Scientia Iranica E (2018) 25(2), 911–930

Sharif University of Technology
Scientia Iranica
Transactions E: Industrial Engineering
http://scientiairanica.sharif.edu

A joint scheduling of production and distribution operations in a flow shop manufacturing system

S. Mohammadi, A. Cheraghalikhani, and R. Ramezanian*

Department of Industrial Engineering, K.N. Toosi University of Technology, Tehran, Iran.

Received 29 May 2016; received in revised form 3 October 2016; accepted 18 December 2016

KEYWORDS
Integrated modeling approach;
Flow shop scheduling;
distribution with routing;
Imperialist competitive algorithm.

Abstract. In traditional scheduling problems and many real-world applications, the production operations are scheduled regardless of distribution decisions. Indeed, the completion time of a job in such problems is defined traditionally as the time when the production sequences of a job are finished. However, in many practical environments, completed orders are delivered to customers immediately after production stages without any further inventory storage. Therefore, in this paper, we investigate an integrated scheduling model of production and distribution problems simultaneously. It is assumed that products proceed through a permutation flow shop scheduling manufacturing system and are delivered to customers via available vehicles. The objective of our integrated model is to minimize the Maximum Returning Time (MRT), which is the time it takes for the last vehicle to deliver the last order to a relevant customer and return to production center. The problem is formulated mathematically, and then an Improved Imperialist Competitive Algorithm (I-ICA) is proposed for solving it. Furthermore, a sufficient number of test problems are generated for computational study. Various parameters of the algorithm are analyzed to calibrate the algorithm by means of the Taguchi method. At the end, the effectiveness of the proposed model and suggested algorithm is evaluated through a computational study where the obtained results show the appropriate performance of the integrated model and solving approach in comparison to those of the other algorithms.

1. Introduction

Due to market globalization, emphasizing the coordination and integration among various functions of a supply chain has become one of the vital strategies of modern manufacturers to gain competitive advantages. The supply chain management has become one of the most important and widely discussed topics in manufacturing research area over the last years [1]. Most of traditional scheduling literature reviews consider job sequencing regardless of the delivery considerations or by assuming delivery after production. Nevertheless, it is no longer suitable, and so the models should be extended with transportation considerations to cope with the distribution issue after production. There are only a few researchers that have considered the joint optimization of machine scheduling and job delivery in the integrated production and delivery scheduling; moreover, after a manufacturing company receives orders from customers, the company processes these orders on the machines, and then delivers the finished orders to the customers. This means that the finished products are delivered immediately to the relevant cus-

*. *Corresponding author. Tel.: +98 21 84063365; Fax: +98 21 88674858*
E-mail addresses: mohammadi@kntu.ac.ir (S. Mohammadi); Ali_Cheraghalikhani@mail.kntu.ac.ir (A. Cheraghalikhani); Ramezanian@kntu.ac.ir (R. Ramezanian)

doi: 10.24200/sci.2017.4437

tomers. Considering the importance and application of flow shop systems in production systems on one the hand and the significance of transportation issues, especially vehicle routing decisions on the other hand, the aim of this paper is the employment of integrated approach to production and distribution in a flow-shop manufacturing system.

In this paper, a new integrated model is developed for a flow shop and delivery-scheduling problem. In the proposed model, a Permutation Flow Shop Scheduling Problem (PFSSP) is assumed for production system. Indeed, in the integrated model, not only starting and finishing times of each job are determined on each machine, but also route of vehicles is determined for achieving optimum MRT through both production and delivery stages simultaneously. In addition, Improved meta-heuristic, Imperialist Competitive Algorithm (I-ICA) was proposed to solve the constructed model. To improve the performance of ICA, some effective policies are added to explore and exploit the solution space in assimilation and revolution steps of ICA and the resulting algorithm, called I-ICA, in this paper. To evaluate the performance of developed model and algorithm, we used three well-known datasets of flow shop scheduling problem in literature to generate suitable test problems. Next, to enhance the performance of the ICA, the algorithm parameters are adapted by pre-experimental planning. On the other side, in order to investigate the value of integration by the developed model, three benchmarks are used as the data for production scheduling part of integrated model, and the distribution with routing data is generated randomly for each of production benchmarks. In order to evaluate the value of integration by the developed model, MRT is used as a performance measure which is defined as the time it takes for a vehicle to deliver the last order to a relevant customer and return to production center. Based on MRT comparison, effectiveness of the developed model is investigated where the obtained results show the adequate performance of the integrated model. At the end, the statistical analysis results show acceptable performance of the proposed model rather than non-integrated approach. Indeed, we evaluate the value of integration by comparing the proposed integrated approach with the hierarchical approach.

The reminder of the paper is structured as follows. Section 2 introduces the problem and develops a mathematical model. Extensive study of the related literature review is done in Section 3. Section 4 presents an improved solution algorithm based on ICA. The computational study and simulation results are provided in Section 5. Managerial insight, which includes examples of real-world application of the proposed model and its sensitivity analysis, is presented in Section 6. Section 7 contains the conclusions and some directions for future research.

2. Literature review

The integrated production and delivery scheduling was first studied by Potts [2]. Potts [2] and Hall and Shmoys [3] studied a single-machine scheduling problem with non-identical delivery and release times, where there are enough number of vehicles to immediately deliver products to a customer. Afterwards, Matsuo [4] considered a similar problem in the single-machine environment in order to couple production with delivery decisions. Herrmann and Lee [5], Yuan [6], Chen [7], and Cheng et al. [8] evaluated that products are delivered after being processed by infinite delivery capacity via enough number of vehicles. Yang [9] and Hall et al. [10] considered some additional assumptions of such a problem. Yang [9] proposed a model similar to the one studied by Cheng et al. [8], yet with given delivery dates. Hall and Potts [11] suggested a variety of scheduling and delivery problems within a supply chain and minimized the overall scheduling and delivery costs. Transportation capacity is not considered in their model. Moreover, they assumed that each vehicle would be shipped to only one downstream destination. Lee and Chen [12] investigated other machine scheduling models by imposing constraints on both transportation capacity and transportation time. Hall and Potts [13] studied a variety of single-machine scheduling problems with the objective of minimizing the overall scheduling and delivery costs. One of the notable characteristics in their model is that there are always transporters available for delivering. Chang and Lee [1] further studied this problem where each job requires different physical spaces for delivery. Garcia et al. [14] dealt with several plants, each equipped with parallel machines and a fleet of vehicles that can deliver no more than one job at a time. Li et al. [15] developed a single-machine scheduling model that incorporates routing decisions of a delivery vehicle, which serves customers at different locations. The objective is to minimize the sum of job arrival times. Meanwhile, they developed a polynomial time algorithm for the case when the number of customers is fixed. He et al. [16], Zhong et al. [17], and Su et al. [18] revisited the problem of Chang and Lee [1] and improved algorithms. Li and Vairaktarakis [19] focused on a similar problem with jobs that consist of two jobs. There is a special machine for each job. Wang and Cheng [20] introduced the machine availability constraint into Lee and Chens' [12] model. Zhong et al. [17] presented some improved approximation results for the problems considered by Chang and Lee [1]. Chen and Lee [21] considered a more general two-stage scheduling problem where, in the second stage of transportation, there are multiple (more than two) transportation modes to select multiple destinations. Lu et al. [22] considered the single-machine scheduling with release dates and job delivery.

Liu and Lu [23] provided an improved approximation algorithm for the problem studied by Lu et al. [22].

Woeginger [24] considered a scheduling problem on parallel machines without considering release dates and proposed a heuristic method with worst-case analysis. Wang and Cheng [25] studied a parallel machine scheduling to integrate scheduling with delivery. In another study, Hall and Potts [26] analyzed the complexity of scheduling problems for many different objectives on single and parallel machine(s) with delivery coordination. Chen and Vairaktarakis [27] extended the problem to more destinations and an infinite number of vehicles with a threshold on loading capacity limited to a maximum number of jobs. Single as well as parallel machine scheduling problems are addressed in their research. In study of Su et al. [18], jobs are processed by two parallel machines and delivered to a customer with the objective of minimizing the makespan. Ullrich [28] integrated the production and outbound distribution scheduling in order to minimize total tardiness. The overall problem consists of two sub problems. The first addresses scheduling of a set of jobs on parallel machines with machine-dependent ready times, while the second focuses on the delivery of completed jobs with a fleet of vehicles with different loading capacities and ready times. Job-dependent processing times, delivery time windows, service times, and destinations were also taken into account in a study done by Ullrich [28]. Wan and Zhang [29] investigated a coordinated scheduling problem on parallel identical machines with job delivery. The objective is to minimize the sum of job arrival times. Moreover, Lee [30] proposed a coordinated scheduling of production and delivery issues, where jobs are processed on a single machine without preemption and then delivered to a customer. It was assumed that the delivery cost depends on time period for delivery, but not dependent on individual jobs. Lu et al. [31] integrated production and delivery scheduling on a serial batch machine to minimize the makespan. Two-machine flow shop scheduling with transportation constraints was suggested by Yuan et al. [32] to minimize makespan. Pan et al. [33] focused on a class of two-machine flow shop problem in which jobs need to be delivered to customers by vehicles after production stages. Finally, it should be noted that the complexity of integrated models with different assumptions was studied by many researchers in literature (e.g. [1,15,29,30,34]).

As can be seen, all the reviewed studies have just investigated the concept of integration of production and delivery of finished products to customers in single-machine, parallel machine(s), and two-machine flow shop environments. To the best of our knowledge, there is no research in literature proposing M machines permutation flow shop in integrated production and routing delivery model with vehicles. It seems that the high difficulty of such an assumption is one of the main reasons, while PFSSP is a well-known class of scheduling problem with many applications in industrial and manufacturing systems rather than single and parallel machines that have been extensively studied within recent years.

