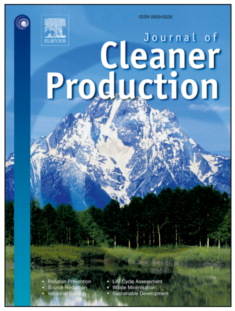


Multi-Objective Optimization for the Reliable Pollution-Routing Problem with Cross-Dock Selection using Pareto-based Algorithms

Erfan Babaee Tirkolaee, Alireza Goli, Amin Faridnia, Mehdi Soltani, Gerhard-Wilhelm Weber

PII: S0959-6526(20)32972-3

DOI: https://doi.org/10.1016/j.jclepro.2020.122927

Reference: JCLP 122927

To appear in: *Journal of Cleaner Production*

Received Date: 31 July 2019

Revised Date: 14 June 2020

Accepted Date: 15 June 2020

Multi-Objective Optimization for the Reliable Pollution-Routing Problem with Cross-Dock Selection using Pareto-based Algorithms

Erfan Babaee Tirkolaee[a,b], Alireza Goli[c], Amin Faridnia[d]

Mehdi Soltani[e], Gerhard-Wilhelm Weber[f,g]

[a]Department of Industrial Engineering, Mazandaran University of Science and Technology, Babol, Iran
e.babaee@ustmb.ac.ir
[b]Department of Industrial and Systems Engineering, Istinye University, Istanbul, Turkey
[c]Department of Industrial Engineering, Yazd University, a.goli@stu.yazd.ac.ir
[d]Department of Civil Engineering, Islamic Azad University Central Tehran Branch, Tehran, Iran, amin.faridnia@yahoo.com
[e]Department of Industrial & Mechanical Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran,
mehdisoltani618@gmail.com
[d]Faculty of Engineering Management, Poznan University of Technology, Poznan, Poland, gerhard.weber@put.poznan.pl
[g]Institute of Applied Mathematics, Middle East Technical University, Ankara, Turkey

Author statement

Erfan Babaee Tirkolaee: Conceptualization, Methodology and Supervision, **Alireza Goli**: Visualization, Formal analysis and Software, **Amin Faridnia:** Investigation and Resources, **Mehdi Soltani**: Writing-Original draft preparation. **Gerhard-Wilhelm Weber**: Project administration and Writing- Reviewing and Editing.

Multi-Objective Optimization for the Reliable Pollution-Routing Problem with Cross-Dock Selection using Pareto-based Algorithms

Abstract

Cross-docking practice plays an important role in improving the efficiency of distribution networks, especially, for optimizing supply chain operations. Moreover, transportation route planning, controlling the Greenhouse Gas (GHG) emissions and customer satisfaction constitute the major parts of the supply chain that need to be taken into account integratedly within a common framework. For this purpose, this paper tries to introduce the reliable Pollution-Routing Problem with Cross-dock Selection (PRP-CDS) where the products are processed and transported through at least one cross-dock. To formulate the problem, a Bi-Objective Mixed-Integer Linear Programming (BOMILP) model is developed, where the first objective is to minimize total cost including pollution and routing costs and the second is to maximize supply reliability. Accordingly, sustainable development of the supply chain is addressed. Due to the high complexity of the problem, two well-known meta-heuristic algorithms including Multi-Objective Simulated-annealing Algorithm (MOSA) and Non-dominated Sorting Genetic Algorithm II (NSGA-II) are designed to provide efficient Pareto solutions. Furthermore, the ε-constraint method is applied to the model to test its applicability in small-sized problems. The efficiency of the suggested solution techniques is evaluated using different measures and a statistical test. To validate the performance of the proposed methodology, a real case study problem is conducted using the sensitivity analysis of demand parameter. Based on the main findings of the study, it is concluded that the solution techniques can yield high-quality solutions and NSGA-II is considered as the most efficient solution tool, the optimal route planning of the case study problem in delivery and pick-up phases is attained using the best-found Pareto solution and the highest change in the objective function occurs for the total cost value by applying a 20% increase in the demand parameter.

Keywords: Pollution-Routing Problem; Cross-Dock Selection; Reliability; Sustainable development; Taguchi design method; NSGA-II.

