A green multi-objective integrated scheduling of production and distribution with heterogeneous fleet vehicle routing and time windows

Maliheh Ganji a, Hamed Kazemipoor a,⁎, Seyyed Mohammad Hadji Molana b, Seyed Mojtaba Sajadi c

a Department of Industrial Engineering, Central Tehran Branch, Islamic Azad University, Tehran, Iran
b Department of Industrial Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran
c New Business Department, Faculty of Entrepreneurship, University of Tehran, Tehran, Iran

ABSTRACT

Scheduling the supply chain in an integrated manner is a quite fundamental subject in supply chain management. To consider due-dates with production and distribution times at the same time could lead to reduced costs and, thereby, increased profitability. Integrating the problems dealing with due-date assignment, production/distribution times, and routing, while incorporating environmental considerations, could not only increase the cohesion among different decision-making levels and reduce costs in the long run, it would also contribute to the improvement of environmental conditions and benefit the world population. In this study, the integrated supply chain scheduling problem features assignment of due dates, batch delivery, assignment to multiple heterogeneous vehicles based on their capacity, and delivery of customer orders in time-windows. The objective is to minimize distribution cost, fixed and variable fuel costs, the carbon emitted by the vehicles, total delivery tardiness, and customer dissatisfaction. In this model, the customers are categorized into five clusters based on the length of their association, how recently they began interacting, how often they interact, and the monetary value of their interaction with the organization. The members of these clusters are called core customers, potential customers, new customers, lost customers, and resource-consumption customers. A mixed integer non-linear programming model is introduced for this problem which is solved using three multi-objective metaheuristic algorithms: Multi-Objective Particle Swarm Optimization, Non-dominated Sorting Genetic Algorithm II, and Multi Objective Ant Colony Optimization. A number of performance criteria and statistical tests are used to evaluate the algorithms.

© 2020 Elsevier Ltd. All rights reserved.

KEYWORDS: Integrated Scheduling Heterogeneous fleet vehicle routing problem Batch delivery Time-windows Customer dissatisfaction

1. Introduction

In relation with green supply chains management (GSCM) activities and the development of products, environmental analysis must be integrated with processes such as manufacturers, distributors and suppliers. Throughout the recent decades, integrated supply chains have been established as a key topic in the literature and practices of industrial engineering. Naturally, were each of these components to plan and work individually and in isolated fashion; they would prioritize their own profit and simply would not feel obligated to protect the interests of the other components. However, in a scenario where all the planning and operations are centralized and the decisions are made with the aim of preserving all the components’ interests, the outcome will be what is known as an integrated supply chain which is also optimized to satisfy the interests of all involved. The literature on the subject is replete with research works that strongly point to the benefits of decisions which are made in an integrated manner. For instance, Chandra and Fisher (1994) concluded that integrated supply chain decisions in production and scheduling phases could lead to a total cost reduction of 3–20 percent. This study, however, focuses on a hypothetical supply chain in which production scheduling and the distribution of customer’s orders are integrated. When it comes to

⁎ Corresponding author.
E-mail addresses: Mal.ganji.eng@iauctb.ac.ir (M. Ganji), h.kazemipoor@iauctb.ac.ir (H. Kazemipoor), Molana@srbiau.ac.ir (S.M. Hadji Molana), msajadi@ut.ac.ir (S.M. Sajadi).

https://doi.org/10.1016/j.jclepro.2020.120824
0959-6526/© 2020 Elsevier Ltd. All rights reserved.
customers who are geographically scattered, efficient order distributions are regarded as a fundamental element of supply chain success. It goes without saying that delivery tardiness could only cause customer dissatisfaction and, in some cases, losing the customer altogether. To prevent this outcome, many businesses have turned to the distribution technique that Cheng and Kahlbacher introduced in 1993 is commonly referred to as batch delivery. In this technique, multiple products are transported to their new owners in one go, and the delivery vehicle starts the distribution only when the last order has been processed and is ready to be delivered. The proposed supply chain in this study features a batch delivery system which includes routing decisions as well.

Today, just-in-time (JIT) manufacturing systems are drawing the attention of most production and service industries (Sajadi and Rad, 2016). A supply chain, as a whole, determines its strategy based on several factors, including customer locations, response time, variety, availability, and product return-ability (Li et al., 2018). Assessing these factors enables us to identify the most suitable distribution strategy, including the transportation method.

When dealing with multiple customers, the delivery of orders requires the development of a suitable strategy which not only meets the customers’ demands but also minimizes the delivery cost through an efficient transportation method. The heterogeneous fleet vehicle routing problem (HFVRP) is another variation of VRP, introduced in 1984 by Golden et al., where the distributors’ vehicles incur different amounts of costs (fixed and variable). Another important variation of the routing problem is the Vehicle Routing Problem with Time Windows (VRPTW) and a heterogeneous fleet. The VRPTW, introduced by Solomon (1986), is an extension of the Capacitated Vehicle Routing Problem (CVRP) where each customer is provided with service within a predefined timespan.

Protecting the environment is another fundamental issue in our world today. Pressured and influenced by the numerous civil movements and institutions now working across the world to preserve planet earth, most organizations have focused their efforts on developing and providing environmentally sound products and services (Afshar-Bakeshloo et al., 2018). The circumstances are now such that the future position of corporations in the global market is shaped by the steps they are taking today to give their contribution to the environmental cause. On this basis, integrating the problems of due-date assignment, production/distribution schedule, and routing, while considering environmental responsibilities can potentially reduce long-term costs, increase decision-making cohesion across the supply chain network, and benefit the world population at large by lowering the damage to the environment. The most basic element of GSCM is arguably a clean production (Zhao et al., 2017).

