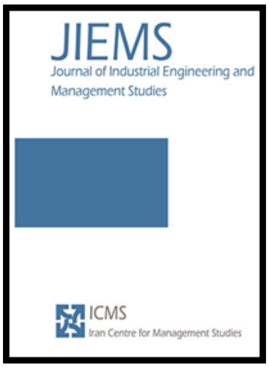
JIEMS
Journal of Industrial Engineering and Management Studies

Vol. 8, No. 1, 2021, pp. 180- 201

DOI: 10.22116/jiems.2021.252150.1395

www.jiems.icms.ac.ir



Developing a mathematical model for a multi-door cross-dock scheduling problem with human factors: A modified imperialist competitive algorithm

Iman Seyedi[1,*], Maryam Hamedi[1], Reza Tavakkoli-Moghaddam[2]

Abstract

This paper deals with optimizing the multi-door cross-docking scheduling problem for incoming and outgoing trucks. Contrary to previous studies, it first considers the simultaneous effects of learning and deteriorating on loading and unloading the jobs. A mixed-integer linear programming (MILP) model is developed for this problem, in which the basic truck scheduling problem in a cross-docking system is strongly considered as NP-hardness. Thus, in this paper, meta-heuristic algorithms namely genetic algorithm, imperialist competitive algorithm, and a new hybrid meta-heuristic algorithm, resulted from the principal component analysis (PCA) and an imperialist competitive algorithm (ICA) called PCICA are proposed and used. Finally, the numerical results obtained from meta-heuristic algorithms are examined using the relative percentage deviation and time criteria. Results show that the hybrid PCICA algorithm performs better than the other algorithms in terms of the solution quality. Computational results indicate when the learning rate increases, its decreasing effect on processing time will growth and the objective function value is improved. Finally, the sensitivity analysis also indicates when the deterioration rate is reduced, its incremental effect is decreased over time.

Keywords: cross-dock scheduling; learning effect; deterioration; meta-heuristic algorithms.

Received: October 2020-20

Revised: December 2020-28

Accepted: January 2021-02

1. Introduction

Meeting customer needs at the right time and place with the least cost is one of the main objectives of supply chain management. Generally, each supply chain is formed by three primary steps including procurement, production, and distribution. The use of the cross-docking system is a novel strategy at the distribution stage to improve customer response time by moving products directly from pickup trucks to delivery trucks. Cross-docks can eliminate storage and selection functions, which have the highest cost of warehouse operations among the main operations of a distribution center (Yu, 2002). Cross-docking is a logistical strategy

* Corresponding author; iman_seyyedi@pnu.ac.ir

[1] Department of Industrial Engineering, Payame Noor University, P.O. Box: 19395-4697, Tehran, Iran.

[2] School of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran.

and acts as a middle node in the supply chain network. The main idea is that products from different suppliers reach the cross-dock through the inbound trucks and unload their products in the receiving docks. After that with the least possible amount of storage and with appropriate categories upon the customer request, products are loaded onto outbound docks (Mohtashami et al. 2013).

A cross-docking system has three main processes. Receiving and unloading products, sorting products according to customers' requirements, loading, and shipping products to the final delivery destination. Failure to handle multiple trucks at the same time in a single inbound door cross-docking system will ultimately slow down the evacuation operations. The same problem arises with the outbound door if the cross-dock storage system has only one shipping door. Essentially, a cross-docking system in the real world includes multiple doors and involves the transfer of several types of products. Therefore, this paper deals with the multi-door and multi-product cross-dock system.

For many decades, one of the main operational decisions in a cross-dock system was truck scheduling problem (Wisittipanich & Hengmeechai, 2017). The truck scheduling problem answers "when" and "where" trucks can do their job well. The purpose of scheduling problems in a cross-docking system is to decide on the sequence of inbound and outbound trucks. If truck scheduling is planned well, it can significantly decrease the cross-docking operation time.

Generally, both machine and human resources are needed for an activity process. Many researchers have already developed numerous planning methods for cross-dock systems; however, they have largely ignored the importance of human resource constraints. Resource planning must consider the unique inherent characteristics of machines and human resources to provide an optimal plan (Seyedi et al. 2016). However, in scheduling problems raised so far, human resource constraints have been ignored and only machines have been considered as a limited resource. However, the ability of a cross-dock system and its success is depending on the capability of resources more than their availability (Fathollahi-Fard et al, 2019).

Unlike machines, human operators can learn and acquire new skills. In the cross-docking research literature, usually processing time considered as a fixed parameter (. However, the processing time of each task usually depends on both positions of that task in a sequence and its starting time.

In the scheduling problem, the processing of a task in subsequent positions (subsequent iterations) by the workforce can reduce the time spent to complete the tasks (Hosseinian et al, 2020). This phenomenon is known as the "learning effect" in the literature. Due to the effects of learning, the processing time of the jobs is less than their normal processing time. Since the most cross-docking tasks (e.g., unloading and loading) are done by labor; therefore, the effect of learning on these problems can be considered. Also, any delay in starting processing leads to increased work time. This phenomenon is known as the "deterioration of tasks" in the literature. According to surveys, the cross-docking literature has not shown and considered any deterioration of jobs so far.

The simultaneous effects of learning and deteriorating jobs may exist in many real-world conditions. Despite this fact, researchers have not considered these effects simultaneously so far. Therefore, in this paper, for multi-door cross-dock scheduling problem, a new mixed-integer programming (MIP) model is presented. In this context, human factors and deterioration of jobs to fill the gap between theoretical planning models, and actual practical work is considered.

The cross-docking system scheduling is a problem with the computational complexity (Sang & Cheng, 2007). Boysen et al. (2010) proved that the truck scheduling problem in a single receiving and shipping door cross-dock is strongly NP-hard. Therefore, it can be concluded that multi-door cross-dock scheduling problem is also NP-hard. Since these problems are complex and their optimal solution times are in the exponential category, in the literature meta-

heuristic algorithms were used to solve them. In this paper, meta-heuristic algorithms (i.e., such as genetic algorithm (GA), Imperialist competitive algorithm (ICA), and a new hybrid meta-heuristic algorithm, called PCICA) are used. Finally, the numerical results obtained from all meta-heuristic algorithms are examined.
The paper is organized as follows. The literature review of cross-docking system scheduling problem is presented in Section 2. In Section 3, the hypotheses of the problem will be comprehensively introduced and the presented mathematical model is explained. Section 3 describes the proposed solution methods. In Section 4, the results and performance of the proposed methods are examined and analyzed. Finally, findings are summarized in Section 5.

2. Literature Review

Yu (2002) presented one of the most important studies in the field of the truck scheduling problem in a cross-dock system. He raised various issues of truck scheduling by focusing on five key elements. These elements include existence or absence of temporary warehouse, the amount of receiving doors, the number of shipping doors, the pattern of truck placement in front of receiving door and pattern of truck placement in front of the shipping door. In general, most studies on the scheduling of trucks in a cross-dock have focused on minimizing the time spent to complete the last outbound truck and determining the inventory levels as an objective function. These findings were presented by Song and Chen (2007), Yu and Egbelo (2008), Hajiaghaei-Keshteli, and Sajadifar (2010), Maknoon and Baptist (2010), Vahdani and Zandieh (2010), Hajiaghaei-Keshteli et al. (2011), Alpan et al. (2011), Bellanger et al. (2013), Heidari et al. (2018), Bakeshloo et al. (2019), Golshahi-Roudbaneh et al. (2019), Baniamerian et al. (2019), and Hasani Goodarzi et al. (2020), Pan et al. (2020).
Yu and Egbelu (2008) presented a mathematical model for a cross-dock system with a receiving and shipping door and a temporary warehouse in front of the shipping door. The aim was to find the best sequence for inbound and outbound trucks when the makespan was minimized. They proposed nine heuristic methods to solve the proposed model and demonstrated the effectiveness of the proposed heuristic methods. Boysen and Fliedner (2010) studied scheduling issues in three aspects: doors, operational specifications, and objective functions and displayed them by $[\alpha/\beta/\gamma]$. α represents the state of the doors, which are divided into four forms, namely E, M, EM, and G. A number may be recorded for each of the above signs indicating the number of doors in the problem. On the other hand, β represents the operational specifications and is divided into 9 general sections. Finally, γ represents the type of objective function.
Vahdani and Zandieh (2010) used five meta-heuristic methods to solve large-sized problems, which are GA, TS, SA, EMA, and VNS. It should be noted that they used a response surface methodology (RSM) method to robust meta-heuristic algorithms. Alpan et al. (2011) studied a scheduling problem in a cross-dock system and assumed that the inbounding trucks are allocated to the doors based on a FIFO policy, and the expected costs are independent of time. In this case, scheduling is only studied for the outbound trucks because the sequence of the inbound trucks is predetermined based on production plans.
Mousavi et al. (2013) presented a two-stage MIP model for the location of cross-dock centers, and vehicle routing planning in distributed networks. They used simulated annealing (SA) and tabu search (TS) algorithms to solve their model. They randomly generated several problems and figured out that the proposed algorithm was effectively and quickly converged with reasonable solutions. Golshahi-Roudbaneh et al. (2017) improved a mathematical model by presenting a new lower bound as well as two new heuristic methods. They also used some meta-heuristics to solve this problem, which yielding acceptable results. Wisittipanich and Hengmeechai (2017) examined the cross-dock scheduling problem with several receiving and