1. Introduction

Supply chain management can reduce the operations and logistics costs in different industries. Therefore, supply chain and logistics management in the current market is a basic task of operational managers, because customers need quick and easy access to various products with high quality. In order to offer products to customers effectively, minimizing costs for each member of the supply chain is not sufficient. Top and bottom members of the supply chain should be considered in minimizing cost in integrated form, because minimizing total system cost improves general performance of the system and products are offered to customers better (Chen et al., 2006). Using cross-docking is one of the methods which results in integration among upstream and downstream members of the supply chain. In fact, cross-docks are designed to facilitate the movement of material which finally results in better service level and faster interaction throughout the supply chain at a low satisfactory value of cost. Today, a large number of companies employ cross-docking techniques to optimize supply chain operations (Van Belle et al., 2012). A cross-dock center should not be considered a storage point because products are stored only for the integration process for a short period (less than 24 hours): then products are transported to customers according to predetermined destinations. It should be mentioned that cross-dock is usually used for productive corporate and retailers that perform transportation activities on large networks.

Figure 1 shows common flow at a cross-dock center including 3 input vehicles and 2 output vehicles, where vehicles bring products from various pick-up locations to this cross-dock and unload it. Products are collected in the cross-dock (which is specified by different colors). Then, these products are merged into 2 or 3 packages. In this example, the products recognized in green color are merged into two kinds of packages simultaneously. In other words, the products are merged for effective delivery and sorted based on destination. Afterward, they are loaded through cross-dock to output vehicles with minimal storage in between (Abad et al., 2018).

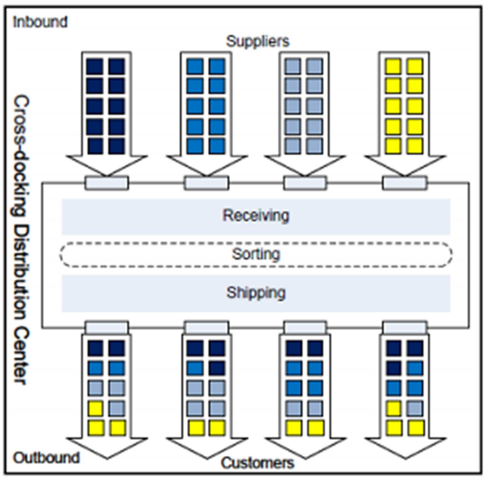


Fig. 1. Cross-dock generic flows.

What can be inferred from Figure 1 is that unlike conventional storages, cross-docking aims to reduce or eliminate the cost of holding existing inventory and shortening the delivery time from suppliers to retailers. However, this paper addresses Cross-Dock Selection (CDS) problem which was first introduced and considered with Vehicle Routing Problem (VRP) by Maknoon and Laporte (2017) with the aims of making optimal decisions on route constructions and freight consolidation at cross-docks. In other words, selecting cross-docks as freight consolidation points leads to gain economies of scale. In the CDS problem, the products are processed and transported to the customers by at least one cross-dock. It has a vast application in production and retail companies as well as logistic service providers that handle various shipments on large networks.

In order to implement the route planning of vehicles using these cross-docks, a Pollution-Routing Problem (PRP) is introduced by considering traffic conditions and vehicle speed as two important factors to determine the amounts of fuel consumption and pollution in the delivery section, which is a developed version of VRP with CDS (VRP-VCDS). Therefore, in this paper, a multi-period single-product PRP with CDS (PRP-CDS) is proposed. Considering Hard Time Windows (HTW) constraint in delivering the product to customer using homogenous vehicles, a Bi-Objective Mixed Integer Linear Programming (BOMILP) model is proposed, where the first objective function to be minimized represents the total costs of the network (including pollution costs depending on vehicles features, speed-related costs in heavy traffic and free speed, speed-related costs in the light traffic condition, delivery drivers' wage and travelling costs of the route in the pick-up section). The second objective function is to maximize the supply reliability which leads to customers' satisfaction maximization. Therefore, sustainable development is addressed through these objectives in the problem (Mardani et al., 2020). To solve the designed BOMILP model, two meta-heuristic algorithms including Multi-objective Simulated-annealing Algorithm (MOSA) and Non-dominated Sorting Genetic Algorithm II (NSGA-II) are presented. The key advantage of these two algorithms is that they are capable of finding multiple solutions at the Pareto front. On the other hand, the structure of these two algorithms is quite different. In the MOSA algorithm, only one solution is evaluated in each iteration, but in the NSGA-II, a set of solutions is evaluated in each iteration. The main reason for developing these algorithms is their relative superiority over other algorithms in the literature.

