E-ISSN: 2676-6167 | P-ISSN: 2538-5100

Journal of Applied Research on Industrial Engineering

JARIE

www.journal-aprie.com

J. Appl. Res. Ind. Eng. Vol. 10, No. 1 (2023) 56–83.

Paper Type: Research Paper

Integration of Two-Stage Assembly Flow Shop Scheduling and Vehicle Routing Using Improved Whale Optimization Algorithm

Vahid Bahmani[1], Mohammad Amin Adibi[1,*], Esmaeil Mehdizadeh[1]

[1] Department of Industrial Engineering, Faculty of Industrial and Mechanical Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran; v.bahmani@ymail.com; maadibi@aut.ac.ir; emqiau@yahoo.com.

Citation:

Bahmani, V., Adibi, M. A., & Mehdizadeh, E. (2023). Integration of two-stage assembly flow shop scheduling and vehicle routing using improved whale optimization algorithm. *Journal of applied research on industrial engineering, 10*(1), 56-83.

Received: 16/02/2022 Reviewed: 20/03/2022 Revised: 09/05/2022 Accepted: 01/06/2022

Abstract

This paper provides an integrated model for a two-stage assembly flow shop scheduling problem and distribution through vehicle routing in a soft time window. So, a Mixed-Integer Linear Programming (MILP) model has been proposed with the objective of minimizing the total cost of distribution, holding of products, and penalties of violating delivery time windows. To solve this problem, an improved meta-heuristic algorithm based on Whale Optimization Algorithm (WOA) has been developed. The main innovations in the study include considering soft time window, sequence-dependent setup time, delivery time window, heterogeneous vehicles, holding costs of final products, and unrelated assembly machines. A comparison of the integrated and non-integrated model in a case study of industrial gearboxes production shows that the integrated model compared to the non-integrated model has saved 15.6% and 13.6% in terms of delay time and total costs, respectively. Computational experiments also indicate the efficiency of improved WOA in converging to optimal solution and achieve better solution in comparison to the Genetic Algorithm (GA). The results show that increasing the setup time can lead to an increase in total costs. It can be said that the increase of setup time increases the completion of time jobs. Also, the costs increased with decreasing the transport fleet capacity to -20%. The reason for this is that by reducing the capacity of vehicles, the model has to use more vehicles.

Keywords: Sentence Two-stage assembly, Vehicle routing, Whale optimization algorithm, Genetic algorithm.

1 | Introduction

In today's competitive environment, manufacturing companies are seeking to minimize costs to succeed in supply chains. Integrating supply chain decisions is one of the cost-saving methods. Efficient supply chain design involves complex issues such as inventory management, production scheduling, and order distribution [1]. On the other hand, flexibility and timely delivery are essential to customer satisfaction [2]. In the non-integrated approach, the manufacturing section prefers fewer preparations to reduce production costs, and more transportation travels to reduce delivery time, while the distribution section prefers less transportation travels with larger volumes to reduce distribution costs. In addition to the conflict between these two, this approach causes an increase in the costs of both sections and the entire system. On the other hand, integrating production and distribution schedules can reduce system costs. Production and distribution are complex problems,

Corresponding Author: maadibi@aut.ac.ir

https://doi.org/10.22105/jarie.2022.329251.1450

and therefore their integration will increase the complexity of the problem. However, it is expected that total costs be reduced by integrating the production and distribution processes [3]-[7].

The objective of this study is to determine order production and distribution schedule and routes that minimize total company costs including fixed costs of using vehicles, variable travel costs, earliness and tardiness penalties, as well as holding costs. In this study, after modeling the integrated production and distribution problem, it will be compared with the non-integrated model. The Whale Optimization Algorithm (WOA) will be also used as a new and efficient algorithm for production and distribution. The other sections of this study are as follows:

Section 2 will present relevant literature. Section 3 will describe the problem and define the mathematical model. In Section 4, the used meta-heuristic algorithms will be introduced and developed. Section 5 of the study contains some experiments and a case study to evaluate the performance of the proposed model and comparison of the performance of the proposed algorithms. Finally, the conclusion and suggestions for the future will be presented in the last section.

2 | Literature Review

The two-stage assembly flow shop is one of the most widely used scheduling problems in manufacturing industries. Potts et al. [8] well demonstrated the utility of this problem. By studying the two-stage assembly scheduling problem with the makespan objective function, they proved that the problem is NP-hard in the first stage due to the existence of two machines. Lee et al. [9] used this problem to assemble a fire engine. They used a branch and bound algorithm for the two-stage assembly scheduling problem with the makespan objective function and analyzed their boundary error by presenting three heuristic approaches. In another study, Allahverdi and Al-Anzi [10] investigated queries scheduling in a distributed database system. In this regard, many real-world problems can be modeled using the two-stage assembly scheduling problem. Therefore, the two-stage assembly problem can be considered as a general case of the flow shop problem. Tozkapan et al. [11] described this problem with the performance measure of total weighted flow time. With creating a lower bound, they obtained good results for problems of logical scale and proposed a heuristic algorithm for large scale problems. Allahverdi and Al-Anzi [10] studied the problem presented by Tozkapan et al. [11] using total completion time. They developed three algorithms, including Tabu Search (TS), Hybrid Tabu Search (HTS), and Simulated Annealing (SA). Computational experiments showed that although the CPU time of all three algorithms was approximately the same, and HTS improved the error rate by 60 and 90 percent compared to TS and SA, respectively. Sung and Kim [13] addressed the problem of scheduling a two-stage multi-machine assembly flow shop. They proposed a branch and bound algorithm and an efficient heuristic algorithm to minimize the sum of completion times. Allahverdi and Al-Anzi [10] also considered the problem with bi-criteria of makespan and maximum tardiness, and proposed three heuristic algorithms, including Particle Swarm Optimization (PSO), TS and Self-adaptive Differential Evolution (SDE). The analyses showed that both SDE and PSO had better performance than TS, but PSO performance was better than SDE. Koulamas and Kyparisis [14] generalized the two-stage problem to the three-stage assembly scheduling problem regarding collection and transportation with the goal of minimizing the makespan. They also proposed several heuristic algorithms and evaluated the worst-case ratio bound of those algorithms.

Zhang and Tang [15] integrated Preventive Maintenance (PM) with the problem of two-step assembly scheduling by presenting a Mixed Integer Linear Programming (MILP) model. To solve the problem, they proposed an iterative greedy algorithm based on PM and two heuristics Mixed Constrained Machine Preventive Maintenance (MCMTPM) and Net Economic Hybrid Preventive Maintenance (NEHPM). Numerical results showed that the proposed Iterated Greedy Preventive Maintenance (IGPM) embedded with NEHPM and local reference search performed better than the other 9 intelligent methods. Pourhejazy et al. [16] addressed the issue of distributed assembly timing with distributed start-up time sequences (DTSAFSP-SDSTs) with the aim of minimizing makespan. They used the Iterated Greedy algorithm to solve this problem effectively. Numerical experiments showed that the Improved Iterated Greedy algorithm offers the best solutions in most cases.

Hatami et al. [17] generalized the model proposed by Koulamas and Kyparisis [14], by considering sequence-dependent set-up times. Their mathematical model aimed to minimize the weighted sum of the mean flow time and maximum tardiness. They proposed two algorithms, SA and TS, to solve the problem and compared their performance. Andrés and Hatami [18] formulated two mathematical models to solve the three-stage assembly flow shop problem. Their objective was to minimize the total completion time by considering the sequence-dependent setup time in the first and third stages. Their results showed that they could find optimal solutions for problems with n = 15 (number of jobs) and m = 4 (number of parts). Maleki-Darounkolaei et al. [19] investigated this problem by considering the sequence-dependent setup time in the first stage as well as the blocking times between successive stages. They proposed a SA algorithm to solve the problem with the objective function of minimizing the weighted sum of the two objectives of the makespan and the mean completion time. Dalfard et al. [20] studied the problem by considering the sequence-dependent setup time and transportation times. Their objective function included minimizing the sum of total weighted squared earliness, total weighted squared tardiness, number of tardy jobs and makespan. For solving the problem, they used a hybrid Genetic Algorithm (GA) and concluded that for jobs more than 10, the results were not comparable between Lingo 8 and the hybrid GA.

Mozdgir et al. [21] considered the problem of two-stage assembly scheduling with non-identical assembly machines with the objective function of minimizing the weighted sum of the two criteria of mean completion time and makespan. They proposed a hybrid variable neighborhood search heuristic to solve the MILP model. Tian et al. [22] considered the problem with two criteria of mean completion time and makespan, and proposed a Discrete Particle Swarm Optimization (DPSO) algorithm to solve the problem. The results of that study indicated the efficiency of DPSO. Allahverdi et al. [23] studied the problem by assuming setup times as zero and the objective function of minimizing total tardiness. They proposed two versions of the SA algorithm, two versions of cloud theory-based SA, an insertion algorithm, and a GA to solve the problem. The results showed that one version of the SA combined with PIA had a better performance than the other algorithms. Allahverdi and Aydilek [12] proposed two new algorithms for a two-stage assembly scheduling problem considering separate setup times and compared four existing algorithms. The results of the analysis indicated that the error of the best algorithm is less than other algorithms by 54%-98%. Navaei et al. [24], also addressed the problemby considering several non-identical assembly machines and the makespan objective function. They developed a MILP model and proposed a hybrid SA algorithm to solve the problem.

Shoaardebili and Fattahi [25] studied the three-stage assembly flow shop scheduling problem simultaneously with machine availability constraints and two objective functions of minimizing the sum of weighted tardiness and earliness and minimizing total weighted completion times. Analyses indicated that of the two NSGA II and MOSA algorithms proposed, NSGA II had a better performance. Komaki and Kayvanfar [26] studied the two-stage assembly scheduling problem with identical assembly machines and the release date of jobs. The objective function of their model was makespan. They proposed a Grey Wolf Optimizer (GWO) algorithm to solve the problem. The analyses showed that the GWO algorithm exhibited better performance than the other more well-known algorithms. The two-stage assembly scheduling problem has developed in many ways over time. Allahverdi and Al-Anzi [10] considered the problem with m machines in the first stage and the objective function of the total completion time. They developed three heuristic approaches, consisting of an HTS, an SDE, and Novel Self-adaptive Differential Evolution (NSDE) algorithm to solve the problem. Computational experiments showed that NSDE had better performance than the other two algorithms.

Wang et al. [27] considered the two-stage assembly flow shop scheduling problem with batch delivery to one customer. For solving the problem with the objective function of minimizing the weighted sum of average arrival time at the customer and the total delivery cost, they proposed two heuristic methods based on SPT and LPT and a new hybrid meta-heuristic (HGA-OVNS). The computational results showed the superiority of the HGA-OVNS meta-heuristic algorithm. Kazemi et al. [28] studied the problem of two-stage assembly flow shop scheduling with identical assembly machines and batch delivery. Their objective was to schedule jobs considering batches that would minimize the sum of tardiness and delivery costs. They proposed the Imperialist Competition Algorithm (ICA) and Hybrid Imperialist Competition Algorithm (HICA) to solve the MILP model. Their computational results indicated the superiority of the HICA algorithm in terms of the objective function but the ICA algorithm needed a comparably less time for implementation. Jung et al. [29] considered the two-stage assembly flow shop scheduling problem to assemble products with dynamic components-sizes and makespan objective function. In their MILP model, they considered the setup time to process the components of a new product. They proposed three GAs with different chromosome representations to solve large-scale problems. Wu et al. [30] addressed the two-stage flow shop scheduling problem using three machines and with learning phenomenon. Their objective was to minimize the total completion time and used the branch and bound algorithm to solve small-scale problems. In addition, six versions of the hybrid PSO algorithm were proposed for small- and large-scale problems and three different data types. In addition, ANOVA was used to evaluate the performance of the proposed algorithms. Goli and Davoodi [31] presented a coordinated model of production and distribution with a constant rate of demand in the supply chain. They developed two algorithms including SA refrigeration simulation and Biography-Based Optimization (BBO) algorithm. Numerical results showed better performance of BBO algorithm.