shipping doors. To solve this problem, they introduced a modified particle swarm optimization (PSO) algorithm. They compared the answers to this problem with the main particle swarm method and concluded that the new method had a high ability to solve this problem.

Rijal et al. (2019) provided an integrated approach to solve two problems simultaneously in a cross-dock, in which doors can act as a combination of both receiving and shipping. They proposed an adaptive large neighborhood search algorithm to solve the problem. The results of extensive computational experiments showed that operating costs in a cross-dock are reduced by an average of 12% compared to the best solutions. Also, with an integrated approach, they could identify the number of entrance doors to work in a mixed-mode and the location of these flexible doors in a cross-dock terminal. The numerical results showed that the average cost savings are 9.7% when 60% of the doors are used in the mixed-mode. This savings increase to 12.3% when these doors are located in the center of the cross-dock. Dulebenets (2019) conducted a study to improve the scheduling of inbound and outbound trucks in a cross-dock and proposed a unified linear programming model for truck scheduling to minimize the service cost of trucks. He used a novel Delayed Start Parallel Evolutionary Algorithm to solve the problem. The computational results showed that the superiority of the proposed algorithm regarding key algorithmic criteria compared to the other five algorithms commonly used in cross-dock literature. He also showed how to use this developed algorithm effectively to analyze important management issues in the field of transportation.

Fonseca et al. (2019) conducted a study on the scheduling of trucks in a parallel cross-dock. They first modeled the issue as a two-machine flow shop scheduling problem with priority constraints and to minimize completion time and then generalized it to a cross-dock problem. To solve this problem, they proposed a hybrid method based on the Lagrangian relaxation technique and volume algorithm. Using Lagrangian coefficients, a useful heuristic with local search steps generated acceptable solutions. This algorithm finds good solutions to the problems of small and large sizes. Seyedi et al. (2019) examined the problem of truck scheduling and door-to-door assignment simultaneously and presented robust heuristic methods, which achieve the desired objective with the least time. Their proposed algorithm performed better compared to other heuristic approaches in a similar model. The presented approaches attempt to minimize, or even eliminate them by identifying delays that occur during the implementation of cross-dock processes. Also, due to space limitations of temporary storage, the policy of heuristic methods was proposed to minimize the temporary storage.

Also, due to the use of simultaneous effects of learning and deteriorating jobs in current research some recent studies are considered. Rostami et al. (2015) examined the parallel machines scheduling problem assuming the effects of learning and deterioration jobs under fuzzy environment. The aim of the problem was minimizing the total tardiness and earliness times, and the maximum completion time. Salehi Mir and Rezaian (2016) proposed a new scheduling model with the setup time of the previous sequence, the time of access to the tasks, and the simultaneous effects of learning and deteriorating jobs. They also used a hybrid meta-heuristic algorithm based on PSO and GA to minimize the sum of workloads on unrelated parallel machines. According to the literature, in a cross-docking system, only Amini et al. (2014) worked on one receiving and one shipping doors, which examine only the effect of learning on the process of unloading and loading by workers. They presented a mathematical model according to the existing models. Then, they used four meta-heuristic algorithms along with a SA algorithm to overcome the complexity of major problems. Finally, they compared the performance of the proposed algorithms with the optimal solutions obtained by the complete counting method.

3. Research Methodology

To eliminate shortcomings and adapt the model to real condition the basic mathematical model presented by Yu and Egbello (2008) is developed in this study. In the developed model, multiple receiving and shipping doors are considered. Also, arrival time of different trucks and the unloading and loading time of different goods are different. In traditional cross-dock scheduling problems proposed so far, the processing time of the jobs is independent of the job sequence and considered to be constant. This assumption is often not appropriate because the ability and skill of the worker increase during the work, and as a result, the processing time of work decreases. As well, in the scheduling of production systems, because of "deteriorating jobs", any delay in the start of processing jobs can increase working time. In the presented model, the simultaneous effects of these issues are considered.

3.1. Mathematical modeling

To address deficiencies and adapt the model to real-world situations, a mathematical model is presented based on the existing models. This model provides more than one door for inbound and outbound trucks. Also, truck arrival times are different and different goods have diverse loading and unloading times in the entrance and exiting doors. To adapt the model to real-world conditions, the effects of simultaneous deterioration and learning on processing time are considered in this model. In this model, the actual processing time of tasks is a general function of the task starting time (deterioration effect) and its scheduling position in the sequence (learning effect).

$$P_{j[r]} = P_j r^b + at \tag{1}$$

where $a \geq 0$ and $b \leq 0$ are deterioration and learning effect rates, respectively. P_j is the initial processing time. r is the position of the job j in the sequence. Finally, t is the start time of job j. (Seyedi et al. 2015)

R	Number of inbound trucks (i = 1, …, R)
S	Number of shipping doors (j = 1, …, S)
N	Number of products (k = 1, …, N)
T	Completion time
r_{ik}	Number of products type k in inbound truck i
s_{jk}	Number of products type k in outbound truck j
D	Trucks replacement time
V	Operation time within the cross-dock
M	A positive big number
c_i	inbound truck i arrival time at receiving door
F_i	inbound truck i leaving time of at receiving door
d_j	outbound truck j arrival time at shipping door
L_j	outbound truck j leaving time at shipping door
x_{ijk}	Number of units of product type k transferred from inbound truck i to outbound truck j
v_{ij}	$\begin{cases} = 1, \text{If any products are transported from truck } i \text{ to truck } j \\ = 0, \text{otherwise} \end{cases}$
p_{ij}	$\begin{cases} = 1, \text{If inbound truck } i \text{ precedes inbound truck} \\ = 0, \text{otherwise} \end{cases}$
q_{ij}	$\begin{cases} = 1, \text{If outbound truck } i \text{ precedes outbound truck} \\ = 0, \text{otherwise} \end{cases}$
t_k^I	Duration of product type k unloading from inbound trucks

t_k^O Duration of product type *k* loading in outbound trucks
b Learning rate
a Deterioration rate
A^i Arrival time of inbound truck *i*

$$\text{Min Z } = T \tag{2}$$

s.t.