Five sets of objectives were defined for the problem considered in this study:

- **Scheduling Objectives**: Tardiness minimization, reduced completion time
- **Transportation Objectives**: Distribution cost minimization, reduced number of load/unload instances
- **Production Objectives**: Inventory cost minimization
- **Environmental Objectives**: Fuel consumption minimization
- **Customer Satisfaction Objectives**: Order delivery within deadline

In the model proposed by this study, the first decision involves the sequence in which the orders should be processed. An order i.e. product may be either delivered to the customer immediately after it has been processed, or wait for the other orders assigned to the same batch to be processed so that a full shipment is delivered. Therefore, a batch cannot be completed until the last order included in it is completed. This characteristic often leads to tardiness; therefore, a tradeoff should be made between distribution costs and scheduling objectives. When the total distance of the vehicle is reduced, it does not mean that an optimal solution is found which automatically reduces energy consumption and contaminant emissions, as well. This is down to the simple fact that distance is not the sole decisive factor when it comes to energy consumption and many other factors (e.g., vehicle load, weight, quality, speed, etc.) are also involved to varying degrees. Therefore, a number of heterogeneous vehicles with limited capacity and different transportation costs are considered for the delivery of orders. The customers are geographically scattered, receive their orders within their prioritized time-window, and may place one or more orders each. Furthermore, each batch does not necessarily contain the orders of one customer and different orders can be assigned to it. Various routes are determined for the delivery of orders to customers and, as a result, vehicle routing decisions are added to the problem. In this mode, the customers should be divided into cluster, with each cluster containing the customers to whom a vehicle is assigned in order that the total transportation cost paid by the customers is minimized. Each vehicle, meanwhile, can provide service to more than one customer in accordance with the aforementioned tradeoff established between the costs that customer waiting time and regular transportation costs can incur.

From a managerial standpoint, this study may lead to reduced distribution, inventory, and holding costs; minimize energy i.e. fuel consumption and delivery tardiness, all of which contribute to increasing customer satisfaction.

In the past two decades, numerous studies have been conducted on integrated production and distribution models at the strategic level, and to a slightly lesser extent, at the tactical level. Several review papers have also been produced on the subject (Sarmiento and Nagi, 1999; Goetschalckx et al., 2002; Chen, 2010; Moons et al., 2017).

It appears as though the first paper on this subject was published in 1980 by Potts. This study considered transportation times but not the loading capacity or transportation costs. Lee and Chen (2001) is one of the first examples of research on integrated models which address the vehicles’ loading capacity, and they considered various problems featuring the direct delivery method.

Cheng et al. (1996) dealt with a single-machine scheduling problem with batch delivery, with the objective of minimizing the instances of earliness penalty as well as the number of batches. Hall and Potts (2003) worked on an order scheduling problem on a single machine, assuming that there are multiple batches and such factors as the number of late jobs, maximum tardiness, and flow time. Mazdeh et al. (2007) studied the scheduling of a set of orders with specific release times on a single machine for delivery in batches to customers or to other machines for further processing. Rostami et al. (2015) attempted to schedule a set of orders with specific release times to be processed by a single machine and then transferred to a customer or another machine through batch delivery. Hamidinia et al. (2012) used a genetic algorithm to solve scheduling problems to minimize the weighted jobs' tardiness or earliness in a distribution system based on batch delivery. Mazdeh et al. (2013) examined single-machine scheduling problems with batch delivery to minimize maximum tardiness but without considering availability times. Yin et al. (2014) deal with the scheduling problem of n non-resumable jobs in a single machine with batch delivery. Gao et al. (2015) studied a production and distribution integration problem where the jobs are first process...
and then delivered in batches through a number of vehicles with limited capacity. Yin et al. (2016) address an integrated production and scheduling problem involving two agents, each having a set of jobs that compete for the use of a common machine. Sawik (2016) presented a supply chain in which the scheduling of supply, production and distribution are integrated and disruption risks are incorporated. Jamili et al. (2016) worked on a bi-objective integrated model for production and distribution scheduling in which there are restrictions on order release dates. Gharaei and Jolai (2018) consider a batch delivery problem in a multi-factory supply chain. Gharaei and Jolai (2019) study a two-agent integrated production and distribution scheduling through a branch and price approach. Yin et al. (2013) worked on a single-machine batch delivery scheduling with an assignable common due date and controllable processing times. Dumitrescu et al. (2015) attempted to achieve optimal delivery times in a batch delivery system with the goal of minimizing tardiness penalties and delivery costs. Rasti-Barzoki and Hejazi (2015) considered an integrated problem of due date assignment, production and batch delivery with controllable processing times for multiple customers in a supply chain with the goal of minimizing the number of late jobs. Li (2015) worked on a single-machine scheduling problem with common due window assignment and batch delivery where the start time and the size of due windows are decision variables. Wang and Liu (2014) presented a scheduling problem on m parallel machines with batch delivery through a single machine. Zhong and Jiang (2015) worked on a parallel machine scheduling problem in a single environment with batch delivery to two customers using the VRP. Karimi and Davoudpour (2017) introduced a new supply chain scheduling problem where stage-dependent inventory costs were also taken into account. Gu et al. (2015) studied a Mutualism Quantum Genetic Algorithm for scheduling an integrated supply chain considering materials pickup, flow shop scheduling, and delivery of the finished products, with the goal of minimizing completion times. Agnetis et al. (2014) worked on coordinating a problem of production and scheduling with batch delivery where a Third-Party Logistics supplier transfers half-finished products from one manufacturing facility to another inside a factory. The findings of the above mentioned studies on integrated manufacturing and distribution problems under different conditions are summarized in Table 1.

The above survey of the various types of integrated production and distribution problems (Table 1) points to the following conclusions:

- A majority of the research works in the literature address objective functions that feature delivery times.
- Problems dealing with multi-customer batch delivery in addition to problems featuring fixed delivery times have not received as much attention as problems that focus on single-customer and immediate deliveries.
- No researchers have so far dealt with a delivery problem comprising heterogeneous fleets and multiple time windows which also aims to fuel cost, carbon emission, and customer dissatisfaction.

The authors consider the following as the main contributions of this paper:

1. The paper is work on a supply chain scheduling problem in which the minimization of fuel cost, carbon emission, customer dissatisfaction, and multiple time-windows.
2. Various mathematical models, including linear and non-linear mixed integer programming are utilized.
3. The paper considers product delivery in multiple time-windows.
4. This study, transportation vehicles make the deliveries in order of customer priority; where customers have each been put in the groups of set \( G = \{1, 2, \ldots, g\} \) as per the aforementioned model of LRFM values.

Section 2 presents a description of the central problem, and a mathematical model will be introduced to solve it. Section 3 is where the proposed metaheuristic algorithm will be introduced. Section 4 consists of extensive analyses and numerical examples. The computational and performance experiments — conducted as part of the algorithm proposed by this paper — generate results which will be presented, compared and elaborated upon in section 5. Finally, section 6 contains the conclusion along with some suggestions for future research.