After adjusting parameters using the Taguchi method, the proposed algorithms are implemented on several numerical examples. The corresponding results are compared based on Mean Ideal Distance (MID) measure, Spacing Metric (SM), Diversification Metric (DM) and CPU Time (CPU-T), which shows the efficiency of the proposed algorithms in solving PRP-CDS.

In summary, the innovative contributions of this work are described as follows.

- Developing a novel mathematical model in order to integrate PRP and CDS which results in a comprehensive problem with a great application in industries,
- Making the problem closer to the real-world conditions by considering the effects of traffic in transportation time and calculating pollution dependent on traffic time,
- Investigating the total cost minimization related to pollution and transportation (1st objective) and reliability maximization (2nd objective) through cross-docking simultaneously,
- Designing high-quality algorithms including NSGA-II and MOSA in order to solve the large-sized problem efficiently,
- Testing the performance of the proposed solution techniques based on different comparative measures and a statistical test,
- Evaluating the applicability and performance of the proposed methodology by investigating a case study in Iran.

The rest of this paper is organized as follows. Section 2 reviews the literature regarding the proposed problem with a specific attention to the application of CDS in routing problems. Section 3 describes the developed model step by step. The suggested solution techniques including the ε-constraint approach and NSGA-II and MOSA are introduced in Section 4.

Section 5 presents the implementation of the Taguchi design method and introduces comparison metrics of the solution techniques. Section 6 provides the numerical and simulation results of the solution approaches using comparison metrics. The real case study problem is investigated in Section 7. Finally, Section 8 concludes the research and offers some suggestions for future research and application.

2. Literature Review

In this section, considering that VRP-CDS is close to the proposed problem, the main focus is on relevant studies and the research gaps are specified. Cross-docking approach was first proposed in 1930 by the United States (US) transportation industry, then it was used in 1980 in Walmart retail industry (Stalk et al., 1992). Using cross-dock terminals, Walmart introduces cheap strategies every day and eliminates all inventory maintenance costs. Hammer et al. (2004) indicated that good customer service of Walmart is the result of the effective implementation of cross-docking. Thereafter, using cross-docking was accepted by companies and researchers. The reported strategy was used in different industries including retail, construction, automobile manufacturing, imaging, aviation, mail service, etc. A cross-docking program might be seen as a mail distribution center in which mails and packets are collected from different mail centers to be sorted and guided to their destination. Among successful implementations of cross-docking, the research performed by Kinnear (1997) and Napolitano (2011) can be mentioned. It should be underlined that more than 85% of all publications on cross-docking were made after 2004 (Boysen and Fliedner, 2010). Considering the inherent complexity of designing cross-dock terminals, several stages of decision-making problems were evaluated in the literature. To this end, interested readers are referred to investigate some new reviews including a wide variety of cross-docking concepts. For instance, Boysen and Fliender (2010) investigated the truck scheduling in cross-docking system and have suggested a classification scheme based on operational features and objectives. Future research challenges were specified by Stephan and Boysen (2011), while instructions for executions and effective employment of cross-docking were studied by Van Belle et al. (2010). Buijs et al. (2014) presented 24 unique decision-making problems to cover a range of cross-docking designs. Among all these problems, there is a class of VRP-CDS.

Santos et al. (2013) presented an IP formula for pick-up and delivery at cross-dock aiming to minimize the cost of optimal routes and load cost. They also proposed a heuristic algorithm which reduces routing cost between 3.1% and 7.7%. Transportation service using several cross-docks along with Soft Time Windows (STW) and HTW was studied by Miao et al. (2012). That is, in addition to HTW, there is a more limited time window called STW. The proposed single-objective model aims to minimize total distribution costs including transportation cost, maintenance cost, and penalty. Moreover, two meta-heuristic algorithms including adaptive TS and adaptive Genetic Algorithm (GA) were proposed which outperform CPLEX in terms of efficiency.