$$T \geq L_j \qquad \forall\, j \tag{3}$$

$$\sum_{j=1}^{S} X_{ijk} = r_{ik} \qquad \forall \;\; i,k \tag{4}$$

$$\sum_{i=1}^{R} X_{ijk} = S_{jk} \qquad \forall \;\; j,k \tag{5}$$

$$X_{ijk} \leq M v_{ij} \qquad \forall \;\; i,j,k \tag{6}$$

$$\sum_{m=1}^{I} z_{im} = 1 \qquad \forall\, i \tag{7}$$

$$\sum_{n=1}^{O} y_{jn} = 1 \qquad \forall\, j \tag{8}$$

$$\sum_{m=1}^{I} \sum_{n=1}^{O} b_{ijmn} = v_{ij} \qquad \forall \;\; i,j \tag{9}$$

$$b_{ijmn} \leq z_{im} \qquad \forall \;\; i,j,m,n \tag{10}$$

$$b_{ijmn} \leq y_{jn} \qquad \forall \;\; i,j,m,n \tag{11}$$

$$z_{im} + z_{jm} - 1 \leq P_{ij} + P_{ji} \qquad \forall \;\; i,j,m \; and \; i \neq j \tag{12}$$

$$P_{ij} + P_{ji} \leq 1 \qquad \forall \;\; i,j \; and \; i \neq j \tag{13}$$

$$y_{in} + y_{jn} - 1 \leq q_{ij} + q_{ji} \qquad \forall \;\; i,j,n \; and \; i \neq j \tag{14}$$

$$q_{ij} + q_{ji} \leq 1 \qquad \forall \;\; i,j \; and \; i \neq j \tag{15}$$

$$F_i \geq c_i + \sum_k t_k^I r_{ik} \left(1 + \sum_j \{1 - P_{ij}\}\right)^b + a c_i \qquad \forall\, i \tag{16}$$

$$C_j \geq A_i - M(1 - P_{ij}) \qquad \forall\, i,j \quad and \; i \neq j \tag{17}$$

$$C_i \geq A_i \qquad \forall\, i \tag{18}$$

$$C_j \geq F_i + D - M(1 - P_{ij}) \qquad \forall\, i,j \quad and \; i \neq j \tag{19}$$

$$C_i \geq F_j + D - M P_{ij} \qquad \forall\, i,j \quad and \; i \neq j \tag{20}$$

$$P_{ij} = 0 \qquad \forall\, i \tag{21}$$

$$L_j \geq d_j + \sum_k t_k^O s_{jk}\left(1 + \sum_i \{1 - q_{ij}\}\right)^b + ad_j \qquad \forall\, j \qquad (22)$$

$$d_j \geq L_i + D - M(1 - q_{ij}) \qquad \forall\, i,j \quad and\ i \neq j \qquad (23)$$

$$d_i \geq L_j + D - Mq_{ij} \qquad \forall\, i,j \quad and\ i \neq j \qquad (24)$$

$$q_{jj} = 0 \qquad \forall\, j \qquad (25)$$

$$\begin{aligned} L_j \geq c_i &+ \sum_{m=1}^{I}\sum_{n=1}^{O} T_{mn} b_{ijmn} \\ &+ \sum_k X_{ijk}\left(t_k^O\left(1 + \sum_i \{1 - q_{ij}\}\right)^b + ad_i \right. \\ &\left. + t_k^I\left(1 + \sum_j \{1 - P_{ij}\}\right)^b + ac_j\right) - M(1 \\ &- v_{ij}) \end{aligned} \qquad \forall\, i,j \qquad (26)$$

$$F_i \leq DDate_i; \qquad \forall\, i \qquad (27)$$

$$L_j \leq RDate_j \qquad \forall\, j \qquad (28)$$

$$\sum_k r_{ik} \leq cap_i \qquad \forall\, i \qquad (29)$$

$$\sum_k s_{jk} \leq cap_j \qquad \forall\, j \qquad (30)$$

$$\text{All variables} \geq 0 \qquad (31)$$

In this model, several doors for inbound and outbound trucks are examined while considering the learning effect and deteriorating jobs. Besides, the arrival time of different trucks and the time of unloading and loading of different goods from the receiving and shipping doors are also considered dissimilarly. The objective function is to minimize the time spent to complete the entire operation of the system. In other words, it starts when the first inbound truck unloads its first product and ends when the last product is loaded in the last outbound truck. The first constraint (i.e., Equation (3)) indicates that the total completion time is greater than or equal to the leaving time of the last shipping truck from the sending platform. Constraints (4) and (5) indicate that the total number of items received by the inbound trucks is equal to the total number of items loaded to outbound trucks. Constraint (6) shows the relationship between variables X_{ijk} and v_{ij}. Constraints (7) and (8) indicate that each inbound truck and each outbound truck is assigned to one receiving door and one shipping door respectively. A correct connection between variables b_{ijmn}, v_{ij}, z_{im}, and y_{in} is created in constraints (9) - (11). Constraints (12) and (13) relate variables z_{im} and P_{ij}. Constraints (14) and (15) connect $y_{in}$and q_{ij} variables. Restrictions (16) - (20) specify constraints entry time and exiting of receiving machines based on their orders in the sequence of machines, their entry time, and the rate of learning and deteriorating jobs. Constraint 21 ensures that no entry truck surpasses itself in sequence. Constraints (22) - (24) specify entry time and exiting of receiving trucks based on their orders in the sequence, the rate of learning, and deteriorating jobs. Constraint (25) ensures that no outbound truck outperforms itself in the sequence. Constraint (26) relates the exiting time of each outbound truck and the entry time for each inbound truck for each type of product. Also, in Constraint 26, the different time of sending goods from each *m* receiving door to each n shipping door is considered. Constraints (27) and (28) specify the due date to each truck. Constraints (29) and (30) specify the capacity for each inbound and outbound trucks.

3.2. Meta-heuristic algorithms

The complexity of using mathematical optimization in many large-scale problems and failing to solve difficult problems with numerous variables has led to the development of alternative solutions. Therefore, sometimes in practice, solutions that are obtained through approximate optimization techniques will be satisfactory. Heuristic and meta-heuristic algorithms from approximate optimization methods have become popular in recent years. Boysen et al. (2010) verified that the single door cross-dock scheduling problem is strongly NP-hard. Therefore, it can be concluded that multi-door cross-dock scheduling problem is NP-hard too. Since these problems are complex and their optimal solution time is in the exponential category, meta-heuristic algorithms were used in the literature to solve these problems.

3.2.1. Solution representation

The structure of problem coding used in all meta-heuristic algorithms in this study is illustrated in Figure 1. Dimensions of the solution are the sum of the receiving and shipping trucks and their assigning doors. Let us assume a cross-dock with two input doors, two output doors, four inbound and four outbound trucks. In this case, the dimensions of the answer are 16 because there are eight trucks and two doors. A uniform random number between [0, 1] generates each dimension value initially. As shown in Figure 1, the structure of an answer consists of four parts, namely receiving door, receiving truck, shipping door, and shipping truck. The lower left part of the answer determines the sequence of the receiving trucks, and the right one determines the sequence of the shipping trucks according to the order of ascending values. The upper part of the answer shows the doors assigned to the trucks. Since in this example there are two doors for inbound trucks and two doors for outbound trucks, we assign numbers 1 and 2 respectively, which represent doors 1 and 2, due to the uptrend of random numbers. This coding scheme is used in all metaheuristic algorithms in this study.

0.33	0.54	0.65	0.86	0.95	0.32	0.08	0.25
0.14	0.32	0.65	0.51	0.33	0.7	0.18	0.96

receiving Door	shipping Door
receiving Truck	shipping Truck

Assigned door				Assigned door			
1	2	1	2	2	1	1	2
1	2	4	3	2	3	1	4
receiving truck sequence				shipping truck sequence			

Figure 1. Example of a coding scheme

3.2.2. Genetic algorithm

Holland (1992) first proposed the idea of using a genetic algorithm (GA) for optimization problems. Then, this algorithm was widely used in a wide range of optimization problems. The GA is inspired by the science of genetics and Darwin's theory of evolution and is based on the survival of the fittest or natural selection. It is a population-based algorithm that starts with an initial population of solutions. Then, it creates a new generation that replaces the current population. Each member of the population is called a chromosome. The primary population is often randomly generated. The fit rate of each member of the population (chromosome) is then determined. In optimization problems, the fitness rate is usually determined according to the objective function (Amiri et al., 2020). The current population then evolves by the GA operators. The GA has three operators: selection, crossover, and mutation. The pseudo-code of the GA is summarized in Figure 2.

```
Number of generation 0:
k := 0;
Pk := a population of n randomly-generated individuals
Evaluate individuals of population Pk:
Compute fitness(i) for each i ∈ Pk
While termination criterion not met Do
      Create generation k + 1:
         5.1. Copy: Select (1 − χ) × n members of Pk and insert into
         Pk+1;
         5.2. Crossover: Select χ × n members of Pk; pair them up;
         produce offspring; insert the offspring into Pk+1;
         5.3. Mutate: Select μ × n members of Pk+1; invert a
        randomly-selected bit in each;
        Evaluate Pk+1: Compute fitness(i) for each i ∈ Pk;
        Increment: k := k + 1; }
while a fitness of the fittest individual in Pk is not high enough;
return the fittest individual from Pk
```

Figure 2. GA Pseudo-code

3.2.3. Imperialist competitive algorithm

Most optimization algorithms such as the GA are largely inspired by natural processes. In these algorithms, other aspects of human evolutions are considered. An imperialist competitive algorithm (ICA) is a new optimization algorithm inspired by a human social phenomenon (Atashpaz-Gargari and Lucas, 2007). In particular, this algorithm looks at the colonial process as a stage in human socio-political evolution. By mathematical modeling, this historical phenomenon can be used as a powerful algorithm in the field of optimization. Similar to all algorithms in the category of evolutionary optimization algorithms, the ICA uses a primary set of possible solutions. In this algorithm, these initial solutions are known as “country” and all countries are grouped into several empires (Fard et al., 2017). The ICA with the following specific process gradually improves these initial solutions (countries) and finally provides an appropriate solution to the optimization problem (desirable country). The steps of the ICA are summarized in Figure 3.