3. Problem formulation

In this work, we developed an integration model of the PFSSP and distribution with routing. In the developed model, not only starting and completion times of orders in each production machines should be calculated, but also the time of each vehicle departing from center, the rout of each vehicle and time of receiving orders to relevant customers, and total completion time for each order are considered as decision variables. Therefore, it seems that the proposed model is more complicated than traditional PFSSP and belongs to the class of NP-hard problems.

Problem parameters:

N	Number of jobs (customers order)
M	Number of machines
V	Number of vehicles
J_i	Job number i
O_{im}	mth operation of J_i
U	Very large positive number
p_{im}	Processing time of O_{im}
f_{ijv}	Travel time between customers i and j by vehicle v
q_i	Units of vehicles capacity needed to carry J_i that is identical for all vehicles
Q_v	Capacity of vehicle v
s_j	Service time at the Jth customer

Decision variables:

t_{im}	Starting time of O_{im}
Z_{ij}	A binary variable that indicates if J_i precedes J_j
y_{iv}	A binary variable that indicates if job J_i is delivered by vehicle v
x_{ijv}	A binary variable that indicates if arc (i, j) is part of a vehicle route v
b_{iv}	Starting time of service to customer i in customer location
ST_v	Starting time of vehicle V route
yy_{ijv}	Auxiliary variable that is equal to $y_{iv} * y_{jv}$
$t_{\max}$	Time when final vehicle returns to production center

By the above notations, the proposed model is as follows (It should be noted that $i = 0$ or $j = 0$ in the proposed model's formulation refers to shop center):

$$\text{Minimise MRT} \qquad (1)$$

Subject to (1-14) :

$$t_{im}+p_{im} \leq t_{jm}+U(1-Z_{ij})t_{jm}+p_{jm} \leq t_{im}+UZ_{ij}$$

$$j \neq j, \quad i,j=1,2,...,N, \quad m=1,2,...,M, \qquad (1\text{-}1)$$

$$t_{im}+p_{im} \leq t_{i(m+1)}$$

$$i=1,2,...,N, \quad m=1,2,...,M-1, \qquad (1\text{-}2)$$

$$\sum_{\nu=1}^{V} y_{i\nu}=1, \qquad i=1,2,...,N, \qquad (1\text{-}3)$$

$$\sum_{j=1}^{N} x_{0j\nu}=1, \qquad \nu=1,2,...,V, \qquad (1\text{-}4)$$

$$y_{i\nu} \leq \sum_{j=0}^{N} x_{ji\nu}, \qquad i=1,2,...,N, \quad \nu=1,2,...,V, \qquad (1\text{-}5)$$

$$x_{ij\nu} \leq y_{i\nu} * y_{j\nu},$$

$$i \neq j, \quad i,j=1,2,...,N, \quad \nu=1,2,...,V, \qquad (1\text{-}6)$$

$$\sum_{i=0}^{N} x_{ip\nu} - \sum_{j=0}^{N} x_{pj\nu}=0,$$

$$p=0,1,2,...,N, \quad \nu=1,2,...,V, \qquad (1\text{-}7)$$

$$\sum_{i=1}^{N} q_i y_{i\nu} \leq Q_\nu, \qquad \nu=1,2,...,V, \qquad (1\text{-}8)$$

$$ST_\nu \geq y_{i\nu}(t_{iM}+p_{iM}),$$

$$i=1,2,...,N, \quad \nu=1,2,...,V, \qquad (1\text{-}9)$$

$$b_{i\nu} \geq y_{i\nu} * ST_\nu, \quad i=1,2,...,N, \quad \nu=1,2,...,V, \qquad (1\text{-}10)$$

$$b_{0\nu} \geq ST_\nu * \sum_{j=1}^{N} x_{0j\nu}, \qquad \nu=1,2,...,V, \qquad (1\text{-}11)$$

$$b_{i\nu}+s_i+f_{ij\nu} \leq b_{j\nu} * x_{ij\nu},$$

$$i=0,1,2,...,N, \quad j=1,2,...,N, \quad \nu=1,2,...,V, \qquad (1\text{-}12)$$

$$\text{MRT} \geq x_{i0\nu}(b_{i\nu}+s_i+f_{i0\nu}),$$

$$i=1,2,...,N, \quad \nu=1,2,...,V, \qquad (1\text{-}13)$$

$$t_{im}, b_{i\nu}, \text{MRT} \geq 0, \qquad \forall i,m,\nu,$$

$$Z_{ij}, y_{i\nu}, x_{ij\nu}, yy_{ij\nu}=\{0,1\}, \qquad \forall i,j,\nu. \qquad (1\text{-}14)$$

The objective function minimizes MRT. It should be noted that $i=0$ or $j=0$ in the proposed model's formulation refers to shop center.

3.1. Production scheduling constraints

Constraint (1-1) ensures that only one job could be processed on a machine at any time; if J_i precedes J_j, then $t_{im}+p_{im} \leq t_{jm}$; if J_j proceeds J_i, $t_{jm}+p_{jm} \leq t_{im}$ for all operations of jobs on each machine. Constraint (1-2) ensures that $O_{j(m+1)}$ begins after completion of O_{jm} on each machine.

3.2. Distribution constraints

Constraint (1-3) ensures that each job, J_i, is delivered only with one vehicle. Constraints (1-4) guarantee that each vehicle must depart from production center to only one customer in its route; indeed, each vehicle can be used once. Constraint (1-5) enforces that each customer is serviced by only one vehicle. Constraint (1-6) guarantees that each vehicle can cross paths between two customers (i and j) if that vehicle is allocated to both customers (i and j). Constraint (1-7) allows that, for each costumer, p, the entering vehicle must eventually leave this node. Constraint (1-8) ensures the sufficient capacity of vehicles.

3.3. Integrated constraints

Constraint (1-9) implies that vehicle starts routing after completion of all orders allocated to the vehicle in production center. Constraint (1-10) ensures that service time of each customer is after departure time of relevant vehicle. Constraint (1-11) implies the departure time of vehicles from production center. Constraint (1-12) implies that service time of each customer after the completion of previous customer service is the routing of allocated vehicle. In addition, Constraint (1-12) acts as a sub tour elimination constraint in this model. Constraint (1-13) is defined as the ending time of each route. The constraint in Eq. (1-13) ensures that the ending time at the delivery stage can be completed only after all vehicles return to the production center (or MRT that can be defined as maximum time of vehicles tour).

The obtained model is solved for small-sized instances using optimization software GAMS 23.8 and solver CPLEX 12.4. For medium- and large-sized instances, GAMS is a diagnosed inappropriate solution method for solving the proposed model due to non-acceptable solving time, thus meta-heuristics must be used for solving the proposed model. Finally, due to the advantages of ICA, ICA has been selected to solve the proposed integrated model.

4. Imperialist competitive algorithm

4.1. *Imperialist competitive algorithm in general*

Imperialist Competitive Algorithm (ICA) is a kind of meta-heuristic algorithm used for solving different optimization problems. This algorithm, which was first proposed by Atashpaz and Lucas [35], is a population-based meta-heuristic inspired from socio-political behaviors. ICA begins with a random initial population. Each individual of the population is named a "country". In the first step of the algorithm, some of the best countries are selected as "imperialist countries", and the remaining countries are considered to be "colonies" of the imperialists.

Colonies are then divided among those imperialists based on their power. After dividing all colonies among the imperialist countries, the colonies move toward their relevant imperialists within the cultural state space in each iteration of algorithm. A collection of an imperialist country and some colonies is named an empire, competing among empires and displacing each other. The survival of an empire depends on its power to take over colonies from other competitors. The power of larger empires increases, while the empires of less power collapse in iterations. Following many iterations of competition among empires, the power of colonies approaches that of the imperialist countries, indicating convergence. The extreme bound of imperialist competition occurs when only one imperialist remains along with some colonies, which are close to the imperialist country in terms of position. Pseudo code of the ICA algorithm is shown in Figure 1.

There have been some reports on the applications of imperialist competitive algorithm in solving the scheduling problems by researchers [36-42]. Thus, based on proper reports of ICA for combinatorial and complicated problems, such as the developed model and appropriate performance of this algorithm, the authors have selected ICA for solving method. A review on the mentioned studies showed that this algorithm has not been used to solve permutation flow-shop problems until now. In all these researches, ICA is stated as an effective algorithm. Therefore, another reason for selecting ICA by authors is the capability of ICA to solve permutation flow-shop problem.

4.2. *The Improved Imperialist Competitive Algorithm (I-ICA)*

Due to some improvement in the general ICA for solving the proposed model, the obtained algorithm was named as improved ICA (I-ICA) in this paper. One of the important issues of solving optimization problems using meta-heuristics is determining an appropriate

```
Begin ICA
        Initialize parameters.
        Choose a termination criterion (maximum number of iterations).
        Produce the first population randomly.
        Compute the objective function for the first population.
        Descend the first population based on their objective function costs.
        Specify the imperialist states.
        Form the empires by sharing the colonies to the imperialist states.
*While (the specified decades are not passed) do
        **While (all empires selected) do
                Choose the i'th empire
                ***While (all colonies selected) do
                        Choose the j'th colony
                        move the colony toward its imperialist state (Assimilation)
                        Change the direction of movement for some colonies (Revaluation)
                        Compute the objective function cost for the populations in new positions
                        Evaluate two new costs and select the best one
                        Substitute j'th colony with new one
                End ***while
                Descend all colonies of i'th empire based on their cost functions
                Check cost functions of all colonies in each empire
                If (There is a colony with a lower cost than its imperialist)
                        Exchange the position of the colony and imperialist
                End If
                Update the location of the i'th empire
        End **While
        Compute total cost of empire
        Search for the weakest empire
        Give one its colonies to the empire which is more powerful
        Check the number of colony in each empire
        If there is an empire with no colony
                Omit the empire and possess its imperialist for the best empire
        End If
End *While
End ICA
```

Figure 1. Pseudo code of the ICA algorithm.