Moghadam et al. (2014) have studied a VRP-CDS in a network comprising several suppliers, several customers, and a cross-dock. They proposed a Mixed Integer Nonlinear Programming (MINLP) formula to minimize total travelling cost and time spent in cross-docking for loading and unloading products. Simulated Annealing (SA) and Ant Colony System-SA (ACS-SA) algorithms were developed to solve the proposed model, where empirical results show superior performance of ACS-SA compared to SA. A Mixed-Integer Linear Programming (MILP) model was developed for open VRP with cross-dock aiming to minimize vehicle employment cost and transportation cost by Vincent et al. (2016). To solve the problem, the SA algorithm was tested on three sets of test problems; the obtained results were compared with those of CPLEX showing the better efficiency of SA in terms of objective function and computation time. An efficient TS algorithm was offered by Nikolopoulou et al. (2017) for VRP-CDS, improving many of the best-known results in this context. A MILP formula was proposed by Wang et al. (2017), which considers several cross-docks and time window limitations to minimize the fixed cost of vehicles (e.g., minimizing the number of used vehicles and total travelling cost). Two construction heuristic-tow layer SA and construction heuristic-two layer TS algorithms were proposed to solve the problem; both were compared with CPLEX. Maknoon and Laporte (2017) introduced VRP-CDS for the first time. Considering several cross-docks and time window limitations, the objective of their model was to find a set of routes to minimize cost for serving other customers. In addition, a heuristic algorithm was proposed which was compared with CPLEX. Recently, Nasiri et al. (2018) simultaneously considered CS and a supplier selection in a single-objective MILP model for the VRP-CDS problem. The suggested problem was solved using a two-stage solution algorithm and compared with exact solutions.

On the other hand, some main research was done on developing an uncertain environment for the problem. Rajabi and Shirazi (2016) investigated the truck scheduling problem in the cross-docks with uncertainty in both inbound and outbound problems. Mousavi and Vahdani (2017) considered the design of cross-docking systems under uncertainty in a model that consists of two phases: (1) a strategic-based decision-making process for selecting the location of cross-docks to operate, and (2) an operational-based decision-making process for vehicle routing scheduling with multiple cross-docks.

In terms of considering reliability in designing supply chain networks, Pasandideh et al. (2015) proposed a MILP model to design an optimal bi-objective three-echelon supply chain network considering multiple products, multiple periods and the reliability of warehouses. Their proposed objectives were to minimize total cost and maximize the average number of products dispatched to customers simultaneously. They applied six exact Multi-objective Decision-Making (MODM) methods to provide a single objective for the problem. Furthermore, Razmi et al. (2017) suggested a bi-objective stochastic mathematical model to redesign a reliable warehouse network. Their proposed objectives were to minimize average costs and maximize coverage percent of customer demand delivered. They solved the model using the augmented ε-constraint method. Recently, Tirkolaee et al. (2020a) developed a hybrid technique based on Multi-Criteria Decision-Making (MCDM) methods and Multi-Objective Mixed-Integer Linear Programming (MOMILP) approach to tackle a sustainable-reliable supplier selection in two-echelon supply chain network. They concurrently studied the minimization of total cost, maximization of weighted value of purchased products and maximization of supply chain reliability.

Table 1 presents a summary of the literature regarding VRP-CDS and compares the problems existing in the literature with the problem proposed in this paper. The only column which should be described is RR/SS which shows eligibility for vehicle routes between retailers and suppliers.

Table 1. Summary of literature regarding VRP-CDS and its different variants.