Basir et al. [32] considered the problem of two-stage assembly with batch delivery. They presented a Mixed-Integer Linear Programming (MILP) model with the aim of minimizing the weighted number of tardy jobs and the sum of delivery costs. They proposed a two-stage Improved Genetic Algorithm (IGA) with a hierarchical decision-making approach. Yavari and Isvandi [33] integrated the two-stage assembly scheduling problem by ordering the parts needed to process the components. They developed a MILP model to minimize the sum of the total weighted completion time, parts ordering, and holding cost. They proposed a GA and the computational results showed that the integrated approach improved the supply chain performance up to 8.16%. Luo et al. [34] investigated a two-stage assembly scheduling problem to minimize makespan considering separate setup times. They proposed a hybrid branch and bound algorithm. Their computational results showed that the algorithm performed better than one of the available methods. Talens et al. [35] addressed the problem of two-stage assembly scheduling with identical assembly machines and the objective function of minimizing total completion time. They presented two heuristic algorithms of CH_{MMA} and $BSCH_{MMA}$. The computational results showed that the proposed heuristic methods had better performance than the available heuristic ones. Lin and Chen [36], provided a dynamic scheduling of two-stage assembly flow shop to minimize the total tardiness as a Markov decision process. A Proximal Policy Optimization (PPO) algorithm was developed to efficiently train agent using production data. Hosseini et al. [37] investigated a two-stage production system consists of a fabrication stage followed by an assembly stage. Moreover, a heuristic algorithm and two proper lower bounds were introduced as references to evaluate the performance of the proposed heuristic algorithm.

In a study, Masruroh et al. [38] addressed the issue of multi-product production planning and integrated distribution in the product supply chain network. Their proposed model has two stages, the first of which is determined in order to maximize the profit of production, delivery and inventory planning for each product. Then, in the second stage, the exact production schedule is optimized to minimize the total start-up costs. Numerical results showed a significant reduction in costs and an increase in annual profits. Based on literature review section the literature review table is as follows.

Table 1. Summary of literature review.

No	Author	Soft Time Window	Sequence-Dependent Setup Time	Delivery Time Window	Heterogeneous Vehicles	Holding Costs of Final Products	Unrelated Assembly Machines	Integration of Flow Shop Scheduling And Vehicle Routing	Metaheuristic/Heuristic
1	Zhang and Tang [15]			*			*		*
2	Pourhejazy et al. [16]		*				*		*
3	Lin and Chen [36]	*		*					*
4	Masruroh et al. [38]	*							-
5	Hosseini et al. [37]				*				*
6	Talens et al. [35]	*					*		*
7	Luo et al. [34]					*	*		*
8	Yavari and Isvandi [33]			*		*			
9	Basir et al. [32]			*					*
10	Goli and Davoodi [31]			*					*
11	Wu et al. [30]	*							*
12	Kazemi et al. [28]	*					*		*
13	Wang et al. [27]				*		*		*
14	Allahverdi et al.[23]		*				*		*
15	Allahverdi and Al-Anzi [10]	*					*		*
16	Komaki and Kayvanfar [26]			*					*
17	Shoaardebili and Fattahi [25]	*					*		*
18	Mozdgir et al. [21]	*					*		*
19	Tian et al. [22]	*					*		*
20	Navaei et al. [24]				*				*
21	Dalfard et al. [20]	*		*					*
22	Hatami et al. [17]	*	*						*
23	This Study	*	*	*	*	*	*		*

In none of the previous studies, the two-stage assembly flow shop problem has been integrated with the distribution of orders through vehicle routing. However, distribution stage decisions are highly dependent on production stage decisions, and their integration can lead to minimizing total system costs. Therefore, the main innovation of the current study is to provide a new mathematical model and solution for the problem of two-stage assembly flow shop and vehicle routing. Other innovations in the study include considering soft time window, sequence-dependent setup time, delivery time window, heterogeneous vehicles, holding costs of final products, and unrelated assembly machines. Since production, assembly, and vehicle routing decisions are all short-term and operational, it is not advisable to use exact solutions that need a long time to perform. Therefore, an improved and novel meta-heuristic methods will be used in this study. For this purpose, the Improved Whale Optimization Algorithm (IWOA) is proposed and applied for the first time to integrate a two-stage assembly flow shop and vehicle routing.

3 | Problem Description and Proposed Mathematical Model

In this section, the structure of the problem is first described in detail. In the next step, after introducing the symbols, integrated and non-integrated mathematical models are presented.

3.1 | Problem Description

The problem under study involves decision making in three related areas including production, assembly, and distribution. In the different production systems, these three areas are performed sequentially and successively and independent planning and optimization are performed for each area. Integrating production, assembly and distribution decisions is a novel approach that has been addressed in this study and has been introduced as the Integration of Production, Assembly and Distribution (IPAD) problem. First, the IPAD model is introduced, and then the non-integrated model is presented. In the integrated model, it is assumed that at moment 0, J jobs are present in the production system. Each job represents an order. Each order has an independent I components, each of which requires independent processing. Each process is performed on a specific machine. It is important to note that there is a sequence-dependent setup time for each job to be processed. Therefore, determining the sequence of jobs has a substantial effect on the completion time of different jobs.

Once all the components of each job are processed, they should be assembled on one of the l unrelated machines in the second stage. Since vehicles are available at the completion time of the last job (Cmax), each product is held in a temporary warehouse after completion of assembly. Therefore, the assembled products must be held in the warehouse until the start of loading and distribution. This holding is costly for the company and it is necessary to plan the production and assembly in such a way that the holding time and costs of the orders are minimized.

In the distribution stage, there are V vehicles with limited capacity in the system. Each vehicle has a fixed cost as well as a variable cost per 1 km distance traveled. At this stage, time is a crucial pillar. Each customer has a time window for delivering orders. Servicing outside this time window will impose stupendous costs on the company. Therefore, at this stage, attempts will be made to determine an optimal route for each vehicle that will lead to the least cost on the route and the least penalties of violating the time window. *Fig. 1* illustrates the structure of this problem schematically.

Other assumptions are as follows:

I. All orders are available in the system at time zero.
II. All production and assembly machines are available since time zero.
III. Preemption is not allowed on any machine.
IV. Machine idle time in the first stage is not allowed.
V. Processing time and machine setup time for all jobs are predetermined.
VI. The assembly time of each job is definite and predetermined.
VII. In the distribution stage, the delivery of orders to customers has a specific time.
VIII. Travel time is the same for all vehicles.
IX. Each customer has a soft time window whose violation imposes costs on the company.

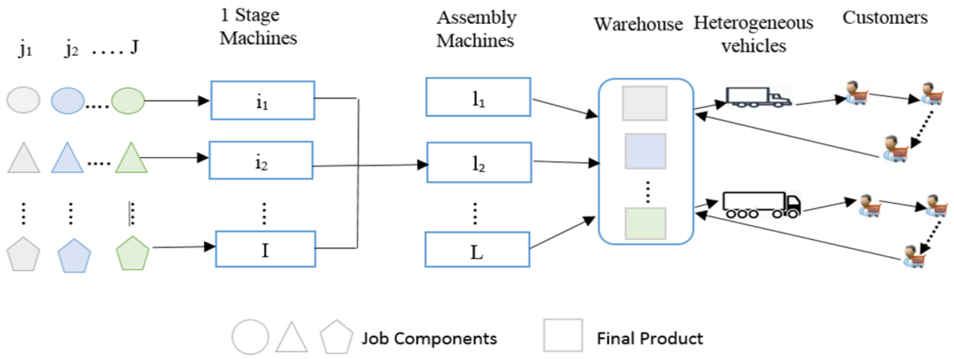

Fig. 1. Schematic representation of the integration of two-stage assembly flow shop and vehicle routing.

In this section, based on the considered assumptions, a distribution-production integrated mathematical model is presented.

3.2 | Notations

We used the following notations to formulate the problem.

Table 2. Model notations.

Notation	Description
Indices	
$i,\ i'$	Index of manufacturing machine or the component related to each order in the first stage, $i = 1,2,\ldots, I.$
j, j'	Index of job(order) or manufactured product, $j = 1,2,\ldots, J$
r	Index of position (rank) of orders in the sequence
l	Index of non-identical assembly machines in the second stage $l = 1,2,\ldots, L.$
v	Index of vehicles
e, e', k	Index of nodes, 0 for origin node (production location-manufacturer) and one node for each customer, $e = \{0\} \cup \{1,\ldots, K\} = 0,1,2,\ldots,K.$
Parameters	
p_{ij}	Processing time of component i of order (job) j in the first stage.
$ST_{ij'ir}$	Setup time of machine i for job j after job j' at position r.
q_{jl}	Assembly time of order j on assembly machine l in the second stage.
ho_j	Cost of holding order j in warehouse after assembly.
av_{ev}	It is 1 if the node e can be serviced by the device v, otherwise 0.
Ca_v	Capacity of vehicle v.
pn_{jk}	Demand of customer k for order j.
$CO_{ee'v}$	Travel cost between node e and e' with vehicle v.
$ti_{vee'}$	Travel time between node e and e' with vehicle v.
S_k	Service time for customer k.
$[ed_k, ld_k]$	Service time window of customer k.
ep_k	Order delivery earliness penalty of customer k.
lp_k	Order delivery tardiness penalty of customer k.
Notation	**Description**
fc_v	Cost of using vehicle v
M	A large number
Decision Variables	
X_{ijr}	It is 1 if processing of job j starts on machine i at position r of the first stage, otherwise 0.
Z_{jrl}	It is 1 if the order j in the at position r of the first stage is assembled on the assembly machine l in the second stage, otherwise 0.
$W_{vee'}$	It is 1 if vehicle v travels to arc $(e,\ e')$, otherwise 0.
CT_{jr}	Completion time of job j on production machines at position r.
C_{ijr}	Completion time of the processing of the machine i of job j in the first stage on the production machines at position r.
$Start_{ijr}$	Start time of the processing of the component i of order j in the first stage on the production machines at position r.
C_{max}	Completion time of the assembly of last job in the second stage.
G_{rl}	Completion time of the assembly of order at position r on the assembly machine l in the second stage.
ET_{vk}	Earliness of vehicle v while arriving at customer k.
LT_{vk}	Tardiness of vehicle v while arriving at customer k.
ρ_{ve}	Arrival time of vehicle v to node e.

3.3 | Mathematical Model Formulation

According to the parameters and variables defined, a MILP model will be developed for the problem.