Sol. Part 1	5	3	2	4	1	
Sol. Part 2	5	0	3	4	0	2

Figure 2. Two-part array for solution representation.

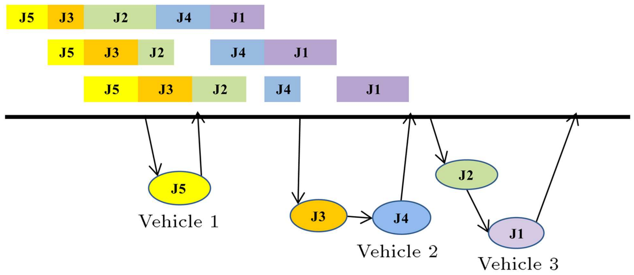


Figure 3. Example of country decoding.

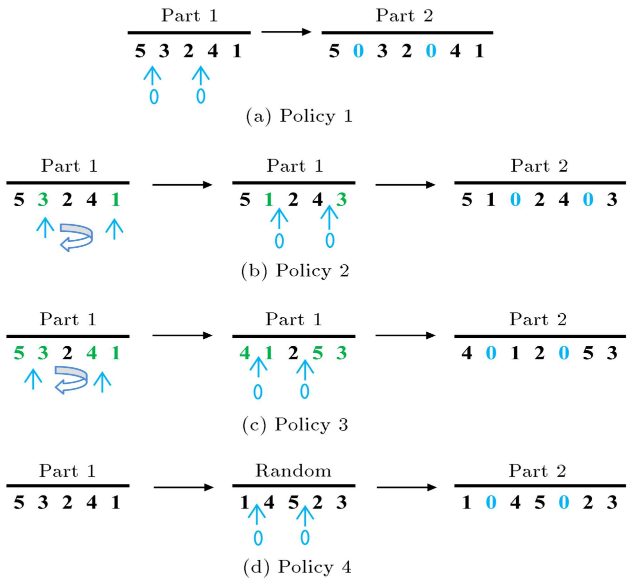


Figure 4. Policies of initial population generation.

format to represent a solution. In ICA, the term "country" is used for this array. Thus, we suggested an array with two parts of showing each solution (country). The first part refers to flow shop stage of solution, and the second part shows distribution with routing stage of solution as follows.

In the example shown in Figure 2, we have three machines in production stage, and there are five orders that must be produced in the sequence of 5-3-2-4-1 in all machines, and then delivered to their customers with three vehicles. Zeros in part two of country represent vehicle separator; for the sake of certainty, the feasible number of constraints of zero must be equal to $(V-1)$ and no two zeros placed behind each other. Figure 3 shows this example schematically.

The cost of a country is found by evaluating the cost function that defined MRT in this study:

$$\text{Cost} = f(\text{country}).$$

4.2.1. Generating initial empires

In the initialization step, we need to generate an initial population with the size of N_{pop}. Next, we have to select some of the best countries with the lowest cost of function values and the size of N_{imp} from N_{pop} and set them to be imperialists. The rest of countries are set to be colonies N_{col} $(N_{\text{col}} = N_{\text{pop}} - N_{\text{imp}})$.

Part one of country is random permutation of $1, 2, ..., N$ numbers for flow shop scheduling solution generated completely randomly. For generating part two of country, three random policies are defined and proposed to obtain part two based on part one in a random approach, and then one random policy is defined to generate parts one and two in a random approach separately:

- Policy 1: Putting $(V-1)$ zeros between part 1 randomly to separate vehicles routines;
- Policy 2: Firstly, exchanging two positions of part 1 (swap), and then putting $(V-1)$ zeros between part 1 randomly to separate vehicles routines;
- Policy 3: Using multiple swaps for part 1, and then putting $(V-1)$ zeros between part 1 randomly to separate vehicles routines;
- Policy 4: Both parts one and two of solution scheme are generated based on random permutation of $1, 2, ..., N$, and then $(V-1)$ zeros are inserted into part two randomly.

Figure 4 shows the policies for generating part 2 from part 1 of country.

Given the importance of initial solution quality in each algorithm, a certain percentage of the generated solutions used 2-opt neighbor search algorithm to enhance the quality of initial population. The 2-opt operator is used for intra-route improvement. In a 2-opt move, two non-adjacent arcs are replaced by another two new ones, and the visited order of the customers between the two arcs is reversed.

Considering initial population, the number of colonies of an empire is different based on the power of the imperialists. In other words, bigger empires have more colonies and smaller ones have less. The number of colonies for each empire is calculated by the following steps:

- Calculate the normalized cost of an imperialist $(t_{\max_n})$:
$$C_n = \text{Max}_i(t_{\max_i}) - t_{\max_n},$$
where $t_{\max_n}$ is the makespan time of the nth imperialist;
- Calculate the normalized power of each imperialist (P_n):
$$P_n = \left| \frac{C_n}{\sum_{i=1}^{N_{\text{imp}}} C_i} \right|.$$

- Calculate the number of initial colonies for the nth empire (NC_n):

$$NC_n = \text{Round}\{P_n.N_{\text{Col}}\} \qquad n = 1, 2, ..., n-1.$$

The following formula was developed for the last imperialist:

$$NC_n = N_{\text{col}} - (NC_1 + NC_2 + ... + NC_{n-1}).$$

4.2.2. Assimilation: Moving the colonies of an empire toward the imperialist

When empires are formed, imperialist states try to be more powerful by absorbing the other colonies. It means that colony moves toward the imperialist state in a random direction. In this step, also, three different policies are defined to assimilate colonies.

Policy 1: Putting two positions of parts 1 and 2 of an imperialist state identically in an assimilated array, bringing the rest from colony, and thus forming an assimilated colony (see Figure 5);

Policy 2: Putting 1 subsequence of parts 1 and 2 of an imperialist state identically in an assimilated array, bringing the rest from colony, and thus forming an assimilated colony. The size of subsequence is linked to country size (see Figure 6);

Policy 3: Using Policy 1 for part 1 and Policy 2 for part 2 separately (see Figure 7).

4.2.3. Revolution

According to the imperial history, the imperialist absorbs colonies of an empire in terms of social, cultural, economic, and political characteristics; however, some colonies might resist being absorbed by imperialists. In fact, those colonies perform some sort of reformations in their characteristics. In ICA, this operation is called revolution. Revolution brings sudden random changes in the position of some colonies in the search space. Revolution operation in ICA increases exploration and prevents the early convergence of countries to local optima. By performing revolution operation, a colony suddenly jumps from one point to another random

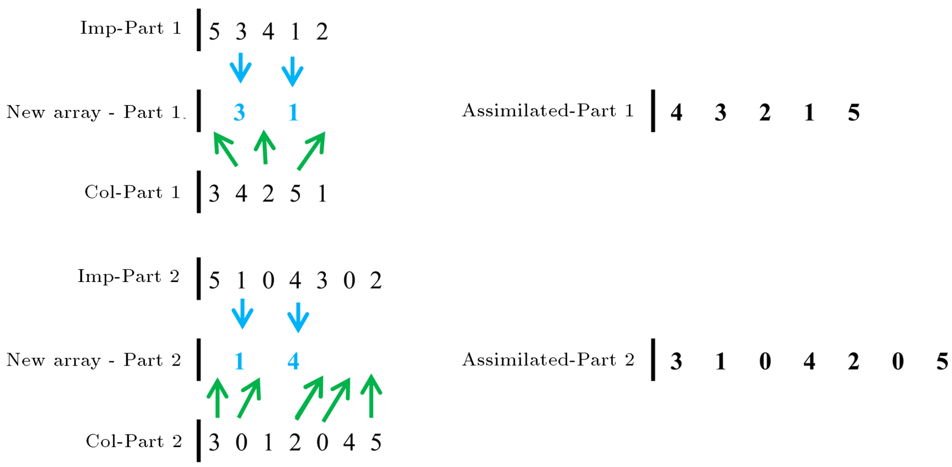


Figure 5. Policy 1 of assimilation.

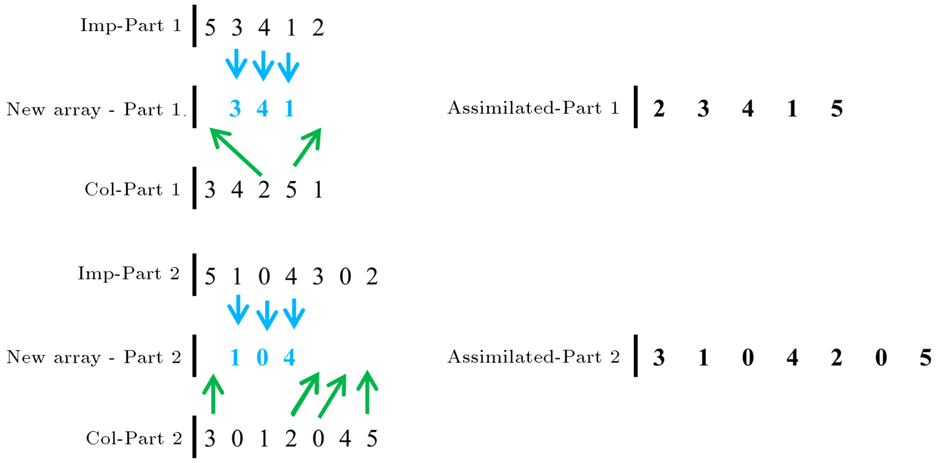


Figure 6. Policy 2 of assimilation.