2. Problem description

In this problem, each customer prioritizes a set of time-windows, rather than just one, for themselves. Customers who are provided service in their prioritized time window will be the most satisfied, while the lower the priority of the time window in which they are served, the more dissatisfied they will be. To further clarify the concept, assume that customer \( j \) proposes a set of five time windows as follows:

\[
\text{TW}_j = \{tw_1 = (a_1, b_1), tw_2 = (a_2, b_2), tw_3 = (a_3, b_3), tw_4 = (a_4, b_4), tw_5 = (a_5, b_5)\}
\]  

(1)

Furthermore, without reducing the universality of what has been said thus far, it is assumed that the said customer’s proposed time windows are sorted by order of priority as follows:

\[
tw_1 > tw_2 > tw_3 > tw_4 > tw_5
\]  

(2)

In this case, the extent of customer \( j \)’s dissatisfaction after the delivery of the order in each of the time windows above is as follows:

\[
\lambda_{tw} = \begin{cases} 
0 & \text{tw} = tw_1 \\
\lambda_2 & \text{tw} = tw_2 \\
\lambda_3 & \text{tw} = tw_3 \\
\lambda_4 & \text{tw} = tw_4 \\
\lambda_5 & \text{tw} = tw_5 \\
\infty & \text{tw} \notin \text{TW}_j
\end{cases}
\]  

(3)

Where \( \lambda_2 < \lambda_3 < \lambda_4 < \lambda_5 \), the customer’s satisfaction with being placed in the time window with the highest priority equals 0, and every customer must be served within one of the windows. What needs to be noted here is the difference between the VRP with Soft Time Windows (VRPSTW) and VRP with Multiple Prioritized Time Windows (VRPMPTW). In the VRPSTW, each customer proposes only one-time window, and the supplier is given the choice to violate the time-window at the cost of a tardiness penalty which should be incorporated by the objective function. With the VRPMPTW, however, each customer proposes a set of prioritized time windows and delivering the order within each time window represents a different value for the customer. Moreover, the priority in each tour is with customers who are more valuable to the system. For instance, customers placed in the first group take the highest priority and are provided service in the quickest time possible.

In this model, three significant customer performance variables are proposed: how recent the last purchase has been made (R), the monetary value of the customer’s purchase (M) and how frequently the customer has made purchases (F). Today, the RFM model has been expanded and parameter \( L \), the length of time between a customer’s first and last purchase from the organization, has been added to generate the LRFM model (Wei et al., 2012). It should be noted that in the problem considered in this study, vehicles provide services based on the priority of the customers placed in each of the groups in set \( G = \{1, 2, \ldots, g\} \) according to the LRFM model.

The presented model may be regarded as a variation on what Li et al. (2018) and Gharaei and Jolai (2018) recently developed, with the difference stated in Table 1. It is worth noting, though, that Li et al. (2018) considered a routing problem with a heterogeneous fleet that aims to reduce fuel cost and carbon emission; while Gharaei and Jolai (2018) studied a multi-agent approach to the integrated production scheduling and distribution problem which is, however, devoid of the innovations of the present research (listed at the end of the previous section).

Carbon emission in vehicles depends on a number of factors, such as the type of vehicle, type of fuel, fuel efficiency, and the distance between the starting point and destination. The classic method of measuring fuel consumption is a linear function of the trip length which is not accurate enough. The model presented in this study, however, is expected to be more reliable as it takes more details into account, such as vehicle load, velocity and actual distance traveled. Table 2 presents a short list of some of the symbols and values used in the model.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Value/Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>( c_f )</td>
<td>Fuel cost per liter (CNY)</td>
<td>7.30</td>
</tr>
<tr>
<td>( c_e )</td>
<td>Emissions cost per liter (CNY)</td>
<td>0.64</td>
</tr>
<tr>
<td>( v_{ij} )</td>
<td>Average arc speed (km/h)</td>
<td>40–100</td>
</tr>
<tr>
<td>( a_{ij} )</td>
<td>Arc specific constant (related to acceleration, road angle)</td>
<td>0.09–0.15</td>
</tr>
<tr>
<td>( \beta_{ij} )</td>
<td>Vehicle-specific constant (frontal surface area, rolling resistance, and air density)</td>
<td>type of vehicle</td>
</tr>
</tbody>
</table>

2.1. Assumptions

- No delay is allowed in scheduling.
- The jobs are all available in the horizon of planning.
- There is not any setup time before processing the jobs and the amount of time required by the processes of loading and unloading will be added to the total transportation time.
- The horizon of planning is single-cycle \( k \), while the production system is devised as single-machine and single-product.
- The transportation method features routing and batch delivery.
- The processing time for each job is definite and fixed.
- The vehicles are considered to have different and limited capacities.
- No external factors are considered throughout the routes.
- Every customer’s demand is delivered by one vehicle.
- Routing is calculated without considering idle times for the vehicles.
- Every customer’s orders are delivered to the customer all at once and in one of the predetermined time windows.

2.2. Notation

The two dummy orders/customers (0) and \((n+1)\) with zero processing times and zero deadlines are introduced as the first and
last jobs in the sequence scheduled for the production facility. Additionally, customers are defined (0) and (n+1) where customer

\[
\text{Min} = \sum_{g=1}^{G} \sum_{j=1}^{n} \frac{TW}{tw} \cdot a_{g,H_{ij},\lambda_{j,\text{tw}},u_{tw,j}}
\]

subject to:

\[
\sum_{i=0}^{n} x_{ij} = 1 \quad \forall \ j = 1, \ldots, n + 1
\]

\[
\sum_{j=1}^{n+1} x_{ij} = 1 \quad \forall \ i = 0, \ldots, n
\]

\[
x_{ij} = 0 \quad \forall \ i = j
\]

\[
x_{0,n+1} = 0
\]

\[
x_{ij} + x_{ji} \leq 1 \quad \forall \ i,j = 1, \ldots, n
\]