$$TC = \sum_{v=1}^{V}\sum_{e=0}^{K}\sum_{\substack{e'=0\\ e\neq e'}}^{K} CO_{ee'v}\, W_{vee'} + \sum_{v=1}^{V}\sum_{e'=1}^{K} fc_{v}\, W_{v0e'} + \sum_{k=1}^{K}\sum_{v=1}^{V} ep_{k}\, ET_{vk} + \sum_{k=1}^{K}\sum_{v=1}^{V} lp_{k}\, LT_{vk}$$

$$+\sum_{j=1}^{n}\sum_{r=1}^{R} ho_{j}\left(Cmax - CT_{jr} - \sum_{l} Z_{jrl} q_{jl} \right). \tag{1}$$

$$\sum_{r} X_{ijr} = 1 \ \text{ for all } \ i, j, \tag{2}$$

$$\sum_{j} X_{ijr} = 1 \ \text{ for all } \ i, r, \tag{3}$$

$$X_{ijr} = X_{i-1jr} \ \text{ for all } \ i, j, r, \tag{4}$$

$$\sum_{l} Z_{jrl} = X_{ijr} \ \text{ for all } \ i, j, r, \tag{5}$$

$$Start_{ijr} \geq \sum\nolimits_{j'\neq j} C_{ij'r-1} - M\left(1 - X_{ijr}\right) \ \text{ for all } \ i, r, \tag{6}$$

$$C_{ijr} = Start_{ijr} + \sum_{j'} X_{ijr}\, X_{ij'r-1} ST_{jj'ir} + X_{ijr} * P_{ij} \ \text{ for all } i, j, r, \tag{7}$$

$$CT_{jr} = \max_{i}\left\{ C_{ijr} \right\} \ \text{ for all } \ j, r, \tag{8}$$

$$CT_{jr} \leq M X_{ijr} \ \text{ for all } \ j, r, \tag{9}$$

$$G_{1l} = \sum_{j} Z_{j1l}\left(CT_{j1} + q_{jl} \right) \ \text{ for all } \ j, l, \tag{10}$$

$$G_{rl} = \max\left\{ G_{(r-1)l}, \sum\nolimits_{j=1}^{J} Z_{jrl} CT_{jr} \right\} + \sum_{j} Z_{jrl} q_{jl} \ \text{ for all } r > 1, l, \tag{11}$$

$$C_{max} \geq G_{rl} \ \text{ for all } r, \tag{12}$$

$$\sum\nolimits_{v=1}^{V} \sum\nolimits_{\substack{e=0\\ e\neq k}}^{K} av_{ve} W_{vek} = 1 \ \text{ for all } k, \tag{13}$$

$$\sum\nolimits_{v=1}^{V} \sum\nolimits_{e,k\in S} W_{vek} \leq |S| - 1, S \subseteq \{1, 2, \ldots, K\}; 2 \leq |S| \leq K - 1. \tag{14}$$

$$\sum\nolimits_{\substack{e=0\\ e\neq k}}^{K} W_{vek} - \sum\nolimits_{\substack{e=0\\ e\neq k}}^{K} W_{vke} = 0 \ \ \forall v \ \&k \ \&av_{vk} = 1, \tag{15}$$

$$\sum_{e}\left(\rho_{ve} + ti_{vek} \right) \leq \rho_{vk} + \sum_{e}\left(1 - W_{vek} \right) M \ \text{ for all } v, k, \tag{16}$$

$$\rho_{v0} = C_{max} \ \text{ for all } v, \tag{17}$$

$$\sum\nolimits_{k=1}^{K} W_{v0k} \leq 1 \ \text{ for all } v, \tag{18}$$

$$\sum\nolimits_{e=0}^{K} \sum_{\substack{k=1\\ e\neq k}}^{K} \sum_{j=1}^{J} pn_{jk} W_{vek} \leq Ca_{v} \ \text{ for all } v. \tag{19}$$

According to the parameters and variables defined, a MILP model will be developed for the problem.

$$ET_{vk} \geq ed_{k} \sum\nolimits_{e=0}^{K} W_{vek} - \rho_{vk} \ \text{ for all } k \,\&\, v. \tag{20}$$

$$LT_{vk} \geq \rho_{vk} - ld_k \quad \text{for all} \;\; k \,\&\, v. \tag{21}$$

$$X_{ijr}, Z_{jrl}, W_{vee'} \; \text{are Binary Variables} \quad \text{for all} \;\; i,j,r,l,e,v. \tag{22}$$

$$Cmax, CT_{jr}, Start_{ijr}, C_{ijr}, ET_{vk}, LT_{vk}, G_{rl}, \rho_{vk} \geq 0 \quad \text{for all} \;\; i,j,r,k,l,v. \tag{23}$$

The *Objective Function (1)* consists of five terms; the first and second terms of which are to minimize the total costs of routing and vehicle usage, respectively. The third and fourth terms calculate the total penalties for earliness and tardiness of the delivery time window, respectively. The fifth term minimizes the cost of holding orders in the warehouse of final products. *Constraints (2)* and *(3)* indicate that each job in each processing is processed only in a specific priority and vice versa; each position contains only 1 job. *Constraint (4)* states that the position of each component of a given job must be the same in all machines. In other words, a unique position is assigned to each job, which is the same in processing all of its components. Based on *Constraint (5)*, a job should only be in one position of the assembly machine sequence. In addition, the index of job j in the first stage (r) is transferred to the second stage according to this relation. *Constraint (6)* states that the start time of processing job j on machine i at position r is higher than the completion time of the job preceding job j on machine j (job at position r-1) if Xijr=1. In other words, the processing of job j on the machine i at position r can begin immediately after the job at position r-1 is completed. In *Constraint (7)*, the completion time of job j on machine i is calculated. Accordingly, the completion time of each job on each machine equals the sum of the start time, setup time and processing time. *Constraint (8)* indicates that the completion time of job j at the production stage is equal to the completion time of the last component of job j. Also, *Constraint (9)* states that the decision variable CTjr takes value only if Xijr=1. Based on *Constraint (10)*, the completion time of assembly of the order placed at the first position of the assembly machine is equal to the completion time of production its components plus its assembly time. Based on *Constraint (11)*, the completion time of the assembly of jobs at the second position and thereafter equals the maximum completion time of the previous job and the completion time of the production of the components of the job at that position plus the completion time of the job at the position r. *Constraint (12)* calculates the completion time of the assembly of the final order. Based on *Constraint (13)*, the customer e is served only by one of the authorized vehicles. Based on *Constraint (14)*, it is not possible to create a sub tour. Based on *Constraint (15)*, if arrival at the node of customer e takes place by vehicle v, the exit from it must take place only by vehicle v. *Constraint (16)* computes the arrival time of the vehicle at each node based on the time traveled by the vehicle and the origin node. The start time of the vehicle's transportation is equal to the completion time of the last order. This Constraint is shown in *Eq. (17)*. According to *Constraint (18)*, each vehicle can remain in the origin node or, if necessary, can depart from the origin node to at most one of the customer nodes. The capacity constraint of each vehicle is shown in *Constraint 19*. *Constraints (20)* and *(21)* calculate the earliness and tardiness of each vehicle, respectively. *Constraints (22)* and *(23)* represent the type of decision variables.

3.4 | Linearization

Given the nonlinearity of the *Constraints (7)*, *(8)* and *(11)*, the following describes how to linearize these Constraints and consider them in the model.

The following variables are used to linearize nonlinear constraints:

$xx_{ijj'r}$: Binary variable for linearizing *Constraint (7)*.

a_{ijr}: Binary variable for linearizing *Constraint (8)*.

β_{jr}: The variable required to linearize *Constraint (11)*.

The non-linearizing factor in *Constraint (7)* is $xx_{ijj'r} = X_{ijr} X_{ij'r-1}$. *Constraint (7)* is replaced by the following relations.

$$xx_{ijj'r} \le X_{ijr} \quad \text{for all } i,j,j',r. \tag{24}$$

$$xx_{ijj'r} \le X_{ij'r-1} \quad \text{for all } i\ ,j,j',r. \tag{25}$$

$$xx_{ijj'r} \ge X_{ijr} + X_{ij'r-1} - 1 \quad \text{for all } i,j,j',r. \tag{26}$$

$$C_{ijr} = \text{Start}_{ijr} + \sum_{j'} xx_{ijj'r} ST_{jj'ir} + X_{ijr} * P_{ij} \quad \text{for all } i,j,r. \tag{27}$$

According to the relation $CT_{jr} = \max_i \left\{ C_{ijr} \right\}$, *Constraint (8)* is a nonlinear equation. For linearization, this relation is replaced by the following Constraints;

$$CT_{jr} - C_{ijr} \ge -M\left(1 - \alpha_{ijr}\right) \quad \text{for all } i,j,r. \tag{28}$$

$$CT_{jr} - C_{ijr} \le \ M\left(1 - \alpha_{ijr}\right) \quad \text{for all } i,j,r. \tag{29}$$

$$C_{i'jr} C_{ijr} + M(1 - \alpha_{ijr}) \quad \text{for all } j,i,i',r; i \ne i'. \tag{30}$$

$$\sum_{i}^{I} = \alpha_{ijr} = 1 \quad \text{for all } j,r. \tag{31}$$

The non-linearizing factor of the *Constraint (11)* is $\beta_{jrl} = Z_{jrl} CT_{jr}$. They can be linearized by considering the following relations:

$$\beta_{jrl} \le Z_{jrl}.M \quad \text{for all } j\ ,r,l, \tag{32}$$

$$\beta_{jrl} \le CT_{jr} \quad \text{for all } j\ ,r,l, \tag{33}$$

$$\beta_{jrl} \ge CT_{jr} - \left(1 - Z_{jrl}\right).M \quad \text{for all } j\ ,r,l. \tag{34}$$

3.5 | Non-Integrated Model

This section presents a non-integrated model. In this approach, first the mathematical model of production and assembly (first stage) is optimized. Then the optimal solution, as a parameter, is transferred to the mathematical model of the distribution (second stage). By optimizing the second stage model, the best distribution of orders among customers is determined. The first stage mathematical model consists of the following relations:

$$\text{Min } Z1 = \sum_{j=1}^{n} \sum_{r=1}^{R} ho_j \left(\text{Cmax} - CT_{jr} - \sum_{l} Z_{jrl} q_{jl} \right). \tag{35}$$

Constraints (2)-(6), Constraints (9) to *(10), Constraint (12),* and *Constraints (24)-(34),* which $CTjr$ and $Zjrl$ are decision variables as described above.

After optimizing the first stage model, the $Cmax$ value is transferred to the second stage model as a parameter. This model includes the following constraints:

$$\text{Min } Z2 = \sum_{v=1}^{V} \sum_{e=0}^{K} \sum_{\substack{e'=0 \\ e \ne e'}}^{K} CO_{ee'} W_{vee'} + \sum_{v=1}^{V} \sum_{e'=1}^{K} fc_v W_{v0e'} + \sum_{k=1}^{K} \sum_{v=1}^{V} ep_k ET_{vk} + \sum_{k=1}^{K} \sum_{v=1}^{V} lp_k LT_{vk}. \tag{36}$$

Constraints (13)-(21), which $Wvee', Wvoe'$ and ET_{vk} are decision variables as described above.

3.6 | Solution Methods

In this section, the proposed solutions method which are WOA, improved WOA, and GA are presented.

3.7 | Whale Optimization Algorithm (WOA)

Whales are cetaceans with a long tail. The interesting thing about whale life, which is the inspiration for this algorithm, is the way of feeding and hunting in humpback whales, known as bubble netting. In this way, each whale releases air bubbles beneath the sea and creates walls of rising air in the water. The fishes that are inside the air wall move toward the center of the circular bubble territory due to fear, and immediately the whale swallows many of them while rising from the water. This algorithm was proposed by Mirjalili and Lewis [39]. According to the WOA, humpback whales are able to detect and surround the prey's position. Because the optimal position in the search space is unclear, the WOA assumes that the best available solution is the target prey or a point near to it. Once this point is determined, the search for other optimal points and the updating of position is continued. This behavior is represented by the following equations.

$$D = | c.X^*(t) - X(t) | . \tag{37}$$

$$\left(t+1\right) = X^*\left(t\right) - A.D. \tag{38}$$

In the above equations, t represents the iteration of the algorithm, C and A the coefficients vectors, $X*$ the best-obtained position, and X the current Whale position. It should be noted that the value of $X*$ is updated in each iteration. The following equations are used to determine A and C values:

$$A = 2a.r - a. \tag{39}$$

$$C = 2.r. \tag{40}$$

Where, a is a vector that controls the variation of the solutions and its initial value is 2 that may reduce to 0 in different iterations. r is also a random vector ranging from 0 to 1. To implement the WOA, it is necessary to define the position of the whales based on a justified solution to the problem. In this study, the position of each whale is defined as a set consisting of 4 different vectors. These vectors are defined as follows. The first vector is a vector containing J cells that shows the sequence of jobs. All these cells take numbers ranging from 0 to 1. Ordering the numbers from largest to smallest can determine the sequence of jobs. The second vector is a vector of J cells ranging from 0 to 1. This vector shows the assignment of jobs to assembly machines. To describe it more clearly, suppose there are three assembly machines. Cells with values ranging from 0 to 0.334 (1 divided by 3) are assigned to assembly machine 1. Cells ranging from 0.334 to 0.664 (2 divided by 3) are assigned to assembly machine 2, and finally those ranging from 0.664 to 1 are assigned to assembly machine 3. The third vector is a vector with E cells ranging from 0 to 1. Each cell, depending on its value, represents the assignment of the customer to a vehicle. The way customers are assigned to machines is exactly in accordance with the procedure described for assigning jobs to machines. For example, if there are two machines, cells with a value of less than 0.5 are assigned to machine 1, and cells with a value greater than 0.5 are assigned to machine 2. The fourth vector has a length of E that shows the priority of customers in visits. Customer priority is determined by ordering the numbers of this vector from largest to smallest. For example, if the number of orders is 3, the number of assembly machines is 2, the number of vehicles is 2, and the number of customers is 3, four vectors relating to a hypothetical solution are according to *Table 3*.