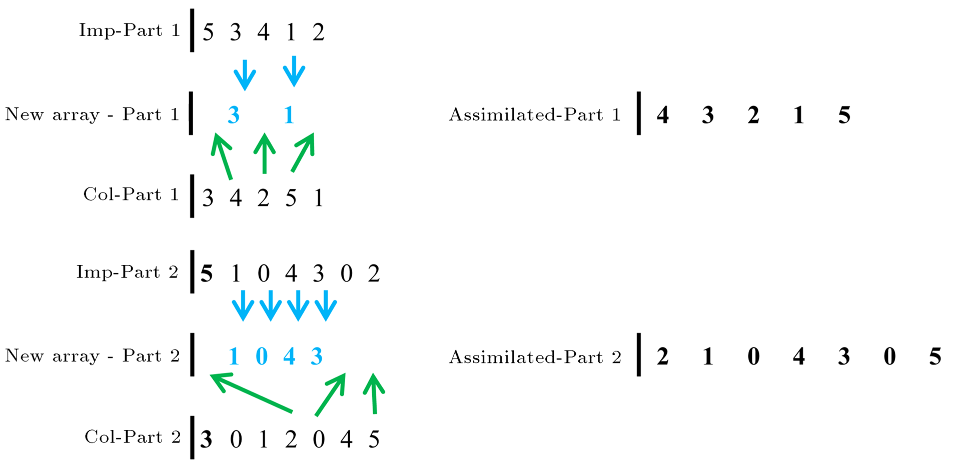
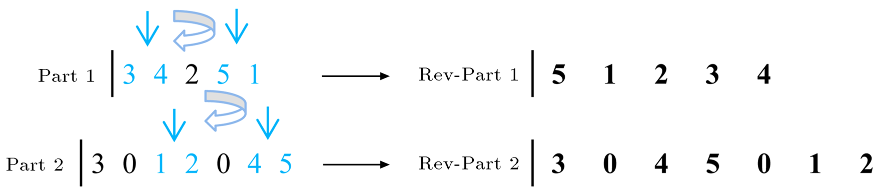
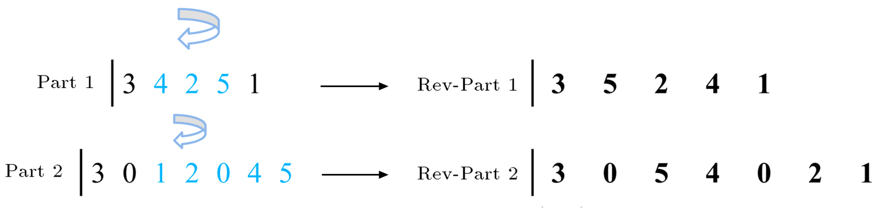


Figure 7. Policy 3 of assimilation.



(a) Swap revolution method



(b) Reversion revolution method

Figure 8. Revolution methods.

point which results in escaping from local optimum. In the proposed I-ICA for this step, three stage strategies are used. In the first strategy, only part one is changed by swap and reversion methods; in the second strategy, only part two is changed by 2-opt method, and finally, in the third strategy, parts one and two are changed simultaneously to revolutionize colonies. Choosing these three strategies is done randomly.

Strategy 1:

Swap procedure: In swap procedure, two arrays of part one containing two cells for each one are selected randomly, and the position of these arrays will be changed. Figure 8(a) shows the used swap procedure. Cells {3} and {5} are selected randomly in part 1 colony, and the new colony is obtained by replacing its positions.

Reversion procedure: For reversion procedure, a subsequence array from part one is selected randomly. The position of the chosen cells is reversed, but the position of the array is unaffected. In Figure 8(b), the used reversion method is presented. Figure 8(b) shows the used reversion procedure.

Strategy 2:

In this strategy, 2-opt operator is applied randomly to one vehicle route in part two. In a 2-opt move, two non-adjacent arcs are replaced by another two new ones, and the visited order of the customers between the two arcs is reversed. Figure 9 is an example of the above with arbitrary input;

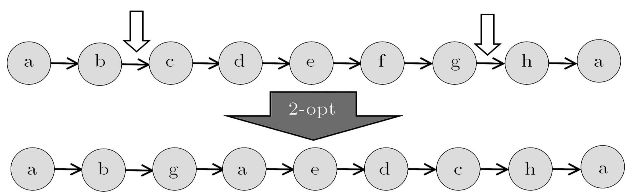


Figure 9. 2-opt strategy for revolution.

Strategy 3:

In this strategy, Strategies 1 and 2 are used for parts one and two of solution scheme, respectively.

4.2.4. Exchanging positions of the imperialist and a colony

Once assimilation and revolution operations are performed on colonies of an empire, the cost functions of colonies' new positions are then compared with those of imperialists. If we find any colony whose cost function is less than that of imperialist, then we swap imperialist with that colony.

4.2.5. Computing total power of an empire

The power of an empire is computed based on the power of its imperialist and a fraction of the power of its colonies:

$$\text{TC}_n = \text{Cost}(\text{Imp}_n) + \varphi * \text{mean}\left\{\text{cost}(\text{colonies of Imp}_n)\right\},$$

where TC_n is the total cost of the nth empire, and φ is a positive number between 0 and 1 that is one of ICA parameters. A small value of φ emphasizes a greater influence of imperialist power on determining the total power of empire, while a large value of φ indicates the influence of the mean power of colonies in determining the total power of the empire.

4.2.6. Imperialistic competition and eliminating the powerless empires

During competition among imperialist countries, weaker empires will collapse gradually. Therefore, one or some of the weakest colonies belonging to the weakest empire will be given to a different empire based on competition that occurs among all empires. Stronger empires have a greater chance of possessing the weakest colony. In order to model the competition process among the empires, we need to compute the normalized total cost of empire as in the following:

$$\text{NTC}_n = \text{Max}_i(\text{TC}_i) - \text{TC}_n,$$

where TC_n is the total cost of the nth empire, and NTC_n is the normalized total cost of the corresponding nth empire. Then, the probability of possessing a colony is computed as follows:

$$P_n = \left| \frac{\text{NTC}_n}{\sum_{i=1}^{N_{\text{imp}}} \text{NTC}_i} \right|.$$

Stopping criterion

Many stopping criteria have been defined and developed in the literature so far. For example, it can be stopped after running a number of iterations called maximum decades or, when there is just one empire left. In this paper, it is decided to specify maximum decades as the stopping criterion.

Repair procedure

In different steps of ICA due to vehicles capacity constraint, some infeasible solutions may be obtained. In this section, repair procedure was proposed for these solutions. If one solution is infeasible in part 2 of a country, it only means that the capacity of one or more vehicles is more than that of a defined vehicle. Therefore, firstly, the zeros of part 2 are eliminated and customers are allocated to vehicles in order of remaining costumers. At the end, if the zeros in a new array are less than $(V-1)$, the remaining necessary zeros are placed randomly between non-zero members of an array (see Figure 10).

5. Computational study

This section describes the computational results obtained to investigate the performance of the proposed integrated model and I-ICA algorithm.

5.1. Test problem

Two types of scrutiny are involved in the computational study. First, the performance of the proposed algorithm is evaluated, and then the value of the developed integrated model is investigated. In order to evaluate the performance of algorithm, we compare our I-ICA algorithm with other solution methods. The authors use well-known PFSSP benchmark problems in literature and compare makespan criteria to evaluate the performance of the proposed algorithm.

	Order 1	Order 2	Order 3	Order 4	Order 5
q_i	8	3	2	4	3

$Q_1 = Q_2 = Q_3 = 10$

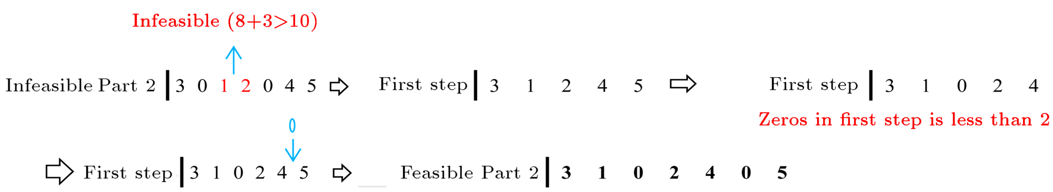


Figure 10. Example of repair procedure.

On the other side, in order to investigate the value of integration by the developed model, some test problems are required. A careful set of instances is developed randomly since the authors are unaware of any available test problems for the proposed integrated model of flow-shop scheduling with routing decision to deliver orders to customers.

For this purpose, in flow-shop scheduling literature, three different types of well-known benchmark problems are considered. The sets of the first and second problems deal with the Carlier [43] and Reeves's [44] studies which can be obtained from the OR-Library, while sets of the third problem have been taken from Taillard's [45]. The above three instances are the famous flow-shop benchmark problems that are widely used in the production scheduling literature. Since the first integrated model of PFSSP and distribution with routing is proposed in this paper, the considered benchmarks do not completely match this problem. Hence, some modifications are needed. We use these three benchmarks as the data for production scheduling part of integrated model and generate the distribution with routing data randomly for each of production benchmarks in Table 1.