References	Year	Cross-dock selection	Multiple cross-docks	Type of vehicle		Time windows	Capacity in cross-docks		RR/SS	Inventory storage	Pollution issues	Multiple objectives	Solution Method			
				Homogeneous	Heterogeneous		Limited	Unlimited					Software	Exact	Heuristic	Meta-heuristic
Sung and Song	2003		*		*			*							*	
Chen et al.	2006		*	*		*	*			*			*		*	
Lee et al.	2006			*				*	*						*	

References	Year	Cross-dock selection	Multiple cross-docks	Type of vehicle		Time windows	Capacity in cross-docks		RR/SS	Inventory storage	Pollution issues	Multiple objectives	Solution Method			
				Homogeneous	Heterogeneous		Limited	Unlimited					Software	Exact	Heuristic	Meta-heuristic
Sung and Yang	2008		*		*			*					*	*		
Wen et al.	2009			*		*		*	*						*	
Liao et al.	2010			*				*	*						*	
Musa et al.	2010		*	*				*					*			*
Dondo et al.	2011		*		*	*		*	*	*			*			
Ma et al.	2011		*	*		*		*		*			*		*	
Miao et al.	2012		*	*		*	*			*			*			*
Dondo and Cerda	2013			*		*		*	*			*			*	
Santos et al.	2013			*				*	*						*	
Dondo and Cerdá	2014			*		*		*	*						*	
Moghadam et al.	2014			*		*		*	*							*
Morais et al.	2014			*		*		*	*	*					*	
Ahmadizar et al.	2015		*		*		*		*	*						*
Vincent et al.	2016			*				*	*				*			*
Nikolopoulou	2017			*		*		*	*							*
Wang et al.	2017		*		*	*		*	*	*			*		*	
Maknoon and Laporte	2017	*	*	*		*		*	*				*		*	
Nasiri et al.	2018	*	*		*	*	*		*	*				*		*
Rahbari et al.	2019		*		*	*		*	*	*		*	*	*		
Baniamerian et al.	2019				*	*		*								*
Current research	-	*	*	*		*		*	*		*	*	*	*		*

Since cross-docks and related transportations have great roles in increasing the efficiency of large-scale supply chain distribution networks, considering the other real-world important factors such as reliability and pollution is attractive in designing or redesigning this network. Reliability maximization leads to customer satisfaction maximization and earning a lot of market share, and pollution minimization leads to creating eco-friendly industries. Therefore the main limitation in the literature and the main contribution of this research can be summarized as follows:

- Considering traffic conditions in the urban area and calculating the routing and pollution costs based on current traffic conditions,
- Integrating pollution costs minimization and customers' satisfaction (reliability) maximization in line with CDS and routing optimization,
- Appling two efficient meta-heuristic algorithms including MOSA and NSGA-II in order to generate Pareto optimal solutions for the PRP-CDS.

3. Problem Statement

In this section, the assumptions and limitations of the proposed problem are explained. Furthermore, the components of the mathematical model are presented. The main objective of the problem is to fulfill the customers' demands for products through a cross-docking system. To get closer to the real-world situations, traffic conditions and its possible effects in urban areas, pollution and the related costs in the supply chain are included.

The network studied in this research includes a graph $G = (V, E)$, V presents network points and E presents edges of the network. Points of this network are divided into three groups C, P and D, such that they represent the existing cross-docks, set of suppliers and set of customers, respectively. The purpose of this network is to collect a specific volume of products in cross-dock from suppliers (pick-up phase) and delivering them to various customers (delivery phase) based on demands. The role of cross-dock in this network is to collect and deliver. High-volume products are received from suppliers, taken to cross-dock and delivered to customers in small volumes.

Different suppliers are in industrial parks and suburban areas; to supply products, a homogeneous transportation fleet with high capacity is used. Different customers are present in urban areas and a lower capacity fleet is used to deliver products to them. Transportation fleet in the pick-up phase is independent of the delivery phase transportation fleet. Figure 2 shows an example of supply scheduling through cross-docks studied in this research.