(0) is the production facility and customer (n+1) represents the return to the production facility. The mathematical model is described below:

\[
\text{Min} = \sum_{g=1}^{G} \sum_{j=1}^{n} \frac{TW}{tw} \cdot a_{g,H_{ij},\lambda_{j,\text{tw}},u_{tw,j}}
\]

subject to:

\[
\sum_{i=0}^{n} x_{ij} = 1 \quad \forall \ j = 1, \ldots, n + 1
\]

\[
\sum_{j=1}^{n+1} x_{ij} = 1 \quad \forall \ i = 0, \ldots, n
\]

\[
x_{ij} = 0 \quad \forall \ i = j
\]

\[
x_{0,n+1} = 0
\]

\[
x_{ij} + x_{ji} \leq 1 \quad \forall \ i,j = 1, \ldots, n
\]
The objective functions are presented in relations (4), (5) and (6). The phrases of this model include: transportation cost among the customers, transportation cost from the manufacturing facility to the first customer; cost of returning from the last customer to the manufacturing facility, fixed cost of each distribution tour, and fuel cost and carbon emission. Additionally, the second objective function represents the tardiness in delivering the customers’ orders. The third and last objective function represents the tardiness with being served in their less-preferred time windows. Relations (7) to (11) guarantee that each job should be either the first or the last job processed across the supply chain; otherwise, there are a direct predecessor job and a direct successor job in the processing sequence as well. Each job is assigned to a specific batch or heterogeneous vehicle through relation (12). If a job gets assigned to a vehicle, relations (13) to (17) guarantee that each job should be either the first or the last job delivered to a customer by a heterogeneous vehicle; otherwise, the job has a direct predecessor job and a direct successor job in the delivery sequence as well.

\[ CB_{vk} \geq C_j - M \left( 1 - y_{i,j}^v \right) \quad \forall \ k = 1, \ldots, K \quad \forall \ v = 1, \ldots, V, i = 1, \ldots, n \]  

\[ A_j \geq CB_{vk} + t_{oj} - M \left( 1 - z_{i,j}^v \right) \quad \forall \ k = 1, \ldots, K \quad \forall \ v = 1, \ldots, V, j = 1, \ldots, n \]  

\[ A_j \geq A_i + t_{ij} - M \left( 1 - z_{i,j}^v \right) \quad \forall \ k = 1, \ldots, K \quad \forall \ v = 1, \ldots, V, i \neq j = 1, \ldots, n \]  

\[ T_j \geq A_j - d_j \quad \forall \ j = 1, \ldots, n \]  

\[ T_j \geq 0 \quad \forall \ j = 1, \ldots, n \]  

\[ W_{lj} \leq \text{cap}_v \quad \forall \ v = 1, \ldots, V, \ i, j = 1, \ldots, n \]  

\[ W_{lj} \geq \sum_{k=1}^{K} \sum_{v=1}^{V} q_{ij} y_{i,j}^v \quad \forall \ i, j = 1, \ldots, n \]  

\[ \sum_{tw=1}^{TW} u_{tw,j} = 1 \quad \forall \ j = 1, \ldots, n \]  

\[ A_j - d_j \geq \sum_{tw=1}^{TW} E_{tw,j} u_{tw,j} \quad \forall \ j = 1, \ldots, n \]  

\[ A_j - d_j \leq \sum_{tw=1}^{TW} L_{tw,j} u_{tw,j} \quad \forall \ j = 1, \ldots, n \]  

\[ x_{ij}, y_{i,j}^v, z_{i,j}^v, u_{tw,j}, H_{g_j} \in \{ 0, 1 \} \quad \forall \ k = 1, \ldots, K \quad \forall \ v = 1, \ldots, V \quad i, j = 1, \ldots, n \]  

\[ C_j, CB_{vk}, A_j, T_j, W_{lj} \geq 0 \quad \forall \ k = 1, \ldots, K \quad \forall \ v = 1, \ldots, V \quad i, j = 1, \ldots, n \]  

Relation (18) regulates the entry and exit capabilities into each customer node in the VRP. Relation (19) features the capacity of the heterogeneous vehicle. Relations (20) and (21) calculate the completion time of the first job and then the rest of the jobs assigned to the same vehicle. Relation (22) determines the start time of a vehicle’s delivery sequence which equals the completion time of the last job assigned to the vehicle. Relations (23) and (24) calculate the delivery time of each of the orders. Relations (25) and (26) specify each order’s tardiness. Relations (27) and (28) express that a vehicle’s load cannot exceed its capacity. Equation (29) expresses that each customer should be served solely within one-time window. Equations (30) and (31) indicate the time window within which each customer is provided with service. Integer constraint and non-negativeness of the variables are indicated by relations (32) and (33). Where \( W_{lj} \) represents the load capacity of the transportation vehicle, \( T_j \) is the tardiness of job \( j \), and \( CB_{vk} \) is the completion time of all the jobs delivered by vehicle \( k \) of type \( v \). The mathematical model above is a mixed integer nonlinear programming model described by binary variables \( x_{ij}, y_{i,j}^v, z_{i,j}^v, u_{tw,j}, H_{g_j} \) and continuous variables \( C_j, CB_{vk}, A_j, T_j, W_{lj} \). The model includes several binary variables which add to the complexity of the problem.

Based on Wang and Cheng (2000) and Mazdeh et al. (2010), production system problems whose objective is tardiness minimization are classified as NP-Hard. Moreover, Demir et al. (2012) consider VRP as strongly NP-Hard. Thus, in the case of the problem investigated in the present study, it can be said that an integrated production, distribution, and routing scheduling problem whose objective function consists of tardiness and distribution cost minimization, is also NP-Hard. Therefore, using heuristics and metaheuristics to solve the problem is logical and perhaps necessary.

Various recent studies in the field of scheduling and supply chain management have addressed multi-objective optimization problems. These studies often use metaheuristic algorithms to find Pareto-optimal solutions to such optimization problems (see moons (2016)). Three of the most widely-used metaheuristic algorithms are non-dominated sorting genetic algorithm (NSGA-II), multi-objective particle swarm optimization (MOPSO), and Multi Objective Ant Colony Optimization (MOACO). As such, the authors found it appropriate to utilize these three algorithms and then compare the obtained results.

3. Discussion

3.1. Solution procedure

There are several challenges when dealing with MOPs. When it comes to MOP, one must seek solutions with optimal variety and dispersion across the Pareto front. At the first stage, the problem was solved in small dimensions using the weighted-sum method, but the time the software took to compute the solution exceeded 5 h. Hence, it was decided to use evolutionary multi-objective optimization methods to address the problem.