In addition, the distance between customers defined with Euclidean distance and number of vehicles, (V), could be defined as in the following function:

$$V = \left\lceil \frac{\text{Sum of customers' demands}}{\text{Vehicles capacity } (Q)} \right\rceil + 1.$$

5.2. Parameter tuning

The choice of parameters' value is a challenging task in each evolutionary algorithm, since it highly influences the quality of problem solutions. This subsection studies the effects of parameters on the proposed I-ICA algorithm. For tuning the parameters, there are several designs including full factorial experiment [46,47], Taguchi experiment [48], and so on. The Taguchi method is conducted in order to decrease the required tests. Taguchi method separates the factors into two main groups: controllable factors and noise factors. It assumes that there is no control on the noise factors, but we can optimize the controllable factors. Taguchi presents a variation measure called S/N ratio (signal to noise). Herein, signal denotes the desirable value and noise denotes the undesirable value. The aim is to maximize the S/N ratio to indicate how a parameter design is a robust design. Taguchi classified the objective function into three types: larger-the-better, smaller-the-better, and on-target, minimum-variation. The proposed integrated model is classified in the second type.

The control factors of this paper are the number of the countries and imperialist, the number of iterations, the revolution probability, and the influence of mean power of colonies on total power of empire. Different levels of these factors are shown in Table 2.

The associated degree of freedom is considered to be 9; therefore, the selected orthogonal array should have a minimum of 9 rows and four columns to

Table 1. Characteristics of distribution with routing test problems.

Factors	Levels
Depot location	$x_{\text{Depot}} = y_{\text{Depot}} = \text{Uniform } (20, 50)$
Customers' location	$\begin{cases} x_{i(\text{Customer})} = \text{Uniform}(0, 2 \times x_{\text{Depot}}) \\ y_{i(\text{Customer})} = \text{Uniform}(0, 2 \times y_{\text{Depot}}) \end{cases}$
Customers' demands (d)	Uniform $(20, 50)$
Service time in customer location	Uniform $(0, 5)$
Number of vehicles (V)	$f(d)$ function based on customers' demands
Vehicles capacity (Q)	$\begin{cases} \text{Small} - \text{sized problem} & 40 \\ \text{Medium} - \text{sized problem} & 100 \\ \text{Large} - \text{sized problem} & 150 \end{cases}$

Table 2. Different levels of the factors.

Factors	Symbols	Levels
Number of the countries and imperialists	$(N_{\text{Col}}, N_{\text{Imp}})$	(50, 10), (100, 30), (150, 40)
Maximum number of iterations	Max Iter	100, 150, 200
Revolution probability	Rev percent	0.2, 0.3, 0.4
Influence of mean power of colonies on total power of empire	Zeta percent	0.1, 0.2, 0.4

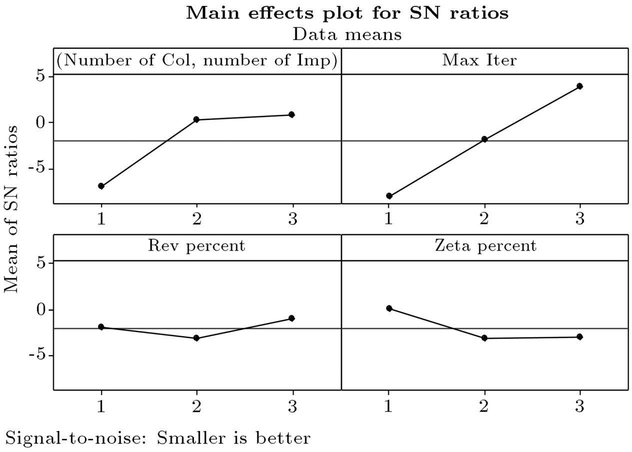


Figure 11. The S/N ratio plot for each level of factors.

accommodate the four factors. From the standard table of orthogonal arrays, L9 is selected as the fittest orthogonal array design that fulfills all our minimum requirements. Ten instances are solved for parameter setting as follows. We consider four small- and medium-sized instances from the first and second sets of problems which are randomly selected [43,44], and six medium- and large-sized instances from the third set of problems [45]. The Relative Percentage Deviation (RPD) is used as a performance measure to compare the methods. The RPD is calculated as follows:

$$\mathrm{RPD}_{i,j} = \frac{x_{i,j} - x_{\min,j}}{x_{\min,j}} * 100,$$

where $\mathrm{RPD}_{i,j}$ is the relative percentage deviation of the j'th instance of the i'th trial. $x_{i,j}$ is the objective function value that is obtained from the j'th of instance of the i'th trial. $x_{\min,j}$ is a minimum value of objective function obtained for j'th instance. When RPDs are calculated, the average of RPD of each trail is transformed to S/N ratio. According to the S/N ratio, the best level of each factor is determined. Figure 11 shows the main effects plot for S/N ratio.

The selected levels of factors are as follows: $(N_{\mathrm{Col}}, N_{\mathrm{Imp}})$ is equal to (150, 40), Max Iter is equal to 200, Rev Percent is equal to 0.3, and Zeta Percent is equal to 0.2. In order to perform the experiments, the suggested algorithm was implemented in MATLAB 7.6.0.324 (R2008a) and run on a PC with 2.80 GHz and 192 GB of RAM memory.

5.3. Comparison method

In this section, two important questions have been answered in computational study. First, does the proposed improved ICA algorithm have a suitable performance? Compared to hierarchical approach, does the integration approach improve the performance of the model? Alternatively, how much is the value of integration in the proposed model?

To answer the first question, the performance of the proposed I-ICA algorithm is compared with those of the four other algorithms in the literature (ODE, NEH, PGA, and HDDE) for solving PFSSP. ODE is a discrete differential evolution algorithm, PGA is a permutation-based genetic algorithm [49], NEH is a well-known constructive heuristic [50], and HDDE is a hybrid discrete differential evolution proposed by Mokhtari et al. [51]. For comparison, the four mentioned flow shop benchmarks are solved with improved ICA, and makespan is compared with other four algorithms solutions.

To answer the second question, the value of integration in the proposed model should be evaluated. For this purpose, distribution with routing data of integrated model is randomly generated based on each PFSSP benchmark, and then obtained benchmarks are solved by means of the proposed I-ICA algorithm; finally, hierarchical (approach) result of two models (flow shop and routing) is obtained. In order to evaluate the value of integration by the developed model, MRT is used. Based on MRT comparison, effectiveness of the proposed model is investigated where the obtained results show the good performance of integrated model. The hierarchical solution of the two models is equal to the sum of the solutions of two models separately solved by I-ICA. Then, the integrated model of flow shop and distribution with routing is solved by I-ICA, and solutions are compared with hierarchical ones. In this paper, statistical analysis using Rank-Based Friedman Test is proposed to evaluate value of integration in the proposed model.

5.4. Experimental results

The proposed I-ICA algorithm must be compared with other algorithms to evaluate the ICA ability in solving flow shop scheduling problem as well as developed integrated model.

Table 3 reports Carlier's benchmark results, while Reeves's, 20-job, and 50-job Taillard's benchmarks are shown in Tables 3-6, respectively.

The first two columns characterize the problem specifications in Table 3, while the third column represents the best-known solution reported for each problem in literature. The makespan measures, concerning ODE method PGA, NEH, and HDDE, are presented in the fourth, fifth, sixth, and seventh columns, respectively. The best value of makespan resulted from the improved ICA on Carlier problems is presented in the eighth column. The results shown in Table 3 revealed that our proposed I-ICA algorithm works better than the other four algorithms in optimizing Carliers problems. As we can see in Table 3, I-ICA can reach the best-known value of makespan in literature in all eight Carlier problems. As expected, I-ICA outperforms all other mentioned methods. Results

Table 3. Computational results on Carliers' data sets.

Problem	$N \times M$	Best known solution	ODE	PGA	NEH	HDDE	I-ICA	Hierarchical approach MRT	Integrated approach MRT
Car1	11,3	7038	7038	7038	7038	7038	**7038**	7115	7082
Car2	13,4	7166	7166	7166	7376	7166	**7166**	7250	7178
Car3	12,5	7312	7366	7399	7399	7340	**7312**	7410	7367
Car4	14,4	8003	8003	8003	8003	8003	**8003**	8090	8048
Car5	10,6	7720	7720	7720	7835	7720	**7720**	7799	7789
Car6	8,9	8505	8505	8505	8773	8505	**8505**	8620	8566
Car7	7,7	6590	6590	6590	6590	6590	**6590**	6684	6636
Car8	8,8	8366	8366	8366	8564	8366	**8366**	8467	8428
Average		7588	7594	7598	7697	7591	**7588**	7679	7637

Table 4. Computational results on Reeves' data sets.

Problem	$N \times M$	Best known solution	ODE	PGA	NEH	HDDE	I-ICA	Hierarchical approach MRT	Integrated approach MRT
Rec01	20,5	1247	1308	1283	1303	1260	1249	1371	1327
Rec03	20,5	1109	1117	1125	1132	1114	**1109**	1261	1203
Rec05	20,5	1242	1257	1245	1281	1245	1245	1377	1285
Rec07	20,10	1566	1599	1622	1626	1584	1568	1691	1660
Rec09	20,10	1537	1592	1587	1583	1582	1547	1686	1614
Rec11	20,10	1431	1494	1476	1550	1469	1445	1577	1524
Rec13	20,15	1930	2013	1975	2002	1975	1942	2062	2041
Rec15	20,15	1950	2028	2005	2013	2004	1971	2098	2076
Rec17	20,15	1902	2008	1960	2019	1956	1910	2017	2008
Rec19	30,10	2093	2230	2185	2185	2104	2104	2277	2229
Rec21	30,10	2017	2122	2113	2131	2091	2050	2177	2133
Rec23	30,10	2011	2130	2105	2155	2068	2028	2200	2188
Rec25	30,15	2513	2687	2654	2644	2631	2600	2777	2720
Rec27	30,15	2373	2503	2465	2498	2460	2413	2548	2532
Rec29	30,15	2287	2468	2459	2391	2389	2363	2551	2513
Rec31	50,10	3045	3291	3223	3173	3181	3148	3419	3365
Rec33	50,10	3114	3259	3206	3241	3177	3141	3385	3352
Rec35	50,10	3277	3356	3358	3313	3307	3280	3532	3459
Rec37	75,20	4951	5487	5299	5227	5255	5229	5515	5430
Rec39	75,20	5087	5542	5386	5308	5382	536 2	5682	5553
Rec41	75,20	4960	5506	5332	5292	5344	5291	5543	5480
Average		2459	2619	2574	2575	2551	2524	2702	2652

shown in Table 3 revealed the better performance of the integrated approach.