Initialization of population.
Do empires formation.
While termination criterion not met Do
For i=1 to Nimp do
Assimilation: move the Colonies toward their relevant imperialist in different directions.
Revolution: Random changes occur in the characteristics of some countries
If there is a colony with a better position than the imperialist exchange their Position.
Imperialistic competition: All imperialists compete to take possession of colonies of each other.
Eliminate the Weak empires that lose their power gradually.
If the stop condition is satisfied, stop, if not go to 2.
End

Figure 3. Imperialist competitive algorithm pseudo-code

3.2.4. Principal component imperialist competitive algorithm

In this paper, a new hybrid algorithm called PCICA for increasing ICA exploration capability is proposed. This algorithm is a combination of the ICA and principal component analysis (PCA). PCA is a powerful technique used for the selection of good features. The PCA is able to handle high-dimensional, noisy and highly correlated data by projecting the data onto a lower-dimensional subspace that constrains most of the variance of the original data. The main purpose of Principal component analysis (PCA) is to convert a set of correlated data x_i to a set of data with minimal correlation z_i. This is done by selecting a set of orthogonal UI vectors, which are used to define new variables using the linear combination of them and the main variables (Huixuan, 2005).
In the ICA, the policy of assimilation updates the position of each country through a linear learning strategy. This strategy uses the previous position of the country and the distance from its imperialist position with a slightly random angle. Despite the simplicity of this strategy in updating the country's position, it is inefficient in multi-dimensional space and has limitations in information discovery. PCA strategy suits to handle the position matrixes of all countries nicely. Thus, in the proposed hybrid algorithm, a new search mechanism based on the PCA is suggested. The PCA optimizes population information to determine the direction of the principal components. The PCICA can store a lot of location information for countries to use to steer them in more promising directions. The steps of the proposed algorithm are shown in Figure 4.

4. Numerical results and analysis

In this section, problems will be first designed to evaluate the performance of the proposed algorithms. We will first design problems to evaluate the performance of the proposed algorithms. Then, the parameters of each algorithm are determined and adjusted. All problems are implemented by algorithms and the results obtained are analyzed using exact solutions. Then, certain comparing criteria are defined, and then the efficiency of algorithms is compared with each other. To compare the performance of meta-heuristic algorithms, 35 small, medium, and large-scale problems designed by Van Bell et al. (2013) were used.

Initialization of the algorithm. Generate some random solution in the search space and create initial empires.
Initialization of the covariance matrix
Calculation of principal components $\acute{u}_i S \, u = L$
Transfer of countries to Z space $z = \acute{U}[x - \bar{x}]$
Calculating the power of all countries in the new space
Colony and imperialist displacement if the power of the colony exceeds that of the imperialist.
Assimilation: Colonies move towards imperialist states in different directions.
Movement of countries in space Z
$$\tilde{z}_{i,j}^{k+1} = \tilde{z}_{i,j}^{k} + \tilde{\beta}.(\tilde{z}_{i,g}^{k} - \tilde{z}_{i,j}^{k}).\tilde{\gamma}.\tilde{U}(-\theta,\theta$$
Revolution: Random changes occur in the characteristics of some countries
If there is a colony with a better position than the imperialist exchange their Position.
Imperialistic competition: All imperialists compete to take possession of colonies of each other.
Eliminate the Weak empires that lose their power gradually.
If the stop condition is satisfied, stop, if not go to 2.
End

Figure 4. Proposed PCICA pseudo-code

4.1. Test problems and parameter setting

Due to the random nature of meta-heuristic algorithms, it is possible for one algorithm to get a good answer in one performance and the same algorithm may get a poor quality answer in another performance. Test design is a way to create the highest efficiency with the least time and cost. Algorithm parameters are set to reduce the likelihood of producing bad solutions and help to produce the right solutions. After determining the number of parameters and their levels, the number of required experiments is determined by Taguchi’s orthogonal arrays. Since the scale of the experimental problems is different, it is not possible to apply the result directly. To eliminate this problem, scaling is done by using the RPD on the values obtained through experiments for each problem (Abdi et al. 2020). The value of the RPD is obtained by:

$$RPD = \frac{Sol_{ij}^{k} - Bestsol^{k}}{Bestsol^{k}} \tag{32}$$

where Sol_{ij}^{k} is the value obtained for problem *k* in the repetition of *j* from the *i* test. $Bestsol^{k}$ is the best value for problem *k*. Figures 5 and 6 show the RPD values obtained for various parameters.

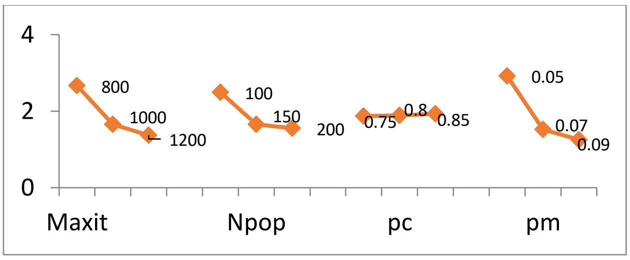


Figure 5. RPD diagram of the GA parameters

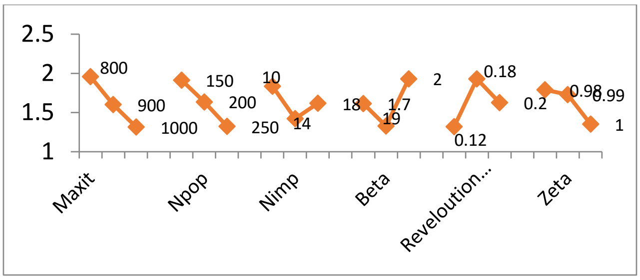


Figure 6. RPD diagram for the ICA parameters

The average RPD is an average response for different combinations of control factor levels in the Taguchi's design. Given the minimization purpose of this research, we determine the level of factor that minimizes the average. For example, in Figure (5) that corresponds to the average RPD of the GA, there are four parameters in three levels. The points indicated in this diagram provide an estimate of the objective function at each level of the parameter. Since the purpose of this paper is to minimize the makespan, so in each parameter the level that has the lowest average is determined. The results of setting the parameters of meta-heuristic algorithms are shown in Table 1.