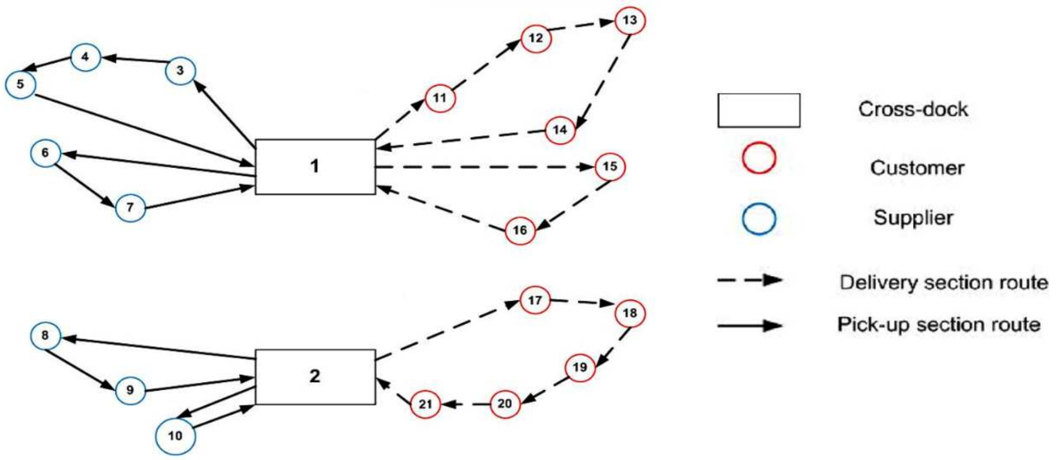


Fig. 2. An example of a supply based on studied cross-docks.

Each customer has an HTW for product delivery; thus, the delivery policy is to use the delivery fleet maximally. Considering different customers in urban areas and their HTW, traffic condition and vehicle speed in urban areas is of great importance. The speed of vehicles in urban areas has a predetermined maximum and minimum. On the other hand, the traffic condition and vehicle speed also affect fuel consumption and pollution. To this end, traffic condition is defined as an effective parameter (m) and vehicle speed at each traffic condition as $V^1, V^2, \ldots, V^R$. Different states of this parameter and conditions are represented in Table 2.

Table 2. Traffic condition description.

m	Traffic Condition	Description
1	Heavy Traffic	In this condition, each $V^{11}, V^{12}, \ldots, V^{1R}$ can be selected for movement and vehicle continues with this speed to the end.
2	Light Traffic	In this condition, one of $V^{21}, V^{22}, \ldots, V^{2R}$ can be selected. The values of these speeds are higher than their equivalent speed in heavy traffic. Since in this condition, traffic is changing from heavy to free speed, the vehicle will also change its speed. The speed increase gradient is $\frac{V^{3r}-V^{2r}}{V^{2r}}$.
3	Free speed	In this condition, one of $V^{31}, V^{32}, \ldots, V^{3R}$ can be selected. It is obvious that the values of these speeds are higher than their equivalent in light and heavy traffic conditions. Each vehicle travels the route with that speed; after visiting the corresponding customer, it can select another speed to arrive at the next customer.

As mentioned before, the traffic condition and speed determine fuel consumption and pollution. Thus, the travelling costs along the route are calculated based on fuel consumption. Accordingly, in this research, the travelling cost in the delivery phase is calculated using pollution cost calculation in Franceschetti et al. (2013). Considering traffic conditions, it is obvious that the time of arriving at the customer point depends on the time of moving from cross-dock and vehicle speed. Thus, the policy is to calculate drivers' wages based on the time they have worked. This policy is not used in the pick-up section because pick-up routes are less in heavy traffic condition, and their wage is paid only based on the travelling route. Hence, the main novelty of this paper is to propose an optimal approach for pick-up and delivery through a cross-docking network considering traffic conditions and resulting pollution caused by transportation. In addition to cost minimization, since pick-up and supply operations are performed well by the suppliers all the time, considering an objective to increase the reliability of suppliers makes the model more applicable. In other words, since there may be many possible suppliers within the network, a reliable network takes into account customer satisfaction by keeping cross-docks full of demanded products from customers. In other words, besides transportation and pollution costs minimization, reliability maximization is addressed and the proposed methodology evaluates the trade-off between these items and chooses the optimal policy. Accordingly, sustainable development is analyzed in the problem.

The most closely related and main contributed researches in the literature are Franceschetti et al. (2013) and Maknoon and Laporte (2017). Franceschetti et al. (2013) proposed a pollution routing mathematical model without cross-dock system. Maknoon and Laporte (2017) formulated a VRP-CDS problem without considering the pollution and reliability factors. Hence, the proposed mathematical model in this research is a novel mathematical formulation presenting a PRP-CDS problem considering the pollution and reliability of the supply chain.