In order to give a better analysis, the Rank-Based Friedman Test is performed. For the statistical computations, we use the MINITAB 16 software environment. In this test, the null hypothesis (H_0) is that the two sets of results (integrated and hierarchical approaches) come from the same population, and there is no significant difference between them. The chosen alpha level for this test is equal to 0.05. Thus, H_0 is rejected, and the effectiveness of the integrated approach changes:

$$\begin{cases} H_0: & \text{Integrated approach} = \text{Hierarchical approach} \\ H_1: & \text{Otherwise} \end{cases}$$

The results in Figure 12 indicate that integrated approach has a more efficient performance than hierarchical approach does.

The second dataset investigated in this section

```
Friedman Test: Integrated Approach versus Hierarchical Approach blocked in Carlier Instances

S = 8.00  DF = 1  P = 0.005

                         Sum of
C2     N  Est Median   Ranks
H.A    8      7612.5    16.0
I.A    8      7570.3     8.0

Grand median = 7591.3
```

Figure 12. Comparison of integration and hierarchical approaches (Carlier set).

Table 5. Computational results on Taillard's benchmarks ($C_{\max}$) ($N = 20$).

Problem ($N \times M$)	**Best known solution**	**Palmer**	**CDS**	**NEH**	**HDDE**	**I-ICA**	**Hierarchical approach MRT**	**Integrated approach MRT**
20×5	1278	1384	1390	1286	1278	**1278**	1413	1347
	1359	1439	1424	1365	1360	**1359**	1484	1380
	1081	1162	1249	1159	1123	**1081**	1234	1191
	1293	1490	1418	1325	1317	**1293**	1419	1356
	1235	1360	1323	1305	1265	1243	1370	1314
	1195	1344	1312	1228	1224	**1195**	1326	1291
	1239	1400	1393	1278	1251	1251	1411	1311
	1206	1313	1341	1223	1220	**1206**	1327	1301
	1230	1426	1360	1291	1271	1231	1336	1325
	1108	1229	1164	1151	1132	**1108**	1248	1178
Average	1222	1355	1337	1261	1244	1225	1357	1299
20×10	1582	1790	1757	1680	1637	1592	1711	1683
	1659	1948	1854	1729	1711	1685	1834	1765
	1496	1729	1645	1557	1556	1511	1622	1583
	1377	1585	1547	1439	1419	1398	1535	1503
	1419	1648	1558	1502	1496	1428	1543	1494
	1397	1527	1591	1453	1446	1415	1534	1503
	1484	1735	1630	1562	1526	1486	1620	1554
	1538	1763	1788	1609	1599	1555	1691	1649
	1593	1836	1720	1647	1644	1615	1749	1696
	1591	1898	1884	1653	1637	1608	1738	1686
Average	1514	1746	1697	1583	1567	1529	1658	1612
20×20	2297	2818	2559	2410	2376	2315	2453	2440
	2099	2331	2285	2150	2142	2124	2245	2189
	2326	2678	2565	2411	2387	2347	2442	2414
	2223	2629	2434	2262	2261	2250	2398	2346
	2291	2704	2506	2397	2383	2327	2456	2414
	2226	2572	2422	2349	2280	2243	2389	2343
	2273	2456	2489	2362	2345	2290	2429	2413
	2200	2435	2362	2249	2249	2216	2362	2313
	2237	2754	2414	2320	2319	2253	2383	2357
	2178	2633	2469	2277	2270	2197	2297	2265
Average	2235	2601	2451	2319	2301	2256	2385	2349

Table 6. Computational results on Taillard's benchmarks ($C_{\max}$) ($N = 50$).

Problem ($N \times M$)	Best known solution	Palmer	CDS	NEH	HDDE	I-ICA	Hierarchical approach MRT	Integrated approach MRT
50 × 5	2724	2774	2816	2733	2729	2729	2970	2904
	2834	3041	3032	2843	2862	2848	3057	2994
	2621	2777	2703	2640	2644	2621	2902	2811
	2751	2860	2884	2782	2789	2770	2986	2908
	2863	2963	3038	2868	2864	2864	3114	3007
	2829	3090	3031	2850	2835	2832	3030	2936
	2725	2845	2969	2758	2746	2741	2982	2932
	2683	2826	2835	2721	2715	2710	2958	2855
	2552	2733	2784	2576	2578	2569	2813	2716
	2782	2915	2942	2790	2784	2783	3008	2913
Average	2736	2882	2903	2756	2755	2747	2982	2898
50 × 10	2991	3478	3421	3135	3126	3082	3333	3273
	2867	3313	3246	3032	2991	2986	3216	3114
	2839	3321	3280	2986	2989	2968	3215	3142
	3063	3511	3393	3198	3178	3142	3343	3308
	2976	3427	3375	3160	3138	3091	3351	3334
	3006	3323	3400	3178	3151	3114	3344	3266
	3093	3457	3520	3277	3237	3176	3415	3373
	3037	3356	3387	3123	3152	3120	3376	3316
	2897	3414	3251	3002	2991	2986	3226	3170
	3065	3404	3429	3257	3222	3152	3417	3411
Average	2983	3400	3370	3135	3118	3082	3324	3271
50 × 20	3771	4272	4328	4082	4067	4025	4274	4218
	3668	4303	4216	3921	3900	3893	4095	4062
	3591	4210	4189	3927	3989	3852	4118	4081
	3635	4233	4280	3969	3962	3931	4173	4094
	3553	4376	4122	3835	3868	3817	4062	3978
	3667	4312	4267	3914	3913	3905	4125	4073
	3672	4306	4134	3952	3938	3907	4171	4138
	3627	4310	4262	3938	3960	3924	4168	4137
	3645	4547	4212	3952	4014	3939	4244	4145
	3696	4197	4270	4079	4146	3907	4155	4042
Average	3653	4307	4228	3957	3976	3910	4159	4097

concerns Reeves' benchmarks in Table 4. The first column in Table 4 indicates the problem names. The second column specifies the size of each problem. The fourth to seventh columns represent the best makespan resulted from ODE, PGA, NEH, and HDDE, respectively. The 8th columns represent the best makespan obtained from the I-ICA algorithm. After comparing the I-ICA with the mentioned algorithms having seemingly better results, the distribution with routing problem is also solved by I-ICA separately. The sum of these two solutions is named as MRT for hierarchical approach of PFSSP and distribution with routing, and the results are shown in the 9th column. At the end, the proposed integrated model was

Friedman Test: Integrated Approach versus Hierarchical Approach blocked by Reeves Instances

```
S = 7.00  DF = 1  P = 0.008
                       Sum of
C10    N  Est Median   Ranks
H.A    7      2226.7    14.0
I.A    7      2174.7     7.0

Grand median = 2200.7
```

Figure 13. Comparison of integrated and hierarchical approaches (Reeves' set).

Friedman Test: Integrated Approach versus Hierarchical Approach blocked by Taillard's Instances

```
S = 6.00  DF = 1  P = 0.014

                       Sum of
C20    N  Est Median   Ranks
H.A    6      2681.2    12.0
I.A    6      2626.7     6.0

Grand median = 2653.6
```

Figure 14. Comparison of integrated and hierarchical approaches (Reeves' set).

solved by the proposed modified ICA algorithm, and MRT results are reported in 10th column of Table 4. By comparing the 9th and 10th columns' results, we can see the better performance of integrated approach clearly rather than hierarchical approach.

As we can see in Table 4, I-ICA can reach the best-known value of makespan in one Reeves' problem. As expected, I-ICA outperforms all other four mentioned methods. According to the attained results, the solutions obtained by I-ICA are better than those obtained by ODE, PGA, NEH, and HDDE are for all instances, respectively.

Meanwhile, the average MRT obtained from I-ICA for the integrated approach is only 2652, whereas the average value of MRT calculated by the hierarchical approach is 2702, respectively. The results shown in Table 4 revealed the better performance of the integrated approach. By using the mentioned Rank-Based Friedman Test, we defined each of the three problems in Revees' instance with the same size as one block, and average of these three problems is used for Freidman test. The results in Figure 13 indicate that the integrated approach, unlike the hierarchical approach, has a more efficient performance for the problem.

The third and fourth datasets investigated in this section concern Taillard's benchmarks. In Tables 5 and 6, the best attained results of the presented I-ICA on both original and modified Taillard's benchmarks are presented, and they are compared with the three well-known heuristics, namely Palmer's index (Palmer 1965), CDS (Campbell et al., 1970 [52]), and NEH (Nawaz et al., 1983 [50]) and one metaheuristic namely HDDE (Mokhtari et al., 2010 [51]). The third column represents an upper bound for each problem obtained from Taillard's website. The last two columns indicate the performances of the integrated and hierarchical approaches.

As the results presented in Tables 5 and 6 show, the I-ICA outperforms other algorithms in all sets of problems and obtains the best-known solution to seven problems (30 out of all problems). The solutions of I-ICA are better than those by the Palmer, CDS, NEH, and HDDE methods are for all instances. Furthermore, the average deviation from the upper bound obtained by the suggested methods is less than the other four algorithms. With regard to the other set of Taillard's problems shown in Tables 5 and 6, the average makespan obtained by I-ICA is less than those by other four methods are. As can be seen, the results obtained by I-ICA are equal to or better than those by the Palmer, CDS, and HDDE methods are for all the instances. By using the mentioned Rank-Based Friedman Test, we defined each of ten problems in Taillard's instance with the same size as one block, and average of these ten problems is used for Friedman test. The results in Figure 14 indicate that the integrated approach has a more efficient performance for the problem.