Table 1. Best level for the parameters of the meta-heuristic algorithms

Algorithm	Parameter	Best level
GA	Max it	1200
	N pop	200
	Pc	0.75
	Pm	0.09
ICA	Max it	1000
	N pop	250
	N imp	14
	Beta	1.9
	Revolution rate(R)	0.12
	Zeta	1

The performance of the proposed algorithms was evaluated using sample problems according to Van Bell et al. (2013). After calculating the value of parameters, each algorithm was executed 30 times. The best and the average results are shown in Tables 2 to 4 according to the different values of the learning and deterioration rates. In these tables, we present the validation of the proposed model. In addition, the sensitivity analysis of the model by changing the values of the input parameters are presented in these tables. The effect of learning and deterioration capabilities is analyzed by changing their rates. For this reason, we change the value of the learning rate from b = -0.1 to b = -0.9. We also change the deterioration rate from α = 0.1 to α = 0.9. Tables 2 to 4 illustrate the computational results obtained by the proposed meta-heuristic algorithm and Lingo software based on different learning and deterioration rates. Lingo 8 software is incapable to solve medium and large-scale problems. However, the proposed meta-heuristic algorithm is used to survey the impact of the learning effect and deteriorating jobs. As the learning rate increases, the processing time will be reduced, so we expect that the value of the objective function will improve. The results in Tables 2 to 4 show that an increase in the learning rate decreases the objective function value. Also, the sensitivity analyses indicate if the deterioration rate decreases, the processing time decreases. Therefore, we expect that the objective function will not worsen. The results presented in Tables 2 to 4 indicate that our expectation is correct. There are several criteria for calculating deviation in statistics. Although the standard deviation and scope were used extensively so far, there are other ways to quantify dispersion. To compare and evaluate the efficiency of meta-heuristic algorithms, the RPD and computational time are used. All sizes of problems are executed 30 times by algorithms and the mean value of the RPD for the obtained answers is shown in Figures 7 to 9. For small samples, our proposed algorithms reached the optimal solution according to the optimal solutions obtained by Lingo software at a reasonable computational time (Tables 2 to 4). The gap obtained from the optimal solution for all meta-heuristics is zero (Figures 7 to 9). For medium-sized problems (Problems 9 to 16), the exact solution can be found only up to Problem 13. The exact method cannot solve them due to the magnification of the problems in the logical time. From problems 9 to 13, the gap obtained from the optimal answer for most meta-heuristics are very small and close to zero. After Problem 13, the proposed meta-heuristics are compared with each other using the RPD criteria to determine their performance. As can be seen in Tables 2 to 4, 20 medium- and large-sized problems are solved and compared. In these tables, among the five meta-heuristics according to the criteria examined, the proposed PCICA algorithm finds the best solutions for most problems among other algorithms. For example, in the case of large problems, the PCICA algorithm has an average RPD close to zero, which indicates the excellent performance of the proposed algorithm. It can be also seen in Figures 7 to 9 that the GA and ICA obtain larger RPD values than other algorithms. In Figures 7 to 9, it is clear that the proposed PCICA performs differently from other algorithms. Some algorithms (e.g., GA) have about 0.016 gaps while the PCICA has smaller gap values. In particular, the PCICA has the lowest average gap values of less than 0.003, which shows the outstanding performance of this algorithm among all algorithms. In the following, the percentage of the mean absolute deviation of meta-heuristic algorithms in all small, medium, large and very large problems is investigated. In this criterion, the PCICA method has a much better performance compared to other methods. The average value of this criterion for this algorithm is less than 15% while for other algorithms it is about 20%.

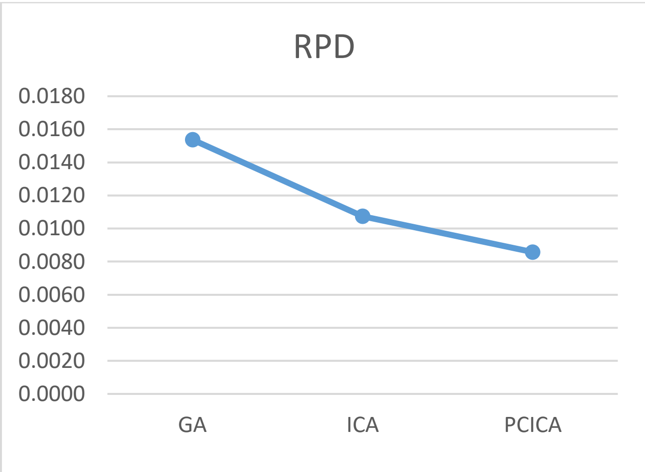


Figure 7. Average RPD of the meta-heuristic algorithms when β = -0.1 and α = 0.1

Table 2. Results obtained by meta-heuristic algorithms in multi-door mode β = -0.1, α = 0.1

Set	Lingo	GA		ICA		PCICA	
		Best	Average	Best	Average	Best	Average
1	7137.87	7137.87	8110.75	7182.05	9827.37	7137.87	8776.38
2	3461.14	3461.14	3912.01	3461.14	4183.71	3461.14	4248.34
3	8737.82	8737.82	13379.08	8737.82	10819.52	8737.82	11558.93
4	2642.56	2642.56	3886.06	2642.56	3299.72	2642.56	2740.87
5	7102.79	7102.79	11003.73	7102.79	9981.19	7102.79	9137.58
6	3428.07	3428.07	5246.49	3428.07	3990.78	3428.07	4309.56
7	6069.31	6069.31	9315.83	6069.31	7893.90	6069.31	6365.99
8	15282.85	15282.8	15989.26	15282.8	16347.45	15282.85	23531.69
9	3926.20	4029.62	5757.54	3998.52	4902.18	3926.20	4670.48
10	7056.30	7242.55	10914.38	7172.05	9587.23	7147.14	7910.50
11	9751.14	9848.59	11000.31	9848.59	11113.36	9828.23	11693.74
12	5186.40	5242.95	6778.57	5236.65	5980.92	5186.40	7446.37
13	10675.54	10725.51	16756.42	10686.7	12279.09	10685.13	12635.47
14	NA	16111.32	19417.41	16159.2	18400.78	16153.23	20346.00
15	NA	14094.93	17570.99	14113.1	17260.29	14081.38	20812.58
16	NA	24887.34	26584.51	24856.7	29720.56	24826.10	28680.30
17	NA	6554.97	7018.08	6537.70	9164.14	6529.79	7148.71
18	NA	12654.30	18597.70	12377.2	14688.97	12377.20	15301.31
19	NA	33315.30	43632.01	33063.8	52268.64	33047.88	42549.27
20	NA	23566.90	28098.43	23371.1	27165.85	23368.35	34750.15
21	NA	10814.25	13910.25	10670.3	11337.21	10663.50	11102.20
22	NA	17984.70	25467.77	17896.1	19665.61	17732.74	22517.24
23	NA	91945.90	96960.99	90858.8	135852.47	91012.86	94523.39
24	NA	18577.00	23727.78	18433.6	25127.57	18474.96	21514.14
25	NA	51769.20	58369.33	51265.2	78371.27	51087.12	56442.13
26	NA	13411.70	15462.49	13411.8	14939.10	13384.02	16864.85
27	NA	9377.41	11327.26	9366.14	10921.99	9276.27	10027.46
28	NA	5049.21	6226.66	5030.96	5803.88	5025.69	6166.15
29	NA	12144.40	13556.89	12180.5	16435.21	12176.03	13224.45
30	NA	8360.63	9286.27	8371.37	10254.61	8352.58	9547.08
31	NA	15055.00	21243.16	15036.5	17943.52	15036.54	15954.63
32	NA	22941.90	29328.67	22881.4	29135.22	22870.41	24865.31
33	NA	18094.30	27421.26	17698	25515.31	17696.00	21880.63
34	NA	6203.37	6301.49	6156.55	6291.14	6111.23	6265.52
35	NA	7641.22	7685.02	7577.71	7666.10	7558.49	7636.67

Table 3. Results obtained by the meta-heuristic algorithms in multi-door mode β = -0.5, α = 0.5