Table 3. An example of a justified solution in the WOA.

Vector 1	0.26	0.48	0.64
Vector 2	0.35	0.27	0.95
Vector3	0.17	0.84	0.35
Vector 4	0.53	0.37	0.18

According to the first row, the sequence of jobs on different production machines is 3-2-1. In the assembly stage, according to the second row, orders 1 and 2 are assigned to assembly machine 1 and order 3 to assembly machine 2. In the distribution stage, according to the third row, customers 1 and 3 are assigned to the first machine and customer 2 to the second machine. To determine the order of customer visits by vehicle 1 (visit order of customers 1 and 3), see the fourth row. According to this row, the number related to customer 1 is greater than that related to customer 3 ($0.18 < 0.53$), so customer 1 is first visited and then customer 3 is visited. For vehicle 2, because it visits only one customer, the customer priority is not applicable.

3.8 | Improved Whale Optimization Algorithm

Despite the suitability of the WOA search algorithm, this algorithm converges quickly in some cases. Also, this algorithm greedily seeks to improve the existing set of solutions, which causes the algorithm to trap in local optima. Due to the problems mentioned for the WOA search algorithm, the necessary changes will be made in the following to develop this algorithm and increase its efficiency. This development is inspired by the study of Alinaghian and Goli [40].

I. In the step of generating a new solution in the WOA algorithm, each cell of the solution vector is randomly generated from the values of the best existing solution, according to *Eqs. (37)* and *(38)*. But in the new method, first a vector is generated from the combination of all existing solutions and then new solutions will be generated with its help and based on a random process.

II. The procedure for setting parameter a exits from the uniform mode-from 2 to 0-and will be set in such a way that the components of the new solution move towards the best possible solution.

III. In order to update the set of solutions, first the answers are sorted according to the value of the objective function. Then the solutions that have the following two characteristics are removed from the set of solutions: 1) have a weak objective function, 2) have similar solutions in the set of solutions.

In order to better understand the developed meta-heuristic algorithm, the IWOA algorithm is described step by step considering these changes.

Step 1. Determining the problem parameters and the algorithm.

Step 2. Generating random initial values for the set of problem solutions and calculating the value of the objective function of each of them (F(xj)).

Step 3. Generating new solutions:

$$FDR_i = \max_{j,l}\{\frac{f(x_i)-f(x_j)}{\left|x_{jl}-x_{il}\right|}\}. \tag{41}$$

$$D = \left|c.X_{i^*}(t) - X(t)\right|. \tag{42}$$

$$X(t+1) = X_{i^*}(t) - A.D. \tag{43}$$

In this relation, $f(x_i)$ and x_{il} are respectively the value of the objective function and the value of the lth component of the new solution, also $f(x_j)$ and x_{jl} are respectively the value of the objective function and the value of the lth component of the jth existing solution:

$$x' = f(worst) - f(best). \tag{44}$$

$$p = \frac{1}{\sqrt{2\pi}} e^{-\frac{x'^2}{2}}. \tag{45}$$

$$a' = 0.1 \times (1-p). \tag{46}$$

In the above equations, f (best) and f (worst) are the values of the objective function and the worst solution in the set of existing solutions, respectively. Finally, parameter a' is placed in equations 39 and 40 to complete the process of generating new solutions. *Fig. 2* shows the flowchart of the IWOA algorithm. Moreover, the pseudocode of IWOA is illustrated as *Algorithm 1*.

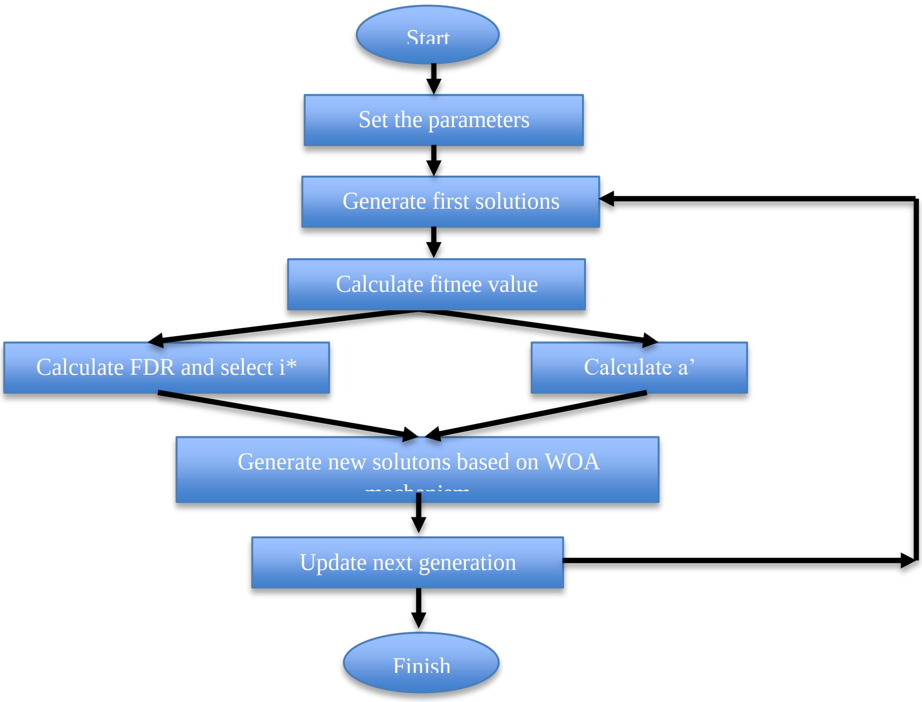

Fig. 2. The flowchart of proposed IWOA.

Algorithm. 1. The pseudocode of proposed IWOA.

```
Randomly initialize tha whale population (X)
Evaluate the fitness value of each whale
Specify the best search agent $X^*$
While t<Max_iter
Calculate a'
Calculate FDR and select $i^*$
For each search agent
Generate new solution using Eqs. (37)-(40) and save it as $X(t+1)$.
End for
Evaluate $X(t+1)$
Update $X^*$
End while
Report $X^*$
```

3.9 | Genetic Algorithm

The most important characteristic of Genetic Algorithm (GA) is simplicity. The steps of the GA are illustrated in Algorithm 3. First, the solution to the problem is defined in the form of a chromosomal structure (coding). By introducing the fitness function, the quality of solutions in each chromosome is expressed as a number. Then, a specific number of chromosomes is generated randomly (or quasi-randomly). These chromosomes are known as the primary population and are evaluated based on the fitness function. Now, two chromosomes are selected for reproduction and using these two chromosomes, a new chromosome is generated (mating). With a specific probability, a number of genes of some chromosomes are changed. Performing the selection, mating and mutation steps creates a new population (generation) of chromosomes. If the chromosomes converge to the optimal response, the reproduction operation will be stopped. Otherwise, each generation will be produced from the previous generation until the desired solution is obtained or the stop criterion of the algorithm is applied [31], [41], [42].

First, a coding system should be defined. This coding system is called the chromosome. The chromosome used in this study is a chromosome with real numbers between 0 and 1. The structure of this chromosome is exactly the same as that proposed for the WOA.

Next, it is necessary to create the initial population. This initial population is randomly generated in the range between 0 and 1. Then the fitness value of each solution is calculated. The fitness value exactly equals the total cost of production, assembly, and distribution. Then, a set of solutions is selected as the parent using the roulette wheel method. Of the parents, two new solutions (children) are generated from each two parents (P1 and P2) using linear combination. For this purpose, one parameter α is randomly generated in the range of [-sigma, + sigma] (the sigma is the control parameter that should be set). Then, new solutions (SP1 and SP2) are generated by the *Eqs. (47)* to *(48).*

$$\mathrm{SP1} = \alpha \mathrm{P1} + \left(1-\alpha\right)\mathrm{P2}. \tag{47}$$

$$\mathrm{SP2} = \alpha \mathrm{P2} + \left(1-\alpha\right)\mathrm{P1}. \tag{48}$$

In such circumstances, it can be assured that the solutions are the perfect combination of the parent's solutions. Finally, it is checked that the cells of SP1 and SP2 take a value between 0 and 1. If a cell takes a value less than 0, it is changed to 0. If a cell takes a value greater than 1, it is changed to 1. This operation is performed with the Probability of Crossover (PC) in each iteration. Then the mutation is performed. Therefore, a cell is selected from a chromosome, and then its value is replaced by a random value. This operation is performed with the probability of mutation (pm) in each iteration. In the next stage, a number of solutions will be selected equal to the number of Pop size from the set of available solutions (parents, Crossover solutions, mutation solutions) and included in the next iteration. The flowchart of proposed GA is as *Fig. 3.*

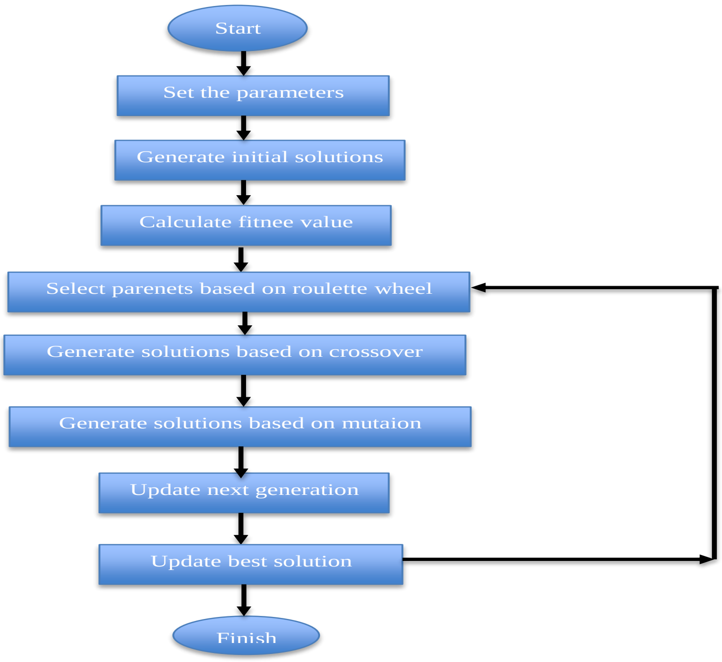
Start
Set the parameters
Generate initial solutions
Calculate fitnee value
Select parenets based on roulette wheel
Generate solutions based on crossover
Generate solutions based on mutaion
Update next generation
Update best solution
Finish

Fig. 3. The general structure of the GA.

3.10 | Computational Results

The aim of this section is fourfold: 1) to validate small-scale mathematical models, 2) to investigate the efficiency of proposed solution methods, 3) to present a case study, and 4) to conduct and report a sensitivity analysis.

3.11 | Validation of Mathematical Models

To validate the integrated mathematical model and the meta-heuristic algorithms, a small-scale numerical example is implemented in this section and the results are presented. The data in this numerical example has been designed in a way that the optimal solution can be clearly understood. The goal is to produce three types of gearboxes, shown by the symbols A, B, and C. The factory produces and sends for two customers in cities C1 and C2. *Table 4* shows data on customer demand, production and assembly times, and holding costs for each product. It should be noted that each time unit is considered to be 5 minutes.

Table 4. Data on processing and assembly times and holding cost of final products.

Job	A	B	C
The demand for customer 1 for each products	2	7	0
The demand of customer 2 for each products	0	1	4
Processing time of component 1 on machine 1 in the first stage	1	1	1
Processing time of component 2 on machine 2 in the first stage	2	2	2
Assembly time on the first assembly machine in the second stage	3	3.2	2.2
Assembly time on the second assembly machine in the second stage	2.7	2.6	1.4
The holding cost of each unit of product in the warehouse	1	1	1

In the production stage, there is a setup time for the production of each component of the different gearboxes. Sequence-dependent setup time for all jobs is defined as one unit of time. This company uses one type of vehicle to deliver products to customers. This vehicle has a capacity of 100 units and a fixed cost of 50 units. The variable costs of applying vehicles are also shown in *Table 5*. The data in *Table 5* shows the distance between the factory and customer location. In this case, 1 monetary unit of cost is considered for each unit of travel time.