3.1.1. NSGAII algorithm

The NSGAII algorithm for solving MOPs was introduced by Deb (2002). The major features of this algorithm include: Defining the concept of Crowding Distance, Using the Binary Tournament selection operator, saving and recording non-dominated solutions obtained in the previous steps of the algorithm. The details of the NSGAII algorithm and crowding distance are shown in Algorithms 1 and 2.
**Algorithm 1.** Pseudo code of NSGAII (adapted from Deb (2002))

```
Input: N, g, f_k(x) > N member evolved g generations to solve f_k(x)
Initialize Population P;
Generating for random population – size N;
Evaluating for Objectives Values;
Assigning Pareto sort Rank (level);
Generating Child Population;
Selecting the Binary Tournament;
Recombination and Mutation;
for i=1 to g do
   for each Parent and Child in Population do
      Assigning Pareto sort Rank (level);
      Generate non-dominated solution sets;
      Determine Crowding distance;
      Loop (inside) by adding solution to next generation starting from
      the first front until N individuals;
      end
   selecting the points on the lower front with high crowding distance;
   creating the next generation;
   Selection the Binary Tournament:
   Recombination and Mutation;
End
```

**Algorithm 2.** The crowding distance pseudo code (adapted from Deb (2002))

```
Getting the number of non-dominated results in the foreign repository
n = |T|
Initializing distance
For i=0 to maximize
T[i]. distance = 0
Calculate the results crowding distance
For each objective m
   Sorting each value of objective
   T = sort (T, m)
For i=1 to (n-1)
   T[i]. distance = T[i]. distance + (T[i+1]. m – T[i-1]. m)
Measure the maximum distance to the frontier points so that they are always chosen T[0]. distance = T[n]. distance = maximum distance
```

### 3.1.2. MOPSO algorithm

In MOPSO algorithm, the particles must adapt to the direction of each objective function. Hence, every particle in every group has its own p_{best}, but the b_{best} of each group is replaced by the b_{best} of the other groups for the next repetitions. In this way, a dynamic archive for exchange of particles is created by each repetition, the particles are compared with one another, the non-dominated solutions are stored, and the rest of the solutions are eliminated at the end of each repetition. The details of the MOPSO algorithm are shown in **Algorithm 3.**

**Algorithm 3.** The Pseudo code of the standard MOPSO algorithm (adapted from Rahimi et al. (2018))

```
Begin
   for initializing for each swarm particle
      position & velocity haphazardly
   end for
   Initialize External Archive (EA) Quality
   do
      for each swarm particle
         select Evaluating a particle from EA in the swarm
         fitness function
         if the objective fitness value is better than the personal best objective fitness value (p_{best}) in record
            current fitness value is selected as the new personal best (p_{best})
         end if
         update the particle velocity
         update the particle position
      end for
   update leader in EA
   Quality (leader)
   until stopping criteria is satisfied
   report the account of EA
End
```
3.1.3. MOACO algorithm

This algorithm was originally conceived by drawing inspiration upon studies and observations conducted on ant colonies. One of the most significant and interesting behaviors displayed by ants is their method of seeking food, particularly the way they find the shortest route between their colony and the food source. The details of the ACO algorithm are shown in Algorithm 4.

Algorithm 4. Ant colony optimization (adapted from Mokhtari and Ghezavati (2018))

In this problem, there are two sets of solutions in this algorithm; EP represents a set of non-dominated solutions and $L_t$ is the prohibited list, which represents customers already visited by the ants. The main problem of this study features two sub-problems which include the scheduling of orders and routing the vehicle between the nodes (customers). Next, an ant movement pattern is defined for each sub-problem. At the end, after generating a full solution for each sub-problem, the solution of the scheduling problem is linked to that of the routing problem. The details of the MACO algorithm are shown in Algorithm 5.

Algorithm 5. Multi-objective ant colony optimization

The maximum number of repeats was used to stop the algorithm. Goksal et al. (2013), Ai and Kachitvichyanukul (2009), and Gharaei and Jolai (2018) were consulted in order to generate the initial MOPSO and NSGAII algorithms.

3.2. Local search

The local search approach is used when full solutions have been generated by the ants. Using local search in evolutionary algorithms improves the quality of the solutions. Local search is executed in two stages. First, a solution is selected after the ants have created a full schedule in the scheduling problem, and then the sequence of jobs will be defined in accordance with the (earliest due date) priority rule so as to reduce the delay in sending the orders to customers. The solution is selected based on the roulette wheel method, where solutions with larger value of selection probability have a bigger chance of being selected.
4. Computational experiments

4.1. Data generation

A set of random data were developed in order to express the model's and the algorithm's efficiency using discrete uniform distributions based on Table 3 and in accordance with Gharaei and Jolai (2018) and Li et al. (2018) as follows:

In this paper, because of the relation between processing times, delivery times and deadlines, which are considered the key parameters, the processing times were developed according to Table 3 and based on several instances of trial and error while executing the model. With tardiness a major factor in the problem, delivery times should have been considered in such a way that there would be tardiness in some of them. Relation \( d_i \sim U \left( \frac{P_i + 2 \times (F + T)}{2} \right) \) is used to calculate delivery times (Gharaei and Jolai, 2018). The other parameters were developed based on Table 4 in the form of eight problems.

The other parameters of this numerical example including the number of customer clusters, importance of each cluster’s customers, total number of customers’ time windows, and customers’ dissatisfaction with being placed in lower-priority time windows are listed in Table 5. In each problem group, the customers are categorized based on the LRFM model into the following five clusters: core customers, potential customers, new customers, lost customers and Resource-consumption customers.

4.2. Performance measurement

Zitzler et al. (2000) proposed a number of meticulously studied metrics to gauge the performance of algorithms. In this study, however, six performance metrics are utilized which are defined as follows: Generational Distance (GD), Average Computational Time (ACT), Ratio of Non-Dominated Individuals (RNI), Mean Ideal Distance (MID), Spacing Metric (SP) and Maximum Diversity (MD). Moreover, to demonstrate the results of measurements conducted on the algorithms’ performances more clearly, the weight metric (WM), a combination of the five metrics, is calculated as follows:

\[
WM = \frac{RNI + (2 \times GD) + (2 \times MID) + (2 \times SP) + (2 \times MD)}{9}
\]  

Due to the higher importance of the extent and even distribution of solutions, the latter have higher weights. In order to use this metric, the values of the four metrics should be normalized using the RDI relation.

4.3. Parameter setting

Algorithm’s parameters have a significant impact on the exploration and exploitation rates, therefore, they must be chosen through an empirical approach. Consequently, following extensive trial and error, a number of parameters were obtained for MOPSO and NSGAII and MOACO which are listed in Table 6. Although the parameters are not absolutely optimal, the execution of the algorithm proved that it performs perceptibly better than with the resulting parameters when compared with other ones. Algorithm parameters and their values are in Table 6.