Set	Lingo	GA		ICA		PCICA	
		Best	Average	Best	Average	Best	Average
1	6619.86	6619.86	8110.75	6619.86	9114.18	6619.86	8139.46
2	3094.40	3094.40	3912.01	3094.40	3740.40	3094.40	3798.19
3	8497.40	8497.40	13379.08	8497.40	10521.82	8497.40	11240.89
4	2405.41	2405.41	3886.06	2405.41	3003.59	2405.41	2494.89
5	6100.73	6100.73	11003.73	6100.73	8573.04	6100.73	7848.45
6	3231.33	3251.98	5246.49	3231.33	3761.75	3231.33	4062.23
7	5882.16	6055.04	9315.83	5882.16	7650.49	5882.16	6169.70
8	15231.24	15328.60	15989.26	15374.3	16292.25	15231.24	23452.22
9	3687.62	3755.54	5757.54	4029.62	4604.29	3763.97	4386.67
10	6931.97	6956.30	10914.38	7242.55	9298.82	6932.14	7672.53
11	4740.72	4788.09	4861.31	4788.09	4856.99	4778.19	4872.83
12	10271.53	10371.00	10383.52	10383.5	10471.23	10271.53	10352.54
13	13224.02	13237.90	14781.29	13285.9	14831.41	13224.02	14896.38
14	NA	15230.70	15185.65	15185.6	15347.58	15225.17	15406.84
15	NA	11977.40	12078.18	11962.1	12081.94	11950.53	12031.59
16	NA	18849.10	19212.45	18872.3	19090.21	18825.89	19120.62
17	NA	10106.30	10229.19	10079.6	10328.57	10056.45	10288.00
18	NA	8150.60	8153.92	7972.11	8036.49	7942.11	8084.04
19	NA	17853.00	17853.03	17718.3	17786.78	17709.74	17816.29
20	NA	23867.10	24705.37	23668.7	24846.22	23665.96	24823.62
21	NA	16278.30	16450.31	16061.8	16352.63	16051.46	16422.83
22	NA	15229.81	15523.27	15154.7	15535.19	15016.37	15373.75
23	NA	74195.00	74366.52	73317.8	73590.28	73442.11	73795.23
24	NA	11967.90	12003.48	11875.5	11978.14	11902.17	12042.40
25	NA	28364.20	28510.38	28088.1	28292.61	27990.55	28181.06
26	NA	14175.20	14242.31	14275.2	14317.80	14145.90	14338.52
27	NA	8508.79	8584.05	8619.02	8633.41	8427.14	8537.31
28	NA	9358.93	9767.63	9392.87	9842.12	9349.12	9917.19
29	NA	12544.30	13193.98	12507.1	13921.56	12539.67	13967.71
30	NA	7641.72	7670.45	7631.91	7751.12	7624.56	7717.99
31	NA	4419.63	4460.47	4425.07	4487.37	4419.63	4508.95
32	NA	9222.90	9265.66	9247.26	9282.86	9218.46	9315.39
33	NA	17698.00	18227.86	18094.3	17904.09	17696.00	17960.65
34	NA	6156.55	6301.49	6203.37	6291.14	6111.23	6265.52
35	NA	7577.71	7685.02	7641.22	7666.10	7558.49	7636.67

Table 4. Results obtained by the meta-heuristic algorithms in multi-door mode β = -0.9, α = 0.9

Set	Lingo	GA		ICA		PCICA	
		Best	Average	Best	Average	Best	Average
1	6307.23	6307.23	7727.71	6307.23	8683.76	6307.23	7755.07
2	2889.22	2889.22	3652.62	2889.22	3492.39	2889.22	3546.34
3	8329.91	8329.91	13115.37	8329.91	10314.43	8329.91	11019.32
4	2273.06	2273.06	3672.24	2273.06	2838.32	2273.06	2357.61
5	5518.48	5518.48	9953.54	5518.48	7754.83	5518.48	7099.40
6	3108.88	3128.75	5047.68	3108.88	3619.20	3108.88	3908.29
7	5780.54	5950.43	9154.90	5834.82	7518.33	5780.54	6063.12
8	15197.23	15594.30	15953.56	15439.9	16255.87	15197.23	23399.86
9	3526.31	3591.27	5505.69	3853.36	4402.89	3599.32	4194.79
10	6644.55	7667.87	10461.84	6942.26	8913.27	6644.71	7354.41
11	4517.92	4580.07	4632.85	4585.07	4628.74	4553.64	4643.83
12	10162.14	10260.60	10272.94	10272.9	10359.72	10162.14	10242.29
13	11752.68	11765.00	13136.68	12807.7	13181.22	11752.68	13238.96
14	NA	14785.00	14741.24	14741.2	14898.43	14779.61	14955.96
15	NA	11881.00	11946.66	11831.8	11950.38	12120.40	12300.58
16	NA	18894.80	19055.13	18717.8	18933.89	18671.73	18964.04
17	NA	10139.10	10261.19	10412.6	10959.91	9989.60	10219.61
18	NA	6310.38	6312.96	6172.19	6222.04	6148.97	6258.85
19	NA	16503.80	16503.81	16379.2	16442.57	16371.35	16469.85
20	NA	22541.90	22816.06	21858.7	22946.14	21856.14	22925.27
21	NA	15118.70	15277.49	14919	15087.36	14809.50	15152.13
22	NA	14037.70	14308.21	14168.5	14319.20	13840.99	14170.39
23	NA	68320.70	68478.66	67512.9	67763.87	67627.44	67952.60
24	NA	11209.60	11742.32	11124.6	11519.01	10949.13	11078.12
25	NA	26803.50	26941.64	26842.6	26935.86	26450.42	26630.44
26	NA	13638.00	13702.60	13734.2	13775.23	13609.84	13795.16
27	NA	8361.80	8435.76	8470.13	8484.26	8281.56	8389.82
28	NA	9338.03	9745.82	9371.90	9820.14	9328.24	9895.05
29	NA	12496.40	13143.62	12459.4	13868.42	12491.81	13914.40
30	NA	7307.45	7334.93	7298.07	7412.07	7291.05	7380.38
31	NA	4336.38	4275.53	4241.59	4301.31	4236.38	4322.00
32	NA	8409.68	8448.68	8431.90	8464.36	8405.64	8494.02
33	NA	43026.40	44314.44	43989.7	43527.31	43021.42	43664.83
34	NA	6101.42	6245.07	6147.83	6234.82	6056.51	6209.42
35	NA	5866.84	5949.93	5916.01	5935.27	5851.96	5912.49

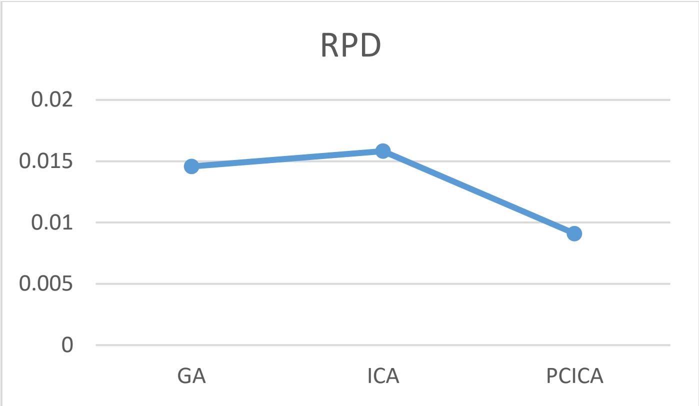


Figure 8. Average RPD of the meta-heuristic algorithms when β = -0.5 and α = 0.5

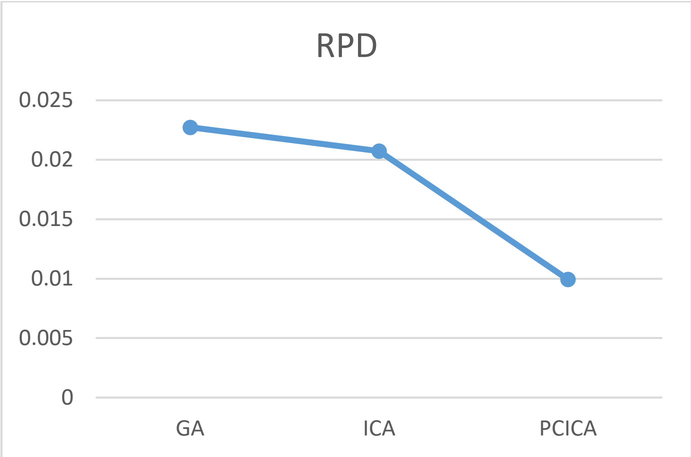


Figure 9. Average RPD of the meta-heuristic algorithms when β = -0.9, α = 0.9

Also, to evaluate the processing time, the average time spent to find the best solution in the implementation of the algorithms is calculated. The GA has the highest convergence speed (Table 5).