Table 5. Data on travel time between different nodes.

	Factory	E1	E2
Factory	0	40	85
E1	40	0	133
E2	85	133	0

In addition, the delivery time windows and other data on customer delivery are shown in *Table 6*.

Table 6. Data on delivery time window, service time, and earliness and tardiness penalties of customer delivery.

Customer	E1	E2
Delivery time window	[50 , 60]	[20 , 60]
Service time	1	0.6
Earliness penalty	5	5
Tardiness penalty	15	17

We used a PC with Core I5 CPU processor under the windows 8.1 operating system with 4GB of RAM. The mathematical MIP model of the problem was implemented in GAMS software and solved with CPLEX 24.0.1 solver. The proposed algorithms were coded by MATLAB 2017 R1 software. The above data was entered the GAMS optimization software as the parameters of the integrated mathematical model, and then the model was optimized. The optimal solution to the problem is equal to 2468.1 monetary units. The Cmax value is also 9.4. In the production stage, the optimal sequence of jobs is defined as 1-2-3. *Fig. 4* shows the scheduling of jobs on different machines.

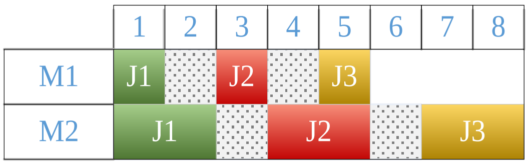

Fig. 4. Scheduling of jobs in the production stage in the integrated model.

In the assembly stage and in the optimal solution, job 2 is assigned to assembly machine 1, and jobs 1 and 3 to the assembly machine 2. *Fig. 5* illustrates the scheduling of these jobs during the assembly stage.

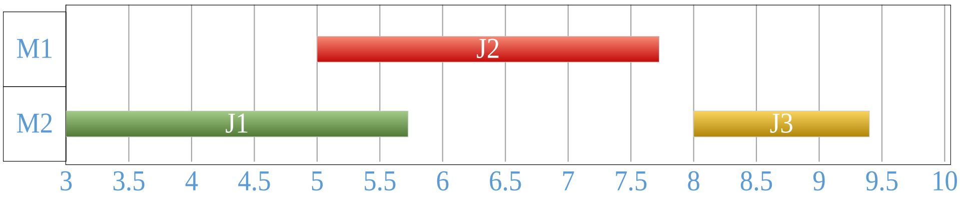

Fig. 5. Assembly scheduling in integrated model.

As *Fig. 3* illustrates, each job is assembled immediately after its production is completed, which confirms the accuracy of results. According to manual calculations, the Cmax value is 9.4, which confirms the validity of the results. In the following, the routing of vehicles between customers is specified. *Fig. 6* illustrates the created tour.

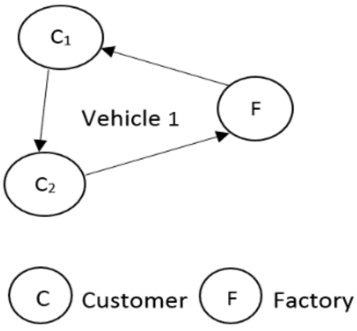

Fig. 6. The optimal solution of product distribution in the integrated model.

After comparing the arrival time of vehicle to each customer with its time window, it is revealed that the earliness and Tardiness delivery of orders to customer 1 and 2 are 0.6 and 122.4, respectively.

Data analysis shows the accuracy of the outputs, and the study of similar solutions shows that the lowest possible cost is 2468.1. Therefore, the mathematical model and its results can be considered as reliable. In the next stage, it will be necessary to investigate the validity of the WOA and GA. So, the problem defined is optimized by each of WOA, IWOA and GA in this section. *Fig. 7* illustrates the convergence of the WOA. Also *Fig. 8* illustrates the convergence of IWOA and finally *Fig. 9* shows the convergence of GA.

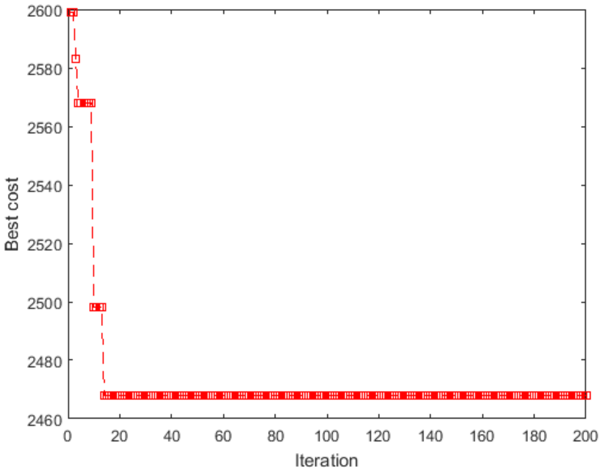

Fig. 7. The convergence of WOA.

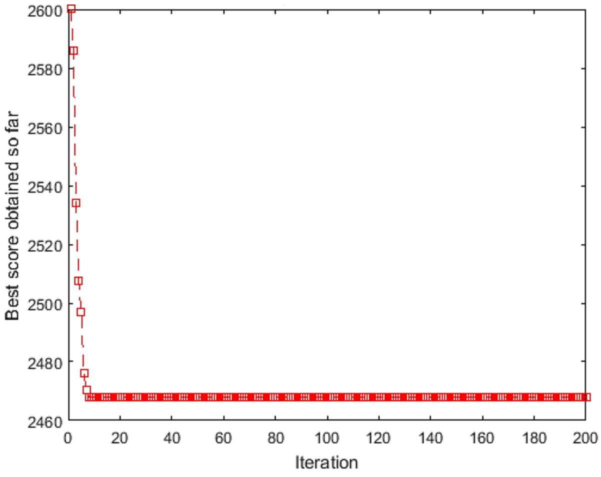

Fig. 8. The convergence of IWOA.

As illustrated in *Fig. 8*, WOA obtained the objective function of 2600 in its first iteration. After various iterations, this value reached 2468.1 in iteration 13. Moreover, in *Fig. 9*, IWOA is converged to the optimal solution after 8 iterations. This suggests that WOA and IWOA can converge to the optimal solution of the problem after a small number of iterations, and therefore its results will be valid. In addition, the proposed improvement for WOA leads to speed up the convergence in about 60%.

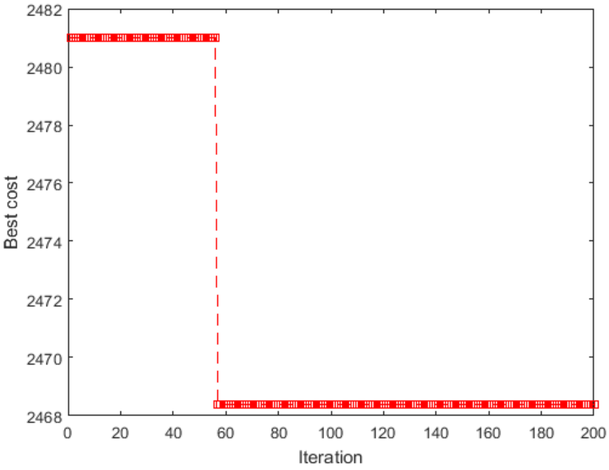

Fig. 9. The convergence of GA.

As can be seen in *Fig. 9*, the GA was able to converge to the overall optimal solution of the problem. Therefore, the results of this algorithm will be valid. The difference between the WOA, IWOA and GA is that the WOA and IWOA converged to the optimal solution after 13 and 8 iterations respectively, but the GA reached this convergence after 57 iterations. Therefore, the IWOA was strongest in this regard. While the WOA and IWOA started at the range of 2600, the GA started at the range of 2481. Therefore, the convergence rate of the WOA and IWOA was higher. In other words, the improvement that the WOA and IWOA is equal to 2600-2468.1 = 118.9. By dividing this phrase by 13 iterations, the convergence rate of the WOA is obtained as 9.1 and dividing by 8 iterations, the convergence of IWOA is obtained as 14.8. In other words, an average cost reduction of 14.8 units occurred per iteration before convergence in IWOA. This index is equal to 2481-2468.1 = 12.9 for the GA and the convergence rate is 0.22. In other words, the GA produced an average improvement of 0.22 units per iteration before convergence.

3.12 | Investigating the Efficiency of Proposed Solution Methods

In this section of the computational results, the performance of WOA, IWOA and GA will be investigated. In this regard, initially 40 instances in small, medium and large scales are generated. Data on these problems is shown in *Table 7*. The values of the parameters for each of the instances are randomly generated from continuous uniform distribution. The lower limit and the upper limit of each parameter are shown in *Table 8*.

These problems were optimized in the GAMS environment for exact solution optimization, and in the MATLAB environment with WOA, IWOA and GA. The time limit for optimization was considered as 3600 seconds. The results are presented in *Table 9*. In this table, Z represents the value of the objective function, T (in seconds) is the solution time, and Gap (%) is the relative error compared to exact method (GAMS).

Table 7. Scale of generated instances.

Size	Samples	I	J	E	V	L
Small scale	P1	2	3	3	1	2
	P2	3	3	3	1	2
	P3	3	4	3	1	2
	P4	3	4	4	1	2
	P5	3	4	5	2	3
Median scale	P6	5	5	5	3	3
	P7	4	5	5	2	3
	P8	5	6	6	2	3
	P9	5	7	7	3	4
	P10	6	8	9	3	3
Large scale	P11	10	10	10	4	5
	P12	12	15	11	5	6
	P13	14	17	15	6	7
	P14	16	20	17	7	9
	P15	18	23	19	9	10
	P16	20	25	20	10	11
	P17	25	27	25	13	13
	P18	30	30	27	15	14
	P19	40	40	35	20	20
	P20	50	50	40	30	25
	P21	60	60	50	30	30
	P22	70	70	60	40	35
	P23	80	80	70	40	40
	P24	90	90	80	50	45
	P26	100	100	90	50	50
	P26	110	100	100	60	55
	P27	120	120	110	60	60
	P28	130	120	120	70	65
	P29	140	140	130	70	70
	P30	150	140	140	80	75
	P31	160	160	150	80	80
	P32	170	160	160	90	85
	P33	180	180	170	90	90
	P34	190	180	180	100	95
	P35	200	200	190	100	100
	P36	210	200	200	110	105

Table 7. Continued.

Size	Samples	I	J	E	V	L
	P37	220	210	210	110	110
	P38	230	210	220	120	115
	P39	240	220	230	120	120
	P40	250	220	240	130	125

Table 8. Upper and lower limits of parameter values.

Parameters	Lower Bound	Upper Bound
Transportation time	20	140
Service time	0.5	1
Assembly time	1.4	3.4
Demand	0	5
Production time	1	4
Holding cost	1	4
Earliness penalty	3	5
Tardiness penalty	10	20
The cost of travel	30	135
Vehicle fixed cost	50	60
Vehicle capacity	1000	1200
Lower limit of the time window	20	90
Upper limit of the time window	90	140
Setup time	1	2

Table 9. Results of optimization of various problems.