6. Managerial insight

Various types of wood products orders by customers who are in different locations will be sent to the production plant. These orders include different models of adult and children bed sets, a wide variety of home wood decorative, different models of TV tables, different models of shelves for shoes, etc. Several reputable companies with experience in this industry are working in Iran. Different workstations are located in manufacturing plants to arrange activities on products from raw materials to final products. Sheet cutting station, PVC covers station, etc. are embedded usually in these factories. Different orders are processed at all stations in order to be ready to be sent to the relevant customers. Thus, a flow shop system can be used to describe the mentioned production environment. After completion of the production, orders must be loaded and departed from plant to the different customers. Routing approach is used for delivery of orders by capacitated vehicles.

Production and delivery schedules are usually done separately in these factories. At certain times of the year when orders are faced with a significant increase in demand for products, the customers will encounter long delays in delivery. Based on management review, this delay reduces customer satisfaction, reduces readmissions dis-satisfied customers, and ultimately reduces profitability. Integrated model presented in this study has been proposed as a way to manage this issue for manufacturing environments. According to the mentioned production system and routing delivery method for transporting the orders by vehicles, employment of the integrated model presented in this paper could be proposed to managers as an effective and efficient solution (completion time is defined as the time when an order is delivered to the destination).

To clarify the real application of the proposed model in wood industry, consider 22 customers' zone from 22 different districts of Tehran urban area with different orders (see Figure 15). The production plant is located in one of industrial complex around of Tehran. Five stations are working in the manufacturing sector: Sheet Cutting station (SC), PVC covers station (PVC), Initial Assembly station (IA), Final Assembly station (FA), and Packaging station (P).

Customers and products data are shown in Table 7. The location of production center (X, Y axes) is (350, 350), and 5 vehicles with capacity of 100 are used to deliver orders.

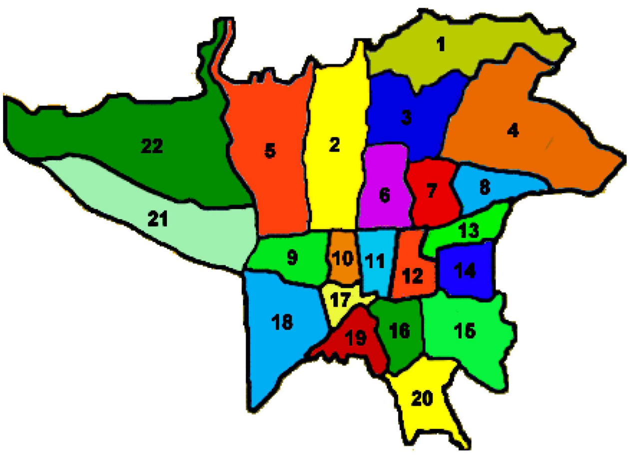


Figure 15. The customers' zones in Tehran.

The proposed model is implemented based on the above data. The optimal solution is shown in Table 8. The total cost is equal to 2320 using the proposed model, optimal production schedule is shown in the second row, and the best vehicles routing for this problem are shown in the last five rows of Table 8.

For evaluation of system performance, five plans have been considered as follows:

1. Plan 1: Replace existing vehicles with 3 new vehicles having a capacity of 200;
2. Plan 2: Replace existing vehicles with 7 vehicles having a capacity of 85;
3. Plan 3: Repair existing vehicles and increase their speed by 10%;
4. Plan 4: 30% reduction in processing time of stations 1, 2, 5;
5. Plan 5: 20% reduction in assembly time (stations 3, 4).

In order to analyze the sensitivity of the model, comprehensive analysis is presented for the following five managers' improvement plan. After making modifications to the model in any plan, a new problem was solved by ICA whose results are shown in Table 9.

Based on the results shown in Table 9, implementation of the first plan will increase objective value by as much as 3.8 percent, because it can increase the length of routes traveled by vehicles. Thus, this plan could be selected only if a significant reduction occurred in the total cost of system by decreasing vehicles. In the following, the second, third, fourth, and fifth planes improve the objective values by 10.2, 14.2, 6.9, and 3.7 percent, respectively. Based on the results, the amount of improvement of the objective function by implementing plan three (overhauling the current vehicles) has led to the greatest reduction in the objective values.

7. Conclusions and further research

In this investigation, the aim was to develop a new integrated model for a flow shop and distribution-scheduling problem. In the proposed model, a permutation flow shop scheduling was assumed for production system, and a distribution with routing was assumed for delivery system. The objective was to minimize the Maximum Returning Time (MRT) for integrated model, meaning that the time of the last order delivered to relevant customer by the last vehicle and that vehicle returning to production center. In order to solve the problem, the imperialist competitive algorithm (I-ICA) was proposed. To enhance the performance of

Table 7. Customers' location and orders data.

Customer/order	SC station	PVC station	IA station	FA station	P station	Location (X axis)	Location (Y axis)	Occupied space
1	27	79	22	93	38	220	550	15
2	92	23	93	22	84	380	340	29
3	75	66	62	64	62	390	380	37
4	94	5	53	81	10	450	610	36
5	18	15	30	94	11	420	690	3
6	41	51	34	97	93	110	10	1
7	37	2	27	54	57	290	250	23
8	58	81	30	82	81	260	370	2
9	56	12	54	11	10	60	520	8
10	20	40	77	91	40	170	680	1
11	2	59	24	23	62	700	490	32
12	39	32	47	32	49	640	190	8
13	91	16	39	26	90	550	670	38
14	81	87	66	22	34	90	640	31
15	33	78	41	12	11	390	320	33
16	14	41	46	23	81	40	510	17
17	88	43	24	34	51	470	390	12
18	22	94	23	87	21	580	700	27
19	36	1	68	59	39	640	270	34
20	65	93	50	2	27	100	620	35
21	25	35	17	12	23	650	510	5
22	33	42	23	18	43	350	420	7

Table 8. Customers' location and orders data.

Objective value	2320
Production scheduling	21-6-22-12-7-16-20-4-11-2-1-18-19-8-3-14-13-15-5-17-9-10
Delivery scheduling	Vehicle 1: 20-10-2-15 Vehicle 2: 12-19-11-1 Vehicle 3: 5-13-18-21 Vehicle 4: 4-16-6-7 Vehicle 5: 3-17-22-14-9-8

Table 9. Sensitivity analysis of the studied problem.

Plan	Objective value	% Improvement
Plan 1 (decrease vehicles)	2409	–3.8
Plan 2 (increase vehicles)	2080	10.3
Plan 3 (repair vehicles)	1991	14.2
Plan 4 (replace production machines)	2160	6.9
Plan 5 (increase production workforce)	2242	3.7

the suggested algorithm, some effective policies are proposed for improving the ICA performance:

- Three policies for generating initial empires;
- Three policies for assimilation;
- Two procedures for revolution;
- New repair procedure for infeasible solutions.

The original ICA was renamed to I-ICA algorithm based on these four main changes in original ICA. Three proper and useful sets of test problems were generated randomly and used for computational study. Taguchi method was used for the proposed algorithm parameter tuning. Computational results done on benchmark problems showed the effectiveness of integration of the proposed model and suggested approach with regard to the other methods. The useful statistical tool is used to evaluate the value of integration performance of the proposed model. One of the more significant findings to emerge from this study is that the integration of scheduling with delivery leads to the improvement of model performance. In this study, all parameters of the proposed model are deterministic; developing stochastic model could be a study case for further research. As another direction for future research, it could be considered to apply other new efficient meta-heuristic algorithms, such as bat algorithm, cuckoo search algorithm, neural network, and firefly algorithm, to this problem to obtain better solutions. In addition, other performance measures, such as mean tardiness, maximum tardiness, and mean completion time, can be considered instead of the MRT criterion. Other distribution approaches could be suggested in integrated model instead of distribution with routing. It would be valuable to extend this paper to allow for some other cost definitions or some other manufacturing environments including flow shops and job shops.