Table 5. Average computational time

Set	GA	ICA	PCICA
21	18.27	25.05	27.56
22	19.92	24.15	25.60
23	15.04	34.59	38.05
24	31.65	73.75	79.64
25	24.40	82.42	90.66
26	34.37	70.77	72.90
27	61.40	98.44	100.40
28	57.48	119.73	126.92
29	65.61	113.43	120.24
30	115.05	132.40	144.32
31	116.76	173.87	189.52
32	133.57	178.55	189.27
33	100.09	179.61	188.60
34	130.77	299.83	311.83
35	126.94	355.85	391.44

6. Conclusion

This study examined the issue of truck scheduling in a cross-dock system with a temporary warehouse close to the shipping door. In this paper, for the first time, the learning effect and deteriorating jobs were considered simultaneously in the mathematical model. By considering these factors, the mathematical model has become structurally closer to reality. Furthermore, two meta-heuristic algorithms (i.e., the genetic algorithm (GA) and the imperialist competitive algorithm (ICA)) were proposed to solve such a hard problem. The related results were compared with the branch-and-bound method reported by Lingo software. This paper also presented a new hybrid meta-heuristic algorithm, which resulted from the combination of the ICA and the principal component analysis (PCA) to solve this problem. The acquired results via Lingo software revealed that as the dimensions of the model grows the time to achieve the optimal answer increased exponentially. In the problems, which Lingo software has responded, the objective function values quality was the same as the proposed meta-heuristics. However, Lingo 8 could not solve the problem as the dimensions became bigger.

Finally, the numerical results obtained from all meta-heuristic algorithms have been investigated and analyzed for sensitivity. Metaheuristic algorithms were compared based on the best solution, average solution, RPD, and time criteria. In contrast, the proposed PCICA algorithm achieves better feasible solutions in an acceptable time. The effect of learning and deterioration of jobs was also considerable. An increase in the learning effect rate and decrease in deterioration rate improved the objective function value. The reason should be sought like the presented model. The learning effect and deteriorating jobs can have a diminishing effect on completion periods.

References

Abdi, A., Abdi, A., Akbarpour, N., Amiri, A. S., and Hajiaghaei-Keshteli, M., (2020). "Innovative approaches to design and address green supply chain network with simultaneous pick-up and split delivery", *Journal of Cleaner Production*, Vol. 250, No.1, pp.119437.

Alpan, G., Ladier, A.-L., Larbi, R., and Penz, B., (2011). "Heuristic solutions for transshipment problems in a multiple door cross-docking warehouse", *Computers and Industrial Engineering*, Vol.61, No. 2, pp.402-408 .

Amini, A., and Tavakkoli-Moghaddam, R., (2016). "A bi-objective truck scheduling problem in a cross-docking center with probability of breakdown for trucks", *Computers and Industrial Engineering*, Vol. 96, No. 1, pp.180-191 .

Amini, A., Tavakkoli-Moghaddam, R., and Omidvar, A., (2014). "Cross-docking truck scheduling with the arrival times for inbound trucks and the learning effect for unloading/loading processes", Vol. 2, No. 1, pp. 784-804.

Amiri, S. A. H. S., Zahedi, A., Kazemi, M., Soroor, J., and Hajiaghaei-Keshteli, M., (2020). "Determination of the optimal sales level of perishable goods in a two-echelon supply chain network", *Computers and Industrial Engineering*, Vol. 139, No. 1, pp. 106-156.

Afshar-Bakeshloo, M., Jolai, F., Mazinani, M., and Tavakkoli-Moghaddam, R., (2019). "A satisfactory multi-agent single-machine considering a cross-docking terminal", *International Journal of System of Systems Engineering*, Vol. 9, No.4, pp. 307-330.

Atashpaz-Gargari, E., and Lucas, C., (2007). "Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition", IEEE congress on evolutionary computation, 4661-4667, Singapore.

Babaee Tirkolaee, E., Hadian, S., and Golpira, H., (2019). "A novel multi-objective model for two-echelon green routing problem of perishable products with intermediate depots", *Journal of Industrial Engineering and Management Studies*, Vol. 6, No. 2, pp. 101-110.

Babaee Tirkolaee, E., Goli, A., Pahlevan, M., and Malekalipour Kordestanizadeh, R., (2019). "A robust bi-objective multi-trip periodic capacitated arc routing problem for urban waste collection using a multi-objective invasive weed optimization", *Waste Management and Research*, Vol. 37, No. 11, pp. 1089-1101.

Babaee Tirkolaee, E., Alinaghian, M., Bakhshi Sasi, M. and Seyyed Esfahani, M.M., (2016). "Solving a robust capacitated arc routing problem using a hybrid simulated annealing algorithm: a waste collection application", *Journal of Industrial Engineering and Management Studies*, Vol. 3, No. 1, pp. 61-76.

Baniamerian, A., Bashiri, M., and Tavakkoli-Moghaddam, R., (2019). "Modified variable neighborhood search and genetic algorithm for profitable heterogeneous vehicle routing problem with cross-docking", *Applied Soft Computing*, Vol. 75, No.1, pp. 441-460.

Bartholdi III, J. J., and Gue, K. R., (2000). "Reducing labor costs in an LTL crossdocking terminal", *Operations Research*, Vol. 48, No.6, pp. 823-832.

Bellanger, A., Hanafi, S., and Wilbaut, C., (2013). "Three-stage hybrid-flowshop model for cross-docking", *Computers and Operations Research*, Vol. 40, No. 4, pp. 1109-1121 .

Boysen, N., and Fliedner, M., (2010). "Cross-dock scheduling: Classification, literature review and research agenda", *Omega*, Vol. 38, No. 6, pp. 413-422.

Davoodi, S. M. R., and Goli, A., (2019). "An integrated disaster relief model based on covering tour using hybrid Benders decomposition and variable neighborhood search: Application in the Iranian context", *Computers and Industrial Engineering*, Vol. 130, No. 1, pp. 370-380.

Dulebenets, M. A. (2019). "A Delayed Start Parallel Evolutionary Algorithm for just-in-time truck scheduling at a cross-docking facility", *International Journal of Production Economics*, Vol. 212, No. 1, pp. 236-258.

Fard, A. F., Gholian-Jouybari, F., Paydar, M. M., and Hajiaghaei-Keshteli, M., (2017). "A bi-objective stochastic closed-loop supply chain network design problem considering downside risk", *Industrial Engineering and Management Systems*, Vol. 16, No. 3, pp. 342-362.

Fathollahi-Fard, A. M., Ranjbar-Bourani, M., Cheikhrouhou, N., and Hajiaghaei-Keshteli, M., (2019). "Novel modifications of social engineering optimizer to solve a truck scheduling problem in a cross-docking system", *Computers and Industrial Engineering*, Vol. 137, No.1, pp. 106103.

Fonseca, G. B., Nogueira, T. H., and Ravetti, M. G. (2019). "A hybrid Lagrangian metaheuristic for the cross-docking flow shop scheduling problem", *European Journal of Operational Research*, Vol. 275, No.1, pp. 139-154.

Gholian-Jouybari, F., Afshari, A. J., and Paydar, M. M., (2018). "Utilizing new approaches to address the fuzzy fixed charge transportation problem", *Journal of Industrial and Production Engineering*, Vol. 35, No.3, pp. 148-159.

Goli, A., Babaee Tirkolaee, E., and Soltani, M., (2019). "A robust just-in-time flow shop scheduling problem with outsourcing option on subcontractors", *Production and Manufacturing Research*, Vol. 7, No.1, pp. 294-315.

Goli, A., and Davoodi, S. M. R., (2018). "Coordination policy for production and delivery scheduling in the closed loop supply chain", *Production Engineering*, Vol. 12, No.5, pp. 621-631.

Goli, A., Zare, H. K., Tavakkoli-Moghaddam, R., and Sadeghieh, A., (2019). "Hybrid artificial intelligence and robust optimization for a multi-objective product portfolio problem Case study: The dairy products industry", *Computers and Industrial Engineering*, Vol. 137, No.1, pp. 106090.

Goli, A., Zare, H. K., Tavakkoli-Moghaddam, R., and Sadegheih, A., (2020). "Multiobjective fuzzy mathematical model for a financially constrained closed-loop supply chain with labor employment", *Computational Intelligence*, Vol. 36, No.1, pp. 4-34.