	GAMS		WOA			IWOA			GA		
Samples	Z	T	Z	T	Gap	Z	T	Gap	Z	T	Gap
P1	2468.1	1.02	2468.1	0.69	0.00%	2468.10	0.74	0.00%	2468.1	3.43	0.00%
P2	2527.6	2.83	2527.6	4.88	0.00%	2527.60	4.88	0.00%	2527.6	6.06	0.00%
P3	2768.6	6.47	2811.6	5.11	1.53%	2811.60	5.53	1.53%	2847.8	7.34	2.78%
P4	1802.8	20.52	1867.8	5.31	3.48%	1860.20	5.82	3.09%	1956.7	7.53	7.87%
P5	3176.2	176.41	3257.7	5.59	2.50%	3203.40	5.77	0.85%	3261.6	7.81	2.62%
P6	2469.6	870.1	2505.7	6.67	1.44%	2493.10	7.31	0.94%	2539.1	8.82	2.74%
P7	2494.2	3600	2594.6	7.11	3.87%	2590.90	7.16	3.73%	2666.1	8.91	6.45%
P8	-	-	5435.2	6.77	0.05%	5432.48	6.86	0.00%	5450	8.44	0.32%
P9	-	-	5925.9	7.68	0.24%	5911.71	8.12	0.00%	5991.4	9.15	1.33%
P10	-	-	10449.3	8.24	0.30%	10417.97	8.38	0.00%	10572.8	9.94	1.46%
P11	-	-	15559.4	10.23	0.66%	15456.43	11.17	0.00%	16330.5	12.02	5.35%
P12	-	-	40140.9	13.31	0.83%	39809.42	15.73	0.00%	41197.2	15.72	3.37%
P13	-	-	63812.5	15.47	3.40%	63646.83	17.77	3.14%	61645.4	18.37	0.00%
P14	-	-	112338.9	18.3	3.02%	111234.88	18.71	2.05%	108951.5	21.94	0.00%
P15	-	-	160383.9	21.86	0.40%	159745.69	23.37	0.00%	167182	25.47	4.45%
P16	-	-	203856.4	24.36	0.52%	202792.18	28.22	0.00%	206792.5	28.74	1.93%
P17	-	-	224476.3	29.36	0.81%	222647.77	30.25	0.00%	225363.8	34.39	1.21%
P18	-	-	308709	32.02	0.93%	305827.44	37.96	0.00%	310627.1	38.1	1.55%
P19	-	-	707211.9	47.04	0.99%	700228.22	50.46	0.00%	722761.3	56.14	3.12%
P20	-	-	1393914.8	69.07	0.71%	1384042.51	79.83	0.00%	1441985.4	78.4	4.02%
P21	-	-	1474033.6	73.96109	1.95%	1473993.55	86.22	1.95%	1445261.2	85.58	0.00%
P22	-	-	1514966.5	77.51197	0.78%	1507566.97	90.98	0.30%	1503110.9	100.37	0.00%
P23	-	-	1586944.4	83.80926	0.00%	1586944.45	85.67	0.00%	1690395.3	104.30	6.12%
P24	-	-	1648002.2	89.64728	0.43%	1640941.26	95.57	0.00%	1879600.5	104.74	12.70%
P25	-	-	1759013.3	90.45446	0.00%	1759013.34	96.16	0.00%	2097065.9	121.79	16.12%
P26	-	-	1859640.8	97.89633	0.00%	1859640.77	99.07	0.00%	2334516.7	132.68	20.34%
P27	-	-	1898552.4	103.3518	0.00%	1898552.37	118.97	0.00%	2446831.0	149.12	22.41%
P28	-	-	1958180.3	108.4347	0.82%	1942096.92	114.81	0.00%	2449723.4	161.90	20.72%
P29	-	-	2068810.9	123.0877	0.00%	2069128.64	140.90	0.02%	2657742.2	191.73	22.16%
P30	-	-	2084668.5	131.1225	0.00%	2097761.34	142.28	0.62%	2834937.8	207.59	26.47%
P31	-	-	2120083.1	143.3223	0.00%	2130588.50	143.42	0.49%	2908072.7	230.02	27.10%
P32	-	-	2271524.2	153.0421	0.36%	2263271.44	180.52	0.00%	3178261.6	232.05	28.79%

Table 9. Continued.

Samples	GAMS		WOA			IWOA			GA		
	Z	T	Z	T	Gap	Z	T	Gap	Z	T	Gap
P33	-	-	2280345.1	171.9941	0.29%	2273815.41	188.03	0.00%	3563090.8	262.49	36.18%
P34	-	-	2294721.4	206.2778	0.06%	2293410.52	234.82	0.00%	3722680.6	290.42	38.39%
P35	-	-	2432103.7	214.6102	0.00%	2442126.30	246.32	0.41%	4022717.4	338.17	39.54%
P36	-	-	2538269.2	220.4266	0.25%	2531966.67	252.29	0.00%	4112688.6	359.08	38.44%
P37	-	-	2548623.6	251.7006	0.54%	2534978.60	266.94	0.00%	4621513.2	384.91	45.15%
P38	-	-	2696561.3	294.6928	0.86%	2673321.19	328.58	0.00%	4715799.6	411.56	43.31%
P39	-	-	2941324.5	318.2205	0.02%	2940708.34	367.36	0.00%	5162264.5	465.35	43.03%
P40	-	-	2990058.0	329.2949	0.61%	2971686.19	384.68	0.00%	5183784.6	541.62	42.67%
Average			1155917	90.54822	0.82%	1153416.531	100.941	0.48%	1646829.41	132.055	14.50%

As seen, GAMS was only able to optimize 7 problems. Moreover, the average WOA solution time is 90.54 seconds and its average error is 0.82%. These indices are 100.94 and 0.48% in the IWOA. The first comparisons demonstrate that the proposed improvement leads to find better solutions with lower costs. The average GA solution time is 132.05 seconds and its average error is 14.23%. Therefore, the IWOA performed better in terms of both solution time and error in comparison to GA too. It should be noted that the gap% obtained is based on the comparison of each approach with the exact solution method (GAMS).

Fig. 10 illustrates the comparison of solution times of GAMS and meta-heuristic methods for seven GAMS-solvable problems. *Fig. 11* illustrates the comparison of solution times of meta-heuristic methods for the 40 solved problems, and *Fig. 12* shows the comparison of the values of the objective functions for the two meta-heuristic methods.

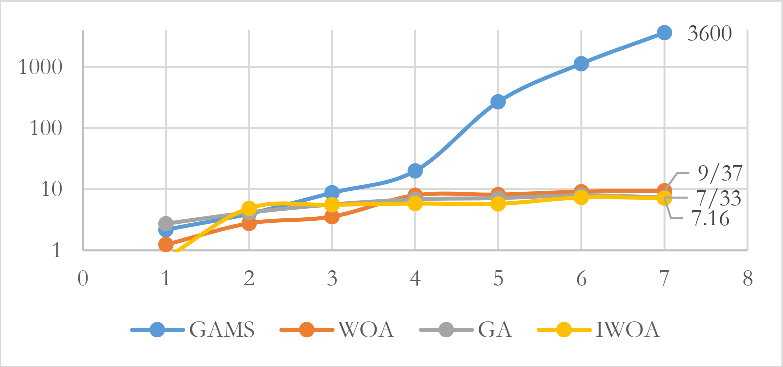

Fig. 10. Comparison of solution times of GAMS and meta-heuristic algorithms in small scale.

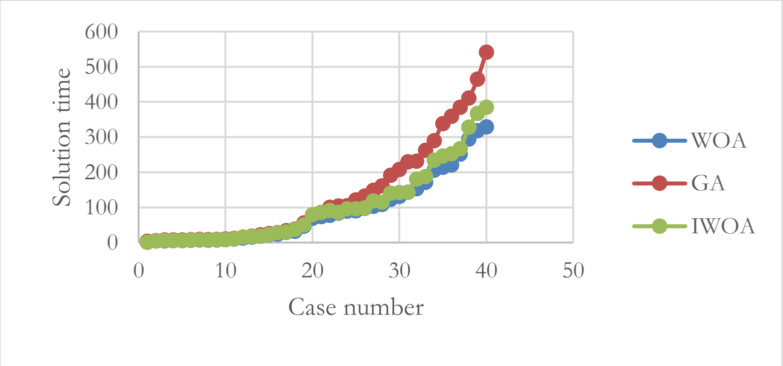

Fig. 11. Comparison of solution times between meta-heuristic algorithms in a test problem.

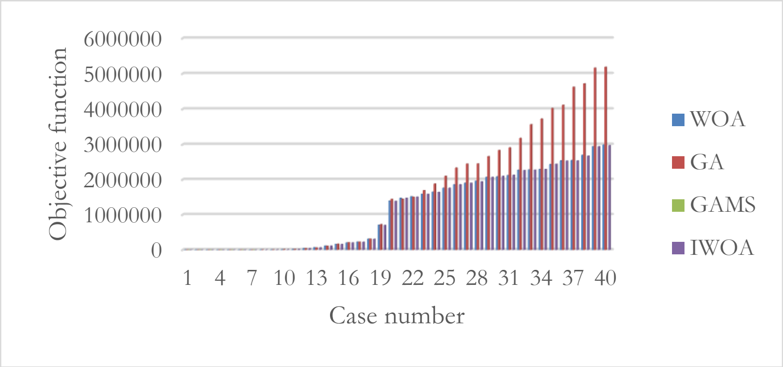

Fig. 12. Comparison of the value of the objective function.

As can be seen, for all solved instances, GA's solution time was longer than the WOA and IWOA. Moreover, for solving large-scale problems, the GA exhibited better performance than WOA only in problems 13 and 14, and in other problems, WOA provided a better objective function. On the other hand, IWOA spend more time than WOA but its convergence faster and achieve more quality solutions. Therefore, the efficiency of IWOA compared to WOA and GA could be well explained.

4 | Case Study

To conduct validation, the integrated and non-integrated mathematical models presented were implemented in an industrial gearbox manufacturing factory in Iran. The results will be presented in this section. This industrial gearbox factory is located in Ar city and produces 7 types of gearboxes. These gearboxes are shown with the symbols A, B, C, D, E, F and G. The factory produces and sends products to six customers in the cities of As, Ta, Kh, Sh, Sm and Tr. *Table 10* presents customers demand data, production and assembly times, and holding costs for each product. It should be noted that each unit of time is considered to be 5 minutes.

In the production stage, there is a setup time for the production of each component of the different gearboxes. *Table 11* shows the setup time of each job on each machine. According to this table, the setup time of each job is determined dependent on the previous job in the sequence. For example, if job A is processed immediately after job B, the setup time for job A is two time units. These times are equal for all machines.

Table 10. Data on processing and assembly times, and holding costs of final products.

Job	A	B	C	D	E	F	G
Demand of customer 1 of products	2	7	0	3	1	2	2
Demand of customer 2 of products	0	1	4	2	1	1	1
Demand of customer 3 of products	4	0	2	3	2	2	2
Demand of customer 4 of products	5	2	2	0	3	3	3
Demand of customer 5 of products	3	1	3	2	3	1	1
Demand of customer 6 of products	2	3	1	2	2	1	1
Processing time of component 1 on machine 1 in the first stage	2	3	2	3	2	1	1
Processing time of component 2 on machine 2 in the first stage	1	1	2	2	4	2	2
Processing time of component 3 on machine 3 in the first stage	2	3	3	3	2	2	2
Processing time of component 4 on machine 4 in the first stage	1	3	3	2	4	3	3
Processing time of component 5 on machine 5 in the first stage	3	3	2	2	3	1	1
Processing time of component 6 on machine 6 in the first stage	3	3	3	1	1	2	3
Processing time of component 7 on machine 7 in the first stage	4	1	2	3	1	2	2
Assembly time on the assembly machine in the second stage	3	3	2	3	3	1	1
The holding cost of each unit of product in the warehouse	10	10	11	9	10	10	11

This company uses 3 types of vehicles to send products to customers. The capacity and fixed cost of using these vehicles are shown in *Table 12*. The variable costs of using vehicles are according to *Table 13*. The data in *Table 14* show the distance between the factory and the customer locations. In this case, for each unit of time (5 minutes), a distance of 5 km is travelled. One monetary unit of cost is considered per one monetary unit.

Table 11. Data on sequence-dependent setup time.

	A	B	C	D	E	F	G
A	1	2	1	2	2	1	1
B	1	2	1	2	1	1	1
C	1	1	1	2	2	1	1
D	2	1	1	1	2	1	1
E	1	1	2	1	1	1	2
F	1	1	1	1	1	1	2
G	1	2	2	1	1	1	2

Table 12. Data on capacity and fixed cost of vehicles.

Vehicle Type	1	2	3
Capacity	100	150	320
Fixed Cost	500	600	1000

Table 13. Data on travel time between different nodes.