The proposed metaheuristics were coded in Python 3.7 and run

<table>
<thead>
<tr>
<th>Table 3</th>
<th>Random data of problem (Gharaei and Jolai, 2018) and (Li et al., 2018).</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
<td>Uniform Distribution ([a,b])</td>
</tr>
<tr>
<td>(p_i)</td>
<td>([100,200])</td>
</tr>
<tr>
<td>(q_i)</td>
<td>([10,40])</td>
</tr>
<tr>
<td>(l_i,t_{ij})</td>
<td>([30,150])</td>
</tr>
<tr>
<td>(c_i)</td>
<td>7.30</td>
</tr>
<tr>
<td>(c_F)</td>
<td>0.64</td>
</tr>
<tr>
<td>(f_{p,i})</td>
<td>([100,500])</td>
</tr>
<tr>
<td>(v_{ij})</td>
<td>([0.09,0.15])</td>
</tr>
<tr>
<td>(w_v)</td>
<td>([1500,5000])</td>
</tr>
<tr>
<td>(d_i)</td>
<td>([30,200])</td>
</tr>
<tr>
<td>(M)</td>
<td>10000</td>
</tr>
<tr>
<td>(l_{w,j})</td>
<td>([2.7])</td>
</tr>
<tr>
<td>(e_{w,j})</td>
<td>([7,15])</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 4</th>
<th>Eight problems to solve the model (Gharaei and Jolai, 2018).</th>
</tr>
</thead>
</table>
| Problem | \begin{tabular}{cccccccccc}
| number of customer & P1 & P2 & P3 & P4 & P5 & P6 & P7 & P8 & Vehicle Type \\
| 150 & 150 & 200 & 200 & 250 & 250 & 300 & 300 & |
| Cap\(v\) & 100 & 150 & 100 & 200 & 150 & 80 & 60 & 120 & A \\
| \(N_v\) & 3 & 4 & 2 & 3 & 5 & 4 & 6 & 5 & |
| \(\beta_v\) & 2.60 & 3.05 & 2.60 & 3.37 & 3.05 & 2.11 & 1.84 & 2.67 & |
| Cap\(v\) & 160 & 200 & 160 & 250 & 200 & 120 & 120 & 160 & B \\
| \(N_v\) & 3 & 3 & 4 & 3 & 3 & 4 & 4 & 5 & |
| \(\beta_v\) & 3.13 & 3.37 & 3.13 & 3.70 & 3.37 & 2.67 & 2.67 & 3.13 & |
| Cap\(v\) & 300 & 250 & 300 & 250 & 300 & 150 & 200 & 200 & C \\
| \(N_v\) & 3 & 3 & 3 & 3 & 3 & 3 & 2 & 4 & |
| \(\beta_v\) & 3.75 & 3.70 & 3.75 & 3.70 & 3.05 & 3.37 & 3.37 & 3.37 & |
| Cap\(v\) & 300 & 350 & 300 & 200 & 250 & 300 & | \\
| \(N_v\) & 2 & 2 & 2 & 3 & 3 & 2 & |
| \(\beta_v\) & 3.75 & 4.05 & 3.75 & 3.70 & 3.70 & 3.75 & |
| Cap\(v\) & 400 & 250 & 300 & 350 & | \\
| \(N_v\) & 2 & 1 & 2 & 1 & |
| \(\beta_v\) & 4.14 & 3.70 & 3.75 & 4.05 & |
| Cap\(v\) & 350 & 350 & 400 & | \\
| \(N_v\) & 1 & 2 & 1 & |
| \(\beta_v\) & 4.05 & 4.05 & 4.14 & |
| number of time window & 3 & 3 & 2 & 4 & 5 & 5 & |
using a computer equipped with an Intel® CPU capable of a maximum 2.4 GHz frequency along with 4 gigabytes of main memory.

5. Results

In this section, the validity of the proposed metaheuristic methods is comprehensively assessed. For this purpose, a number of experiments were conducted.

### Table 5

<table>
<thead>
<tr>
<th>$\lambda_{tw}$</th>
<th>Time window</th>
<th>Value of dissatisfaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_5$</td>
<td>Customer group</td>
<td>Coefficient of Customer priority</td>
</tr>
</tbody>
</table>

### Table 6

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
<th>Parameters</th>
<th>Value</th>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N_{\text{pop}}$</td>
<td>50</td>
<td>$N_{\text{pop}}$</td>
<td>50</td>
<td>$N^\text{init}$</td>
<td>40</td>
</tr>
<tr>
<td>$\omega$</td>
<td>0.6</td>
<td>$\psi_1$</td>
<td>0.2</td>
<td>$\beta$</td>
<td>1</td>
</tr>
<tr>
<td>$c_1$</td>
<td>1</td>
<td>$\sigma$</td>
<td>0.12</td>
<td>$Q$</td>
<td>10</td>
</tr>
<tr>
<td>$c_2$</td>
<td>1.25</td>
<td>$\chi$</td>
<td>0.5</td>
<td>$\mu$</td>
<td>50</td>
</tr>
</tbody>
</table>

### Table 7

<table>
<thead>
<tr>
<th>Problem</th>
<th>RNI</th>
<th>MID</th>
<th>SP</th>
<th>MD</th>
<th>GD</th>
<th>ACT (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>0.002</td>
<td>783</td>
<td>315</td>
<td>6</td>
<td>1995982</td>
<td>55</td>
</tr>
<tr>
<td>P2</td>
<td>0.001</td>
<td>551</td>
<td>421</td>
<td>5</td>
<td>884035</td>
<td>18</td>
</tr>
<tr>
<td>P3</td>
<td>0.000</td>
<td>1756</td>
<td>191</td>
<td>6</td>
<td>6158077</td>
<td>63</td>
</tr>
</tbody>
</table>

### Table 8

<table>
<thead>
<tr>
<th>Problem</th>
<th>RNI</th>
<th>MID</th>
<th>SP</th>
<th>MD</th>
<th>GD</th>
<th>ACT (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>0.005</td>
<td>383</td>
<td>311</td>
<td>56</td>
<td>717712</td>
<td>56</td>
</tr>
<tr>
<td>P2</td>
<td>0.005</td>
<td>387</td>
<td>484</td>
<td>19</td>
<td>544074</td>
<td>54</td>
</tr>
<tr>
<td>P3</td>
<td>0.007</td>
<td>292</td>
<td>138</td>
<td>29</td>
<td>714826</td>
<td>74</td>
</tr>
<tr>
<td>P4</td>
<td>0.004</td>
<td>328</td>
<td>57</td>
<td>36</td>
<td>352820</td>
<td>36</td>
</tr>
</tbody>
</table>