Golmohamadi, S., Tavakkoli-Moghaddam, R., and Hajiaghaei-Keshteli, M., (2017). "Solving a fuzzy fixed charge solid transportation problem using batch transferring by new approaches in meta-heuristic", *Electronic Notes in Discrete Mathematics*, Vol. 58, No.1, pp. 143-150.

Golshahi-Roudbaneh, A., Hajiaghaei-Keshteli, M., and Paydar, M. M., (2017). "Developing a lower bound and strong heuristics for a truck scheduling problem in a cross-docking center", *Knowledge-Based Systems*, Vol. 129, No.1, pp. 17-38.

Golshahi-Roudbaneh, A., Hajiaghaei-Keshteli, M., and Paydar, M. M., (2019). "A hybrid approach in metaheuristics for a cross-dock scheduling considering time windows and deadline for trucks departure", *Scientia Iranica*, Article in Press.

Gupta J.N.D., Gupta S.K., (1988). "Single facility scheduling with nonlinear processing times", *Computers and Industrial Engineering*, Vol. 14, No. 4, pp. 387-393.

Hajiaghaei-Keshteli, M., Sajadifar, S. M., and Haji, R., (2011). "Determination of the economical policy of a three-echelon inventory system with (R, Q) ordering policy and information sharing", *The International Journal of Advanced Manufacturing Technology*, Vol. 55, No.5, pp. 831-841.

Hajiaghaei-Keshteli, M., and Sajadifar, S. M., (2010). "Deriving the cost function for a class of three-echelon inventory system with N-retailers and one-for-one ordering policy", *The International Journal of Advanced Manufacturing Technology*, Vol. 50, No. 1, pp. 343-351.

Hasani Goodarzi, A., Tavakkoli-Moghaddam, R., and Amini, A., (2020). "A new bi-objective vehicle routing-scheduling problem with cross-docking: Mathematical model and algorithms", *Computers and Industrial Engineering*, Vol. 149, No.1, pp. 106832.

Heidari, F., Zegordi, S. H., and Tavakkoli-Moghaddam, R., (2018). "Modeling truck scheduling problem at a cross-dock facility through a bi-objective bi-level optimization approach", *Journal of Intelligent Manufacturing*, Vol. 29, No. 5, pp. 1155-1170.

Holland, J. H., (1992). "Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence", MIT press, USA.

Hosseinian, A. H., and Bardaran, V., (2020). "Modified Pareto archived evolution strategy for the multi-skill project scheduling problem with generalized precedence relations", Journal of Industrial Engineering and Management Studies, Vol. 7, No.1, pp 59-86.

Huang, X., Wang, M. Z., and Ji, P., (2014). "Parallel machines scheduling with deteriorating and learning effects", *Optimization Letters*, Vol. 8, No.2, pp. 1-8.

Huixuan, G., (2005). Applied multivariate statistical analysis. Beijing University Press.

Ladier, A. L., and Alpan, G., (2016). "Cross-docking operations: Current research versus industry practice", *Omega*, Vol. 62, No. 1, pp. 145-162.

Maknoon, M. Y., and Baptiste, P., (2010). "Moving freight inside cross-docking terminals", 8th International Conference on Supply Chain Management and Information. IEEE, 1-6, Hong Kong.

Mir, M. S. S., and Rezaeian, J., (2016). "A robust hybrid approach based on particle swarm optimization and genetic algorithm to minimize the total machine load on unrelated parallel machines", *Applied Soft Computing*, Vol. 41, No. 1, pp. 488-504.

Mohtashami, A., Tavana, M., Santos-Arteaga, F. J., and Fallahian-Najafabadi, A., (2015). "A novel multi-objective meta-heuristic model for solving cross-docking scheduling problems", Applied Soft Computing, Vol. 31, No.1, pp. 30-47 .

Mousavi, S. M., Tavakkoli-Moghaddam, R., and Jolai, F., (2013). "A possibilistic programming approach for the location problem of multiple cross-docks and vehicle routing scheduling under uncertainty", *Engineering Optimization*, Vol. 45, No. 10, pp. 1223-1249.

Molavi, D., Shahmardan, A., and Sajadieh, M. S., (2018). "Truck scheduling in a cross-docking systems with fixed due dates and shipment sorting", *Computers and Industrial Engineering*, Vol. 117, No. 1, pp. 29-40.

Pan, F., Zhou, W., Fan, T., Li, S., and Zhang, C. (2020). "Deterioration rate variation risk for sustainable cross-docking service operations", *International Journal of Production Economics*, 107932.

Rijal, A., Bijvank, M., and de Koster, R. (2019). "Integrated scheduling and assignment of trucks at unit-load cross-dock terminals with mixed service mode dock doors", *European Journal of Operational Research*, Vol. 278. No. 3, pp. 752-771.

Rostami, M., Pilerood, A. E., and Mazdeh, M. M. (2015). "Multi-objective parallel machine scheduling problem with job deterioration and learning effect under fuzzy environment", *Computers and Industrial Engineering,* Vol. 85, No.1, pp. 206-215.

Rohrer, M., (1995). "Simulation and cross-docking", *The Simulation Conference Proceedings*, IEEE, 846-849, USA.

Sangaiah, A. K., Tirkolaee, E. B., Goli, A., and Dehnavi-Arani, S., (2020). "Robust optimization and mixed-integer linear programming model for LNG supply chain planning problem", *Soft Computing*, Vol. 24, No.11, pp. 7885-7905.

Seyedi, I., Hamedi, M., and Tavakkoli-Moghaddam, R., (2019). "Truck scheduling in a cross-docking terminal by using novel robust heuristics", *International Journal of Engineering*, Vol. 32, No. 2, pp 296-305.

Seyedi, I., Mirzazadeh, S., Maleki-Daronkolaei, A., Mukhtar, M., and Sahran, S., (2016). "An inventory model with reworking and setup time to consider effect of inflation and time value of money", *Journal of engineering science and Technology,* Vol. 11, No. 3, pp. 416-430.

Seyedi, I., Maleki-Daronkolaei, A., and Kalashi, F., (2012). "Tabu search and simulated annealing for new three-stage assembly flow shop scheduling with blocking", *Interdisciplinary Journal of Contemporary Research in Business*, Vol. 4, No. 8, pp. 394-402.

Seyedi, I., and Maleki-Daronkolaei, A., (2013). "Solving a two-stage assembly flowshop scheduling problem to minimize the mean tardiness and earliness penalties by three meta-heuristics", *Caspian Journal of Applied Sciences Research*, Vol. 2, No. 4, pp. 67-78.

Song, K., and Chen, F., (2007). "Scheduling cross-docking logistics optimization problem with multiple inbound vehicles and one outbound vehicle", *International Conference On Automation And Logistics*, IEEE, pp. 3089-3094.

Vahdani, B., and Zandieh, M., (2010). "Scheduling trucks in cross-docking systems: Robust meta-heuristics", *Computers and Industrial Engineering,* Vol. 58, No. 1, pp. 12-24

Wisittipanich, W., and Hengmeechai, P., (2017). "Truck scheduling in multi-door cross-docking terminal by modified particle swarm optimization", *Computers and Industrial Engineering*, Vol. 113, No. 1, pp. 793-802.

Xu, J., Xu, X., and Xie, S. Q. (2011). "Recent developments in Dual Resource Constrained (DRC) system research", *European Journal of Operational Research,* Vol. 215, No. 2, pp. 309-318.

Yu, W. (2002). *Operational strategies for cross-docking systems*, Iowa State University, USA.

Yu, W., and Egbelu, P. J. (2008). "Scheduling of inbound and outbound trucks in cross-docking systems with temporary storage", *European Journal of Operational Research*, Vol. 184, No. 1, pp 377-396.

Zhao, Q. H., and Cheng, T. E. (2009). "An analytical study of the modification ability of distribution centers", *European Journal of Operational Research*, Vol. 194, No. 3, pp. 901-910.

This article can be cited: Seyedi, I., Hamedi, M., Tavakkoli-Moghaddam, R., (2021). "Developing a mathematical model for a multi-door cross-dock scheduling problem with human factors: A modified imperialist competitive algorithm", *Journal of Industrial Engineering and Management Studies*, Vol. 8, No. 1, pp. 180-201.