	Ar	As	Ta	Kh	Sh	Sm	Tr
Ar	0	40	85	65	33	16	20
As	40	0	133	114	7	30	15
Ta	85	133	0	165	121	45	90
Kh	65	114	165	0	105	40	25
Sh	33	7	121	105	0	30	60
Sm	90	30	20	80	30	0	90
Tr	35	20	60	55	33	20	0

The above data was included in the GAMS optimization software as the parameters of the integrated mathematical model and the model was optimized. The optimal solution of the problem was obtained as 22462 monetary units. Moreover, the Cmax value was obtained as 28. In the production stage, the optimal sequence of jobs was determined as 6-5-2-7-3-4-1. *Fig. 13* illustrates the scheduling of jobs on different machines.

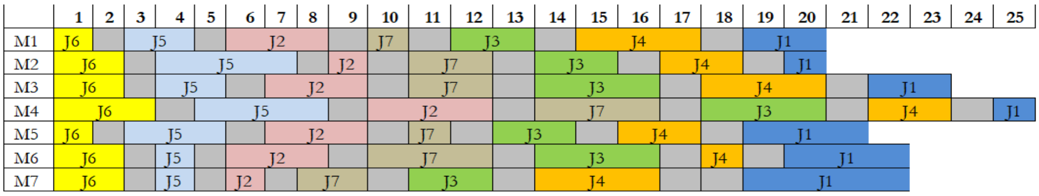

Fig. 13. Scheduling jobs in the production stage of the integrated model.

Based on the obtained results in *Fig. 13* in the optimal solution, different machines have idle times. The reason for this is that the integrated model seeks to complete the processing of the various components. The important point in *Fig. 13* is that the sequence of jobs is defined in a way that all setup times are equal to its minimum value *Fig. 14* illustrates the scheduling of these jobs during the assembly stage.

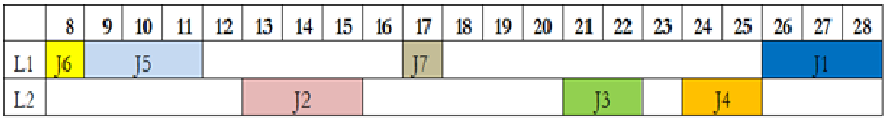

Fig. 14. Assembly scheduling in integrated model.

As illustrated in *Fig. 14*, each job is placed in the assembly stage immediately after production, which confirms the accuracy of the results. In addition, based on manual calculations, the Cmax value is equal to 28, which shows the validity of the results. In the following, the routing of vehicles between customers will be determined. For this purpose, vehicles 1 and 3 were used. The tours of each vehicle are illustrated in

Fig. 15. Based on the results, it is clear that all customers were visited once and each vehicle formed a tour which started and finished in the factory. This also confirms the accuracy of the results.

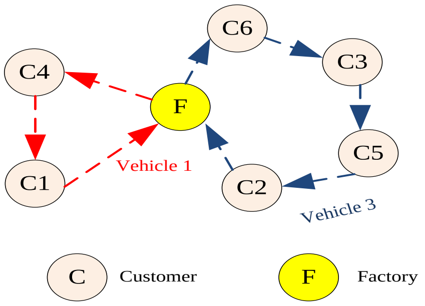

Fig. 15. The optimal solution of product distribution in the integrated model.

To evaluate the results of the integrated model, a two-stage model was optimized with data provided in *Tables 11-15*. In the first stage model, after optimization, the sequence of jobs was obtained as 6-5-2-7-4-3-1 and the Cmax value was equal to 28. In other words, the results of the first stage model are exactly equal to the results of the integrated model. Then the second stage model was optimized. *Fig. 16* illustrates the optimal routes of the second stage model.

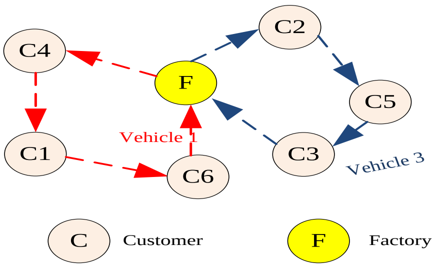

Fig. 16. Optimal routes in the non-integrated model.

The results of this section show that the routes formed in the non-integrated model are completely different from those in the integrated model. Therefore, earliness and tardiness can serve as a criterion to determine the superiority of either of the integrated and non-integrated approaches relative to the other one. *Table 14* shows comparison of earliness and tardiness as well as production and distribution costs in the two models. *Table 14* shows comparison between the different costs of the two models.

As can be seen, both models performed equally well. The reason for this is that the transportation of products started at 28. Moreover, the lower limit of the time window of all customers is less than this time. Therefore, it was not possible to create earliness in any of the models. In total, the integrated model had a tardiness of 142 time units, while the non-integrated model had a tardiness of 186. In other words, the integrated model saves 23.65% in delay time. An examination of the costs of the two models shows that the integrated model has a higher fixed cost than the non-integrated model. However, with expending higher fixed costs, 2.34% of variable costs will be saved. Regarding delays, the cost of tardiness is 2698 in the integrated model and 3361 in the non-integrated model, which the integrated model is 19.70% better than the non-integrated model. Overall, the total cost of distributing products among customers is 20196 monetary units in the integrated model and approximately 21760 monetary units in the non-integrated model, which about 7% financial savings is obtained in the integrated model. Therefore, a comprehensive study of the above models shows that the performance of the integrated model is much better than the non-integrated model; and simultaneous decision-making on production, assembly and distribution can significantly minimize system costs, and also increase customer satisfaction by reducing delays.

Table 14. Earliness and tardiness in integrated and non-integrated models.

	Customer	C1	C2	C3	C4	C5	C6
Integrated model	Earliness	-	-	-	-	-	-
	Tardiness	24	44	21	-	49	4
Non-integrated models	Earliness	-	-	-	-	-	-
	Tardiness	24	19	18	-	57	68

Table 15. Distribution costs in integrated and non-integrated models.

	Earliness Total	Total Tardiness	Distribution Fixed Cost	Distribution Variable Cost	Delays Cost	Total Distribution Cost
Integrated model	0	142	1500	17968	2698	20196
Non-integrated model	0	186	1500	18399	3361	21760
Percentage of Integrated model superiority	-	23.65%	0%	2.34%	19.70%	7.31%

4.1 | Sensitivity Analysis

The purpose of sensitivity analysis is to investigate the effect of fluctuations in important parameters of the mathematical model on the value of the objective function. This effect will be examined independently. In other words, by assuming other parameters constant, the effect of one parameter on the value of the objective function will be analyzed. In terms of cost factors, this analysis is quite clear. Fluctuations in any type of cost have a direct effect on the value of the objective function.

However, some parameters do not have a significant effect on the objective function. These parameters include setup time and capacity of the transport fleet. The effect of these parameters on the objective function will be examined below.

4.2 | Sensitivity Analysis of Setup Time

In the validation problem, the values of the setup time were presented. In this section, the values of this parameter fluctuate between -20% and +20% (*Table 16*, *Fig. 17*).

Table 16. Sensitivity analysis of setup time.

Percentage change in the parameter	-20%	-10%	0%	10%	20%
Objective function value	6050.8	6053.9	6053.9	6054.1	6057.9

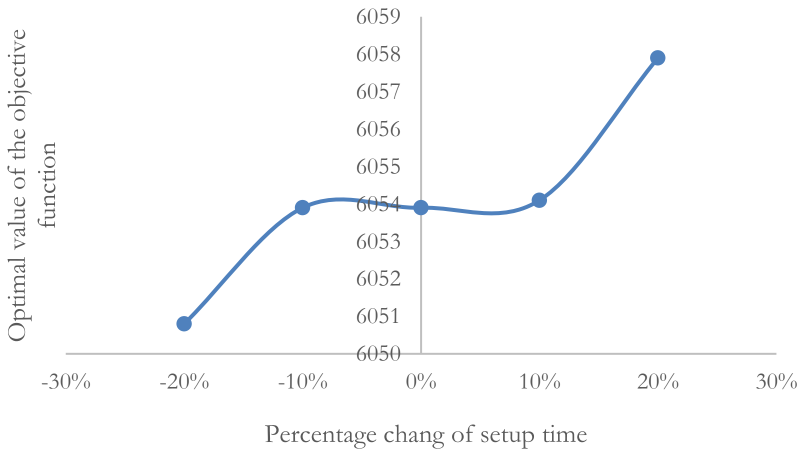

Fig. 17. Objective function values relative to setup time changes.

The results in *Fig. 17* and *Table 16* show that increasing the setup time can lead to an increase in total costs. It can be said that the increase of setup time increases the completion of time jobs. As a result, the distribution stage starts later and eventually the customers receive their orders with more delay. Therefore, the cost of delay in customer delivery increases. It should be noted that minor changes in this parameter did not influence the value of the objective function and only a severe increase of up to +20% leads to changes.

4.3 | Sensitivity Analysis of Transport Fleet Capacity

In the validation problem, the capacity values of the transport fleet were presented. In this section, the values of this parameter fluctuate between -20% and +20%, and in *Table 17* and *Fig. 18*, the results of the sensitivity analysis of this parameter are presented.

Table 17. Sensitivity analysis of transport fleet capacity.

Percentage change in the parameter	-20%	-10%	0%	10%	20%
Objective function value	6050.8	6053.9	6053.9	6054.1	6057.9

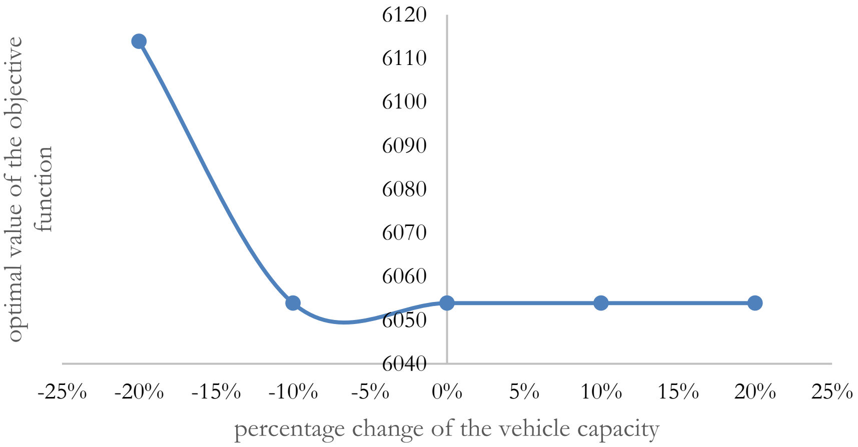

Fig. 18. The value of objective function relative to variations in transport fleet capacity.

As shown in *Fig. 18* and *Table 17*, no change occurs in the value of the objective function from -10% up to +20% fluctuations. The costs increased with decreasing the transport fleet capacity to -20%. The reason for this is that by reducing the capacity of vehicles, the model has to use more vehicles; and as a result, more fixed costs are imposed on the model, and therefore the optimal value of the objective function increases.

5 | Conclusion

In this study, the two-stage assembly flow shop problem and transport fleet routing were studied. The key innovation of this research was the integration of two-stage assembly scheduling and vehicle routing decisions. In this study, the costs of holding, routing, and penalties for violating the time window were minimized. Therefore, an integrated model and a non-integrated model were presented. An improved version of WOA is proposed to optimize the studied problem. Comparison of the integrated model and the two-stage model showed that the integrated model saved 23.65% of delay time, the integrated model showed better performance than the two-stage model by 13.6% in terms of total costs.

An examination and comparison of the applied solutions showed that in all solved problems, the WOA's solution time was less than the GA's and IWOA’s. However, the proposed IWOA leads to about 60% improvement in cost reduction in comparison to WOA. This is while WOA performs better than GA in 80% of large-scale problems, and GA only provides a better objective function value in 20% of the problems (problems 13 and 14). Therefore, the efficiency of the IWOA, compared to the WOA and GA, can be well approved.

Analysis of the results on the algorithms studied in this study shows that new meta-heuristic algorithms such as WOA can perform much better and more powerfully than conventional and old algorithms and provide higher speed and higher quality, and this algorithm Have the ability to replace the old algorithms well. Also, due to the newness of the WOA algorithm, various improvements can be made in it, which can implement the search process in the WOA algorithm better and more powerfully.