In this model, since the capacity of each supplier is limited, waiting time till failure of supplier i at period T_i follows an exponential distribution with mean σ_{it}. This failure might be due to natural events, terroristic attacks, changing owners, faults of workers, weather conditions, etc. Therefore, the reliability of supplier i (R_i) in transporting and distributing products to cross-docks in a period is represented in Eq. (1):

$$R_i = P(T_i > \tau) = e^{-\sigma_{it}\tau} \qquad \forall i \in P, t \in T. \tag{1}$$

The average amount of products transported to cross-dock j by supplier i is equal to $e^{-\sigma_{it}.\tau} f_{ijt}$, where τ denotes the working time without any failure. Here, f_{ijt} represents the amount of products transported between points i and j at period t. Therefore, in addition to the minimization of the transportation costs which affect the amount of transported products, the reliability maximization of suppliers also determines the amount of transported products.

In summary, the assumptions of this research are as follows:

- Time horizon (planning periods) includes several time periods (e.g., days) and the main purpose of CDS is to supply products from different supplier and deliver them to different customers.
- There are K_1 homogenous vehicles with capacity Q_1 in the pick-up section.
- There are K_2 homogeneous vehicles with capacity Q_2 in the delivery section.
- All vehicles in the delivery section should be loaded up to their maximum capacity.
- Total received load at each cross-dock should be delivered to the customers at the end of the time horizon.
- Capacity of cross-docks for receiving products is infinite.
- All vehicles cannot move between cross-docks. In other words, vehicles are only allowed to deliver or pick up from exactly one cross-dock.
- Pick-up drivers' wage is calculated based on the traversed route.
- Delivery drivers' wage is paid based on starting time and duration of traversing the route.
- Each supplier has a specific delivery capacity.
- Each supplier and customer should be met only once.
- As the time period is very short (as days) the customers' demand for each period can be obtained exactly and is employed as a certain parameter in the model.

Here, sets, indexes, parameters and decision variables can be seen in Tables 3-5, respectively:

Table 3. Sets and indexes of the model.

Sets and indices	Description
C	Set of cross-docks,
P	Set of pick-up points (suppliers),
D	Set of delivery points (customers),
T	Set of planning period,
R	Set of vehicle speed,

Sets and indices	Description
i, j	Network points index; $i, j \in C \cup D \cup P$,
m	Traffic condition index,
r	Vehicle speed index,
t	Planning period index.

Table 4. Parameters of the model.

Parameters	Description
K_1	Number of vehicles in the pick-up phase at each period,
K_2	Number of vehicles in the delivery phase at each period,
Q_1	Capacity of each vehicle in the pick-up phase,
Q_2	Capacity of each vehicle in the delivery phase,
v^{mr}	Speed of vehicles in the delivery phase in m^{th} traffic condition under r^{th} state,
σ_{it}	Exponential distribution parameter indicating the failure rate of supplier *i* at period *t*,
q_{it}	Demand of i^{th} customer for the pick-up at t^{th} period,
$scap_i$	Capacity of i^{th} supplier for the pick-up at each period,
d_{ij}	Distance between points *i* and *j* of the whole network,
$[l_{it}, u_{it}]$	Time window of visiting i^{th} customer at t^{th} period
dc_t	Drivers' wage in delivery phase for a time unit at period *t*,
c_{ij}	Cost of transporting material between points *i* and *j* in the pick-up section,
ξ	Ratio of fuel to air,
κ	Heat caused by 1 gram of diesel fuel (Kj/g),
ψ	Liter to gram conversion rate (g/l),
Γ	Engine friction rate (Kj/l),
N_e	Engine speed (rev/s),
v	Engine volume (l),
ρ	Air pressure (Kg/m^3),
c_d	Aerodynamic traction coefficient,
A	Front surface area of delivery section vehicles,
μ	Total weight of delivery section vehicle in standard form (Kg),
ε	Exploitation rate of delivery section vehicles,
f_t^c	Cost of Co2 emissions per each liter of consumption fuel at period *t*,

Parameters	Description
αl_{ij}	Road angle factor between *i* and *j*,
ϖ	Efficiency rate of diesel vehicle in the delivery section,
a	Time which traffic condition changes to free speed,
h_{it}	Service time at i^{th} customer for pick-up at period *t*,
M	An optional large number,
λ	Fuel consumption coefficient in the delivery section vehicles,
γ	Vehicle performance coefficient in the delivery section,
β	Aerodynamic impact factor,
θ^{mr}	Speed gradient in m^{th} traffic condition under r^{th} speed,
η_{ij}^{mr}	Duration of traversing from point *i* to *j* in m^{th} traffic condition under r^{th} speed.