References

1. Chang, Y.C. and Lee, C.Y. "Machine scheduling with job delivery coordination", *European Journal of Operational Research*, **158**, pp. 470-487 (2004).
2. Potts, C.N. "Analysis of a heuristic for one machine sequencing with release dates and delivery times", *Operations Research*, **28**, pp. 1436-1441 (1980).
3. Hall, L.A. and Shmoys, B. "Jackson's rule for single-machine scheduling: Making a good heuristic better", *Mathematics of Operations Research*, **17**, pp. 22-35 (1992).
4. Matsuo, H. "The weighted total tardiness problem with fixed shipping times and overtime utilization", *Operations Research*, **36**, pp. 293-307 (1988).
5. Herrmann, J.W. and Lee, C.Y. "On scheduling to minimize earliness-tardiness and batch delivery costs with a common due date", *European Journal of Operational Research*, **70**, pp. 272-288 (1993).
6. Yuan, J. "A note on the complexity of single-machine scheduling with a common due date, earliness-tardiness, and batch delivery costs", *European Journal of Operational Research*, **94**, pp. 203-205 (1996).
7. Chen, Z.L. "Scheduling and common due date assignment with earliness-tardiness penalties and batch delivery costs", *European Journal of Operational Research*, **93**, pp. 49-60 (1996).
8. Cheng, T.C.E., Gordon, V.S., and Kovalyov, M.Y. "Single machine scheduling with batch deliveries", *European Journal of Operational Research*, **94**, pp. 277-283 (1996).
9. Yang, X. "Scheduling with generalized batch delivery dates and earliness penalties", *IIE Transactions*, **32**, pp. 735-741 (2000).
10. Hall, N.G., Lesaoana, M.A., and Potts, C.N. "Scheduling with fixed delivery dates", *Operations Research*, **49**, pp. 134-144 (2001).
11. Hall, N.G. and Potts, C.N. "Supply chain scheduling: Batching and delivery", *Working Paper*, Department of Management Sciences, The Ohio State University, Columbus, OH (2000).
12. Lee, C.Y. and Chen, Z.L. "Machine scheduling with transportation considerations", *Journal of Scheduling*, **4**, pp. 3-24 (2001).
13. Hall, N.G. and Potts, C.N. "Supply chain scheduling: batching and delivery", *Operations Research*, **51**, pp. 566-584 (2003).
14. Garcia, J.M., Lozano, S., and Canca, D. "Coordinated scheduling of production and delivery from multiple plants", *Robotics and Computer-Integrated Manufacturing*, **20**, pp. 191-198 (2004).
15. Li, C.L., Vairaktarakis, G., and Lee, C.Y. "Machine scheduling with deliveries to multiple customer locations", *European Journal of Operational Research*, **164**, pp. 39-51 (2005).
16. He, Y., Zhong, W., and Gu, H. "Improved algorithms for two single machine scheduling problems", *Theoretical Computer Science*, **363**, pp. 257-265 (2006).
17. Zhong, W.Y., Dosa, G., and Tan, Z.Y. "On the machine scheduling problem with job delivery coordination", *European Journal of Operational Research*, **182**, pp. 1057-1072 (2007).
18. Su, C.Sh., Pan, J.C.H., and Hsu, T.S. "A new heuristic algorithm for the machine scheduling problem with job delivery coordination", *Theoretical Computer Science*, **410**, pp. 2581-2591 (2009).
19. Li, C.L. and Vairaktarakis, G.L. "Coordinating production and distribution of jobs with bundling operations", *IIE Transactions*, **39**, pp. 203-215 (2007).
20. Wang, X.L. and Cheng, T.C.E. "Machine scheduling with an availability constraint and job delivery coordination", *Naval Research Logistics*, **54**, pp. 11-20 (2007).
21. Chen, B. and Lee, C.Y. "Logistics scheduling with batching and transportation", *European Journal of Operational Research*, **189**, pp. 871-876 (2008).
22. Lu, L., Yuang, J., and Zhang, L. "Single machine scheduling with release dates and job delivery to minimize the makespan", *Theoretical Computer Science*, **393**, pp. 102-108 (2008).
23. Liu, P. and Lu, X. "An improved approximation algorithm for single machine scheduling with job delivery", *Theoretical Computer Science*, **412**, pp. 270-274 (2011).
24. Woeginger, G.J. "Heuristics for parallel machine scheduling with delivery times", *Acta Informatica*, **31**, pp. 503-512 (1994).

25. Wang, G. and Cheng, T.C.E. "Parallel machine scheduling with batch delivery costs", *International Journal of Production Economics*, **68**, pp. 177-183 (2000).

26. Hall, N.G. and Potts, C.N. "The coordination of scheduling and batch deliveries", *Annals of Operations Research*, **135**, pp. 41-64 (2005).

27. Chen, Z.L. and Vairaktarakis, G.L. "Integrated scheduling of production and distribution operations", *Management Science*, **51**, pp. 614-628 (2005).

28. Ullrich, C.A. "Integrated machine scheduling and vehicle routing with time windows", *European Journal of Operational Research*, **227**, pp. 152-165 (2013).

29. Wan, L. and Zhang, A. "Coordinated scheduling on parallel machines with batch delivery", *International Journal of Production Economics*, **150**, pp. 199-203 (2014).

30. Lee, I.S. "A coordinated scheduling of production-and-delivery under dynamic delivery cost environments", *Computers & Industrial Engineering*, **81**, pp. 22-25 (2015).

31. Lu, L., Zhang, L., and Wan, L. "Integrated production and delivery scheduling on a serial batch machine to minimize the makespan", *Theoretical Computer Science*, **572**, pp. 50-57 (2015).

32. Yuan, J., Soukhal, A., Chen, Y., and Lu, L. "A note on the complexity of flow shop scheduling with transportation constraints", *European Journal of Operational Research*, **178**, pp. 918-925 (2007).

33. Pan, J.C.H., Wu, C.L., Huang, H.C., and Su, C.S. "Coordinating scheduling with batch deliveries in a two-machine flow shop", *International Journal of Advanced Manufacturing Technology*, **40**, pp. 607-616 (2009).

34. Soukhal, A., Oulamara, A., and Martineau, P. "Complexity of flow shop scheduling problems with transportation constraints", *European Journal of Operational Research*, **161**, pp. 32-41 (2005).

35. Atashpaz, G.E. and Lucas, C. "Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition", *IEEE Congress*, pp. 4661-4667 (2007).

36. Seidgar, H., Kiani, M., Abedi, M., and Fazlollahtabar, H. "An efficient imperialist competitive algorithm for scheduling in the two-stage assembly flow shop problem", *International Journal of Production Research*, **52**, pp. 1240-1256 (2014).

37. Rabiee, M., Zandieh, M., and Jafarian, A. "Scheduling of a no-wait two-machine flow shop with sequence-dependent setup times and probable rework using robust meta-heuristics", *International Journal of Production Research*, **50**, pp. 7428-7446 (2012).

38. Behnamian, J. and Zandieh. M. "A discrete colonial competitive algorithm for hybrid flowshop scheduling to minimize earliness and quadratic tardiness penalties", *Expert Systems with Applications*, **38**, pp. 14490-14498 (2011).

39. Shokrollahpour, E., Zandieh, M., and Dorri, B. "A novel imperialist competitive algorithm for bi-criteria scheduling of the assembly flowshop problem", *International Journal of Production Research*, **49**, pp. 3087-3103 (2011).

40. Attar, S.F., Mohammadi, M., and Tavakkoli-Moghaddam, R. "A novel imperialist competitive algorithm to solve flexible flow shop scheduling problem in order to minimize maximum completion time", *International Journal of Computer Applications*, **28**, pp. 27-32 (2011).

41. Karimi, N., Zandieh, M., and Najafi, A.A. "Group scheduling in flexible flow shops: a hybridised approach of imperialist competitive algorithm and electromagnetic-like mechanism", *International Journal of Production Research*, **49**, pp. 4965-4977 (2011).

42. Javadian, N., Rezaeian, J., Khorshidian, H., and Rahmani, K. "Single machine preemptive scheduling by hybridized meta-heuristic approach", *IEEE*, pp. 750-753 (2011).

43. Carlier, J. "Ordonnancements a contraintes disjonctives", *Recherche Perationelle/Operations Research*, **12**, pp. 333-351 (1978).

44. Reeves, C.R. "A genetic algorithm for flow shop sequencing", *Computers & Operations Research*, **22**, pp. 5-11 (1995).

45. Taillard, E. "Benchmarks for basic scheduling problems", *European Journal of Operational Research*, **64**, pp. 278-285 (1993).

46. Montgomery, D.C., *Design and Analysis of Experiments*, 5th ed., Wiley, New York (2000).

47. Naderi, B., Zandieh, M., and Fatemi-Ghomi, S.M.T. "Scheduling job shop problems with sequence dependent setup times", *International Journal of Production Research*, **47**, pp. 5959-5976 (2009).

48. Taguchi, G. "Introduction to quality engineering", *White Plains*, Asian Productivity Organization/UNIPUB (2000).

49. Qian, B., Wang, L., Hu, R., Wang, W.L., Huang, D.X., and Wang, X. "A hybrid differential evolution method for permutation flow-shop scheduling", *International Journal of Advanced Manufacturing Technology*, **38**, pp. 757-777 (2008).

50. Nawaz, M., Enscore, J.E., and Ham, I. "A heuristic algorithm for the m-machine, n-job flowshop sequencing problem", *OMEGA*, **11**, pp. 91-95 (1983).

51. Mokhtari, H., Kamal-Abadi, I.N., and Cheraghalikhani, A. "A multi-objective flow shop scheduling with resource-dependent processing times: trade-off between makespan and cost of resources", *International Journal of Production Research*, **49**(19), pp. 5851-5875 (2010).

52. Campbell, H.G., Dudek, R.A., and Smith, M.L. "A heuristic algorithm for the n job, m machine sequencing problem", *Management Science*, **16**(10), pp. 630-637 (1970).

Biographies

Shahriar Mohammadi is a former Senior Lecturer at the University of Derby, UK. He also used to be a Network Consultant in the UK for more than fifteen years. He is currently a Lecturer in the Industrial Engineering Department of the University of K.N. Toosi, Iran. His main research interests and lectures are in the fields of networking, data security, network security, e-commerce and e-commerce security. He has published more than eighty papers in various journals and conferences as well as four books.

Ali Cheraghalikhani received his BSc degree from Iran University of Science and Technology, Tehran, Iran in 2007 and MS degrees from K.N. Toosi University of Technology, Tehran, Iran in 2010, both in Industrial Engineering. Currently, he is a PhD candidate in the same field at K.N. Toosi University of Technology. His research background and interests mainly include supply chain management, algorithms and applied production and operations research.

Reza Ramezanian received BSc, MSc, and PhD degrees, all in Industrial Engineering, from Iran University of Science and Technology, Tehran, Iran. He is currently an Assistant Professor of Industrial Engineering at K.N. Toosi University of Technology, Tehran, Iran and is reviewer of several international journals. His research interests include operation research, stochastic optimization, supply chain management, production planning, dynamic lot-sizing, and scheduling theory and applications.