### Table 9

<table>
<thead>
<tr>
<th>Problem</th>
<th>RNI</th>
<th>MID</th>
<th>SP</th>
<th>MD</th>
<th>GD</th>
<th>ACT (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>0.002</td>
<td>815</td>
<td>105</td>
<td>8</td>
<td>816533</td>
<td>96</td>
</tr>
<tr>
<td>P2</td>
<td>0.015</td>
<td>941</td>
<td>180</td>
<td>5</td>
<td>8523402</td>
<td>146</td>
</tr>
<tr>
<td>P3</td>
<td>0.012</td>
<td>341</td>
<td>670</td>
<td>12</td>
<td>108201</td>
<td>141</td>
</tr>
<tr>
<td>P4</td>
<td>0.008</td>
<td>401</td>
<td>811</td>
<td>13</td>
<td>342449</td>
<td>230</td>
</tr>
<tr>
<td>P5</td>
<td>0.025</td>
<td>710</td>
<td>840</td>
<td>9</td>
<td>4688380</td>
<td>231</td>
</tr>
</tbody>
</table>

### Table 10

<table>
<thead>
<tr>
<th>Problem</th>
<th>RNI</th>
<th>MID</th>
<th>SP</th>
<th>MD</th>
<th>GD</th>
<th>ACT (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>0.008</td>
<td>257</td>
<td>684</td>
<td>26</td>
<td>365561</td>
<td>52</td>
</tr>
<tr>
<td>P2</td>
<td>0.008</td>
<td>321</td>
<td>351</td>
<td>22</td>
<td>469563</td>
<td>50</td>
</tr>
<tr>
<td>P3</td>
<td>0.004</td>
<td>514</td>
<td>110</td>
<td>23</td>
<td>1230590</td>
<td>61</td>
</tr>
<tr>
<td>P4</td>
<td>0.008</td>
<td>300</td>
<td>67</td>
<td>38</td>
<td>349966</td>
<td>57</td>
</tr>
<tr>
<td>P5</td>
<td>0.006</td>
<td>376</td>
<td>328</td>
<td>34</td>
<td>741514</td>
<td>87</td>
</tr>
<tr>
<td>P6</td>
<td>0.008</td>
<td>321</td>
<td>154</td>
<td>40</td>
<td>447231</td>
<td>101</td>
</tr>
<tr>
<td>P7</td>
<td>0.006</td>
<td>325</td>
<td>207</td>
<td>42</td>
<td>856959</td>
<td>105</td>
</tr>
<tr>
<td>P8</td>
<td>0.004</td>
<td>458</td>
<td>228</td>
<td>30</td>
<td>1420370</td>
<td>72</td>
</tr>
</tbody>
</table>

### Table 11

<table>
<thead>
<tr>
<th>Problem</th>
<th>RNI</th>
<th>MID</th>
<th>SP</th>
<th>MD</th>
<th>GD</th>
<th>ACT (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>0.014</td>
<td>122</td>
<td>209</td>
<td>31</td>
<td>81161</td>
<td>114</td>
</tr>
<tr>
<td>P2</td>
<td>0.029</td>
<td>341</td>
<td>1916</td>
<td>12</td>
<td>4292901</td>
<td>99</td>
</tr>
<tr>
<td>P3</td>
<td>0.028</td>
<td>651</td>
<td>1512</td>
<td>10</td>
<td>4010341</td>
<td>166</td>
</tr>
<tr>
<td>P4</td>
<td>0.006</td>
<td>847</td>
<td>2490</td>
<td>6</td>
<td>2344107</td>
<td>389</td>
</tr>
<tr>
<td>P5</td>
<td>0.010</td>
<td>1207</td>
<td>2531</td>
<td>5</td>
<td>7122417</td>
<td>228</td>
</tr>
<tr>
<td>P6</td>
<td>0.014</td>
<td>1242</td>
<td>290</td>
<td>8</td>
<td>4523021</td>
<td>319</td>
</tr>
<tr>
<td>P7</td>
<td>0.009</td>
<td>928</td>
<td>1148</td>
<td>9</td>
<td>12634280</td>
<td>318</td>
</tr>
<tr>
<td>P8</td>
<td>0.020</td>
<td>1147</td>
<td>1827</td>
<td>10</td>
<td>4655573</td>
<td>289</td>
</tr>
</tbody>
</table>
of evaluation criteria and experimental problems were defined in the previous section at various medium and large scales, and then the performance of the MOPSO algorithm was evaluated in comparison with the NSGAII and MOACO algorithms. To evaluate the algorithms correctly, the performance metrics introduced in the previous section are used and the results are presented in Tables 7 and 8. In order to measure the impact of carrying out local search, the results are compared which the algorithms achieved, first in the absence of local search, and then in its presence. Finally, a combinatorial metric WM was used to provide an accurate description of the results, presented in Table 9.

As demonstrated in Table 7 and Fig. 1, in absence of local search, MOACO generates a better output and at various sizes in terms of RNI and the difference is significant in small sizes. NSGAII ranks second in this regard. In terms of MID, however, NSGAII generates a more optimal solution in comparison with MOACO and MOPSO. In terms of SP, algorithms with smaller dimensions have a better Pareto front distribution; therefore, NSGAII and MOPSO perform better. MOPSO and MOACO are able to generate better solutions than NSGAII does when it comes to MD which shows the range of available solutions. As mentioned previously, lower values of the GD index are more optimal and, as may be seen in the figure below, MOPSO and NSGAII are able to generate consistent and varying solutions as regards this index. Finally, in terms of ACT, NSGAII and MOPSO were able to solve the problem more quickly than MOACO was.

