam, R. and Amin-Tahmasbi, H. (2012), A Multi-objective immune system for a new bi-objective permutation flowshop problem with sequence-dependent setup times. *Iranian Journal of Operations Research*, 3(2): 66-82.
[22] Wright, T.P. (1936), Factors affecting the cost of airplanes. *Journal of the Aeronautical Sciences*, 3(4), 122-128.
[23] Xu, J., Wu, C.C., Yin, Y. and Lin, W.C. (2017), An iterated local search for the multi-objective permutation flowshop scheduling problem with sequence-dependent setup times. *Applied Soft Computing*, 52, 39-47.
[24] Yu, W. (2002), Operational strategies for cross docking systems.

[Downloaded from iors.ir on 2026-11-10]

[DOI: 10.29252/iors.8.1.2]