In this study, IWOA algorithm have led to improvements in both the neighborhood creation structure and the choice of answers for subsequent iterations, resulting in a 60% improvement in cost savings over the WOA. Therefore, it is generally concluded that the IWOA algorithm can be introduced as a new and efficient algorithm both in terms of optimization speed and quality of the results found, and other researchers in this field are suggested to focus more on Rely on this algorithm and use its advantages over other meta-heuristic algorithms.

The management-related achievements of this research show that in factories, integration of decisions related to production, assembly and distribution can help managers in controlling costs and creating coordination between production and distribution units. The need for this integration will be intensified when there are multiple customer orders. In such a situation, rough-cut planning cannot provide the necessary coordination among production, assembly, and distribution, and it is necessary to use up-to-date scientific tools. This research can be a comprehensive decision making for managers of manufacturing organizations. The limitations of the research are as follows:

I. The meta-heuristics algorithms require access to a computer system equipped with features such as high RAM and CPU.
II. As there was no official database for some parts of cost elements, the driver's estimations were asked to help. The questions about the travel costs for each route have been categorized and the estimated costs have been entered into the mathematical model.

To develop this research, it is suggested that uncertainty in customer demand be considered as possibilistic programming or robust optimization. It is also recommended to include planning on product waste and supply of raw materials in the problem. Regarding the solution methods, it is suggested that a heuristic algorithm be developed for the problem, and also other meta-heuristic algorithms such as Runner-Root Algorithm (RRA) be used to solve this problem.

Acknowledgments

The authors are tankful from Deputy for Research and Technology of Islamic Azad University, Qazvin Branch with the code 21440311962004.

Funding

This study is extracted from the Ph.D dissertation in which is supported by Islamic Azad University, Qazvin Branch with the code 21440311962004.

Conflicts of Interest

There is no financial interest to report.

Reference

[1] Negri, M., Cagno, E., Colicchia, C., & Sarkis, J. (2021). Integrating sustainability and resilience in the supply chain: a systematic literature review and a research agenda. *Business strategy and the environment*, *30*(7), 2858-2886.

[2] Burggräf, P., Wagner, J., Lück, K., & Adlon, T. (2017). Cost-benefit analysis for disruption prevention in low-volume assembly. *Production engineering*, *11*, 331-342.

[3] Mohammadi, S., Al-e-Hashem, S. M., & Rekik, Y. (2020). An integrated production scheduling and delivery route planning with multi-purpose machines: a case study from a furniture manufacturing company. *International journal of production economics*, *219*, 347-359.

[4] Wu, C. C., Gupta, J. N., Cheng, S. R., Lin, B. M., Yip, S. H., & Lin, W. C. (2021). Robust scheduling for a two-stage assembly shop with scenario-dependent processing times. *International journal of production research*, *59*(17), 5372-5387.

[5] Wu, C. C., Zhang, X., Azzouz, A., Shen, W. L., Cheng, S. R., Hsu, P. H., & Lin, W. C. (2021). Metaheuristics for two-stage flow-shop assembly problem with a truncation learning function. *Engineering optimization*, *53*(5), 843-866.

[6] Abdolazimi, O., Shishebori, D., Goodarzian, F., Ghasemi, P., & Appolloni, A. (2021). Designing a new mathematical model based on ABC analysis for inventory control problem: a real case study. *RAIRO-operations research*, *55*(4), 2309-2335.

[7] Khanchehzarrin, S., Shahmizad, M., Mahdavi, I., Mahdavi-Amiri, N., & Ghasemi, P. (2021). A model for the time dependent vehicle routing problem with time windows under traffic conditions with intelligent travel times. *RAIRO-Operations research*, *55*(4), 2203-2222.

[8] Potts, C. N., Sevast'Janov, S. V., Strusevich, V. A., Van Wassenhove, L. N., & Zwaneveld, C. M. (1995). The two-stage assembly scheduling problem: complexity and approximation. *Operations research*, *43*(2), 346-355.

[9] Lee, C. Y., Cheng, T. C. E., & Lin, B. M. T. (1993). Minimizing the makespan in the 3-machine assembly-type flowshop scheduling problem. *Management science*, *39*(5), 616-625.

[10] Allahverdi, A., & Al-Anzi, F. S. (2006). Evolutionary heuristics and an algorithm for the two-stage assembly scheduling problem to minimize makespan with setup times. *International journal of production research*, *44*(22), 4713-4735.

[11] Tozkapan, A., Kırca, Ö., & Chung, C. S. (2003). A branch and bound algorithm to minimize the total weighted flowtime for the two-stage assembly scheduling problem. *Computers & operations research*, *30*(2), 309-320.

[12] Allahverdi, A., & Aydilek, H. (2015). The two stage assembly flowshop scheduling problem to minimize total tardiness. *Journal of intelligent manufacturing*, *26*, 225-237.

[13] Sung, C. S., & Kim, H. A. (2008). A two-stage multiple-machine assembly scheduling problem for minimizing sum of completion times. *International journal of production economics*, *113*(2), 1038-1048.

[14] Koulamas, C., & Kyparisis, G. J. (2001). The three-stage assembly flowshop scheduling problem. *Computers & operations research*, *28*(7), 689-704.

[15] Zhang, Z., & Tang, Q. (2021). Integrating flexible preventive maintenance activities into two-stage assembly flow shop scheduling with multiple assembly machines. *Computers & industrial engineering*, *159*, 107493. https://doi.org/10.1016/j.cie.2021.107493

[16] Pourhejazy, P., Cheng, C. Y., Ying, K. C., & Lin, S. Y. (2021). Supply chain-oriented two-stage assembly flowshops with sequence-dependent setup times. *Journal of manufacturing systems*, *61*, 139-154.

[17] Hatami, S., Ebrahimnejad, S., Tavakkoli-Moghaddam, R., & Maboudian, Y. (2010). Two meta-heuristics for three-stage assembly flowshop scheduling with sequence-dependent setup times. *International journal of advanced manufacturing technology*, *50*(9-12), 1153-1164.

[18] Andrés, C., & Hatami, S. (2011, September). The three stage assembly permutation flowshop scheduling problem. *V international conference on industrial engineering and industrial management* (pp. 867-875). http://adingor.es/congresos/web/uploads/cio/cio2011/metodos_cuantitativos/867-875.pdf

[19] Maleki-Darounkolaei, A., Modiri, M., Tavakkoli-Moghaddam, R., & Seyyedi, I. (2012). A three-stage assembly flow shop scheduling problem with blocking and sequence-dependent set up times. *Journal of industrial engineering international*, *8*, 1-7.

[20] Dalfard, V. M., Ardakani, A., & Banihashemi, T. N. (2011). Hybrid genetic algorithm for assembly flow-shop scheduling problem with sequence-dependent setup and transportation times. *Technical gazette*, *18*(4), 497-504.

[21] Mozdgir, A., Fatemi Ghomi, S. M. T., Jolai, F., & Navaei, J. (2013). Two-stage assembly flow-shop scheduling problem with non-identical assembly machines considering setup times. *International journal of production research*, *51*(12), 3625-3642.

[22] Tian, Y., Liu, D., Yuan, D., & Wang, K. (2013). A discrete PSO for two-stage assembly scheduling problem. *The international journal of advanced manufacturing technology*, *66*, 481-499.

[23] Allahverdi, A., Aydilek, H., & Aydilek, A. (2016). Two-stage assembly scheduling problem for minimizing total tardiness with setup times. *Applied mathematical modelling*, *40*(17-18), 7796-7815.

[24] Navaei, J., Ghomi, S. F., Jolai, F., Shiraqai, M. E., & Hidaji, H. (2013). Two-stage flow-shop scheduling problem with non-identical second stage assembly machines. *The international journal of advanced manufacturing technology*, *69*, 2215-2226.

[25] Shoaardebili, N., & Fattahi, P. (2015). Multi-objective meta-heuristics to solve three-stage assembly flow shop scheduling problem with machine availability constraints. *International journal of production research*, *53*(3), 944-968.

[26] Komaki, G. M., & Kayvanfar, V. (2015). Grey Wolf Optimizer algorithm for the two-stage assembly flow shop scheduling problem with release time. *Journal of computational science*, *8*, 109-120.

[27] Wang, K., Ma, W. Q., Luo, H., & Qin, H. (2016). Coordinated scheduling of production and transportation in a two-stage assembly flowshop. *International journal of production research*, *54*(22), 6891-6911.

[28] Kazemi, H., Mazdeh, M. M., & Rostami, M. (2017). The two stage assembly flow-shop scheduling problem with batching and delivery. *Engineering applications of artificial intelligence*, *63*, 98-107.

[29] Jung, S., Woo, Y. B., & Kim, B. S. (2017). Two-stage assembly scheduling problem for processing products with dynamic component-sizes and a setup time. *Computers & industrial engineering*, *104*, 98-113.

[30] Wu, C. C., Chen, J. Y., Lin, W. C., Lai, K., Liu, S. C., & Yu, P. W. (2018). A two-stage three-machine assembly flow shop scheduling with learning consideration to minimize the flowtime by six hybrids of particle swarm optimization. *Swarm and evolutionary computation*, *41*, 97-110.

[31] Goli, A., & Davoodi, S. M. R. (2018). Coordination policy for production and delivery scheduling in the closed loop supply chain. *Production engineering*, *12*(5), 621-631.

[32] Basir, S. A., Mazdeh, M. M., & Namakshenas, M. (2018). Bi-level genetic algorithms for a two-stage assembly flow-shop scheduling problem with batch delivery system. *Computers & industrial engineering*, *126*, 217-231.

[33] Yavari, M., & Isvandi, S. (2018). Integrated decision making for parts ordering and scheduling of jobs on two-stage assembly problem in three level supply chain. *Journal of manufacturing systems*, *46*, 137-151.

[34] Luo, J., Liu, Z., & Xing, K. (2019). Hybrid branch and bound algorithms for the two-stage assembly scheduling problem with separated setup times. *International journal of production research*, *57*(5), 1398-1412.

[35] Talens, C., Fernandez-Viagas, V., Perez-Gonzalez, P., & Framinan, J. M. (2020). New efficient constructive heuristics for the two-stage multi-machine assembly scheduling problem. *Computers & industrial engineering*, *140*, 106223. https://doi.org/10.1016/j.cie.2019.106223

[36] Lin, X., & Chen, J. (2021). Deep reinforcement learning for dynamic scheduling of two-stage assembly flowshop. *Advances in swarm intelligence: international conference on Swarm Intelligence, ICSI 2021* (pp. 263-271). Springer International Publishing. https://doi.org/10.1007/978-3-030-78811-7_25

[37] Hosseini, S. M. H., Sana, S. S., & Rostami, M. (2022). Assembly flow shop scheduling problem considering machine eligibility restrictions and auxiliary resource constraints. *International journal of systems science: operations & logistics*, *9*(4), 512-528.

[38] Masruroh, N. A., Fauziah, H. A., & Sulistyo, S. R. (2020). Integrated production scheduling and distribution allocation for multi-products considering sequence-dependent setups: a practical application. *Production engineering*, *14*, 191-206.

[39] Mirjalili, S., & Lewis, A. (2016). The whale optimization algorithm. *Advances in engineering software*, *95*, 51-67.

[40] Alinaghian, M., & Goli, A. (2017). Location, allocation and routing of temporary health centers in rural areas in crisis, solved by improved harmony search algorithm. *International journal of computational intelligence systems*, *10*(1), 894-913.

[41] Tirkolaee, E. B., Goli, A., Hematian, M., Sangaiah, A. K., & Han, T. (2019). Multi-objective multi-mode resource constrained project scheduling problem using Pareto-based algorithms. *Computing*, *101*, 547-570.

[42] Rashidi Komijan, A., Ghasemi, P., Khalili-Damghani, K., & HashemiYazdi, F. (2021). A new school bus routing problem considering gender separation, special students and mix loading: a genetic algorithm approach. *Journal of optimization in industrial engineering*, *14*(2), 23-39.