To assess the efficiency of local search, it was applied for the

### Table 9
Comparison of weighted metric.

<table>
<thead>
<tr>
<th>Problem</th>
<th>Weighted Metric Without local search</th>
<th>Weighted Metric With local search</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MOPSO      NSGAII      MOACO</td>
<td>MOPSO      NSGAII      MOACO</td>
</tr>
<tr>
<td>P1</td>
<td>363117   66651     14843</td>
<td>1055908   130640     148647</td>
</tr>
<tr>
<td>P2</td>
<td>160915   85510     780811</td>
<td>1213760   99094      1549941</td>
</tr>
<tr>
<td>P3</td>
<td>620017   223872    729578</td>
<td>1024271   130065     190793</td>
</tr>
<tr>
<td>P4</td>
<td>465212   63714     426880</td>
<td>1208534   64232      62528</td>
</tr>
<tr>
<td>P5</td>
<td>52038    134971    1295707</td>
<td>759026    110716     852758</td>
</tr>
<tr>
<td>P6</td>
<td>284127   81427     822705</td>
<td>1149033   167866     142074</td>
</tr>
<tr>
<td>P7</td>
<td>429510   155934    1497575</td>
<td>450573    92968      33564</td>
</tr>
<tr>
<td>P8</td>
<td>564313   258393    847136</td>
<td>396755    99895      300478</td>
</tr>
</tbody>
</table>

Fig. 1. Compare of algorithms performance measurement without local search.
above-mentioned two algorithms and the results are listed in Table 8 and Fig. 2. As with the situation with local search, MOACO generates a better output and at various sizes in terms of RNI and the difference is significant in small sizes. NSGAII ranks second in this regard. In terms of MID, however, NSGAII generates a more optimal solution in comparison with MOACO and MOPSO. In terms of SP, algorithms with smaller dimensions have a better Pareto front distribution; therefore, NSGAII and MOPSO perform better. MOPSO and MOACO are able to generate better solutions than NSGAII does when it comes to MD which shows the range of the available solutions. Lower values of the GD index, as was previously mentioned, are more optimal and the tables below indicate that MOPSO and NSGAII are able to generate consistent and varying solutions as regards this index. Finally, in terms of ACT, NSGAII and MOPSO were able to solve the problem in a shorter time compared to MOACO.

Based on Table 8 and Fig. 3, NSGAII is superior to the other two algorithms in terms of WM. The lower the value of this criterion the more optimal.

Statistical analysis was used to verify and compare the algorithms more accurately. To better visualize the distribution of the number of obtained solutions, box plot diagrams are presented. Box plots can be useful to display differences between populations without making any hypothesis of the fundamental statistical distribution: they are nonparametric. A box plot may also demonstrate the outliers, if any, which appear in circle and are robust statistics compared to the mean or the standard deviation. It can be seen that the values of mean and median are very close to each other for the most test problems for NSGAII and MOPSO. The results based on the Performance measurement for each test problem 10 replications are summarized by box plots in Fig. 4.

As may be seen in the dispersion diagram, when it comes to the RNI criterion, the algorithm MOACO is more dispersed compared to MOPSO and NSGAII, and according to the box plot diagram, MOACO’s median value is also higher compared to the other two algorithms. Moreover, the value of RNI is almost equal in both MOPSO and NSGAII, pointing to the similar performance of the two algorithms. Given the observations above, MOACO seems to be the optimal algorithm. However, in terms of the MID criterion, the results of MOACO and MOPSO are more dispersed. Therefore, according to the dispersion and box plot diagrams, NSGAII has done better than the other two algorithms. When it comes to the SP criterion, the dispersion and median values obtained for NSGAII and MOPSO are almost the same and, keeping in mind that the lower the value of SP the better, the two algorithms have performed
Fig. 2. Compare of algorithms performance measurement with local search.
Fig. 2. (continued)
better than MOACO. Finally, in terms of the MD, GD, and ACT criteria, it was observed that NSGAII, MOPSO, and NSGAII performed better respectively, indicating the superior performance of the two algorithms in this regard. Based on the results of performance criteria as well as dispersion and box plot diagrams, NSGAII appears to have performed better in solving the problem compared to MOPSO and MOACO.

6. Conclusion

In this study, an integrated problem of supply chain scheduling with assignment of due dates, order scheduling on a single machine in the production system, batch delivery, assignment to a fleet of delivery vehicles which were heterogeneous in terms of capacity, and finally, delivery to customers in time windows was presented. Given the currently worrying situation of global warming, the majority of large organizations have begun to invest more time and resources on the minimization of energy consumption and contaminant emissions. In order to actualize this goal, the routing arrangements adopted for transportation vehicles must be properly optimized by reducing costs (fixed and variable), fuel consumption and carbon emission, to name a few. On this basis, the present study also investigates the EHFFVRP (emission-based heterogeneous fixed-fleet vehicle routing problem), where fuel consumption and contaminant emission are minimized. The programming model for this problem was a mathematical mixed-integer nonlinear one. Adopting the findings of this research could lead to reduction in distribution costs across the supply chain as well as in delivery tardiness, with the latter certain to enhance customer satisfaction and reduce the environmental risks by the minimization of costs, contaminant emissions, and use of hazardous raw materials. Moreover, this study may help reduce inventory and holding costs. Where the customers first determine their time windows and state the priority of each, then the retailer categorizes the customers based on their purchase record. Finally, by solving the proposed model, the best possible route for distribution i.e. delivery of the orders is identified. In order to evaluate and validate the proposed model, the metaheuristic algorithms MOPSO and NSGAII and MOACO were compared in numerical experiments, while a number of performance criteria and statistical analysis were used to evaluate the solutions to multi-objective problems. The results indicate that NSGAII shows better overall performance.

The problems related to integrated decision-making on supply chain scheduling and its models are highly diverse. For this reason, employing new hypotheses may help develop existing models and bring them closer to real-world conditions. The present study may be improved or expanded upon by the following:

- All the parameters in the proposed model are considered as certain while, in practice, some parameters (e.g., vehicles’ starting time) have an uncertain nature. On this basis, it is suggested that in future research this factor be addressed by adopting an optimization approach in uncertain environments.
- Other metaheuristic algorithms than the ones used in this study may be employed to solve the presented model and to compare it with the proposed method.
- Several time periods can be considered for the presented problem.

![Fig. 3. Compare algorithm based on performance measurement.](image-url)
Fig. 4. Box plots based on the performance measure.
Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Maliheh Ganji: Formal analysis, Writing - original draft. Seyed Mojtaba Sajadi: Formal analysis.

References

Goksal, F.P., Karaoğlan, I., Altiparmak, F., 2013. A hybrid discrete particle swarm optimization for vehicle routing problem with simultaneous pickup and

Fig. 4. (continued).