Table 5. Decision variables of the model.

Variables	Description
x_{ijt}	A binary variable in the delivery section equals to 1 if arc (*i*, *j*) is traversed by a vehicle at period *t* and 0, otherwise,
y_{ijt}	A binary variable in the pick-up section equal to 1 if arc (*i*, *j*) is traversed by a vehicle at period *t* and 0, otherwise,
z_{ijt}^{mr}	A binary variable equals to 1 if a vehicle in the delivery section selects speed level *r* at period *t* from point *i* in *m* traffic condition, 0 otherwise,
f_{ijt}	Amount of product transported between points *i* and *j* from the whole network at period *t*,
w_{ijt}^{mr}	Movement time of vehicle at period *t* in the delivery section from point *i* towards *j* under traffic condition *m* and speed level *r*,
s_{it}	Total time elapsed at period *t* until arriving at point *i* as the last met point on the delivery route,
φ_{it}	Time of starting serving customer *i* at period *t*,
Lo_{it}	Total load received at i^{th} cross-docking from the pick-up section at period *t*,
au_{ij}	An integer auxiliary variable for eliminating sub-tour.

The computational equations for some parameters are presented in the Appendix. Therefore, the mathematical model of the proposed MILP problems is as follows.

3.1.Formulation of the objective functions

The sum of formulas (2.1) to (2.6) represents the first objective function of the developed mathematical model including the total network cost in all periods. Formula (2.1) reflects the pollution cost related to the vehicle's features. In this equation, $f_t^c\,\lambda\,\Gamma\,N_e v$ is the cost of consumed fuel when the vehicle is moving from points *i* to *j* by considering vehicle speed and its performance and $\theta^{mr} w_{ijt}^{mr} + \eta_{ij}^{mr} z_{ijt}^{mr}$ denotes the spend time in traveling from points *i* to *j*. This equation does not calculate the cost of CO_2 emissions in traffic. Formula (2.2) indicates the cost of CO_2 emissions in traffic and free speed conditions. In this equation, $f_t^c\,\lambda\,\gamma\,\beta\,(v^{mr})^3$ denotes the exceed cost of real fuel consumption per transportation time (especially hour) by considering Aerodynamic impact factor and vehicle performance. Formula (2.3) is similar to Formula (2.2) and indicates the cost of consumed exceed fuel in light speed traffic. Formula (2.4) calculates the pollution costs related to the root angle and weight of vehicles. In this equation $f_t^c\,\lambda\,\gamma\,\alpha l_{ij} d_{ij}$ indicates the cost of exceeding consumed fuel for root angle and per Kg and $\mu\, x_{ijt} + f_{ijt}$ is the total weight of vehicle moving from points *i* to *j*. Formula (2.5) computes total drivers' wages in the delivery section and Formula (2.6) calculates costs of traversing the route in the pick-up section.

$$\text{minimize TC} = \sum_{i\in C\cup D}\sum_{j\in C\cup D}\sum_{r\in R}\sum_{m\in\{1,2,3\}}\sum_{t\in T} f_t^c\,\lambda\,\Gamma\,N_e v\left(\theta^{mr} w_{ijt}^{mr} + \eta_{ij}^{mr} z_{ijt}^{mr}\right) \tag{2.1}$$

$$+\sum_{i\in C\cup D}\sum_{j\in C\cup D}\sum_{r\in R}\sum_{m\in\{1,3\}}\sum_{t\in T} f_t^c\,\lambda\,\gamma\,\beta\,(v^{mr})^3\left(\theta^{mr} w_{ijt}^{mr} + \eta_{ij}^{mr} z_{ijt}^{mr}\right) \tag{2.2}$$

$$+\sum_{i\in C\cup D}\sum_{j\in C\cup D}\sum_{r\in R}\sum_{t\in T} f_t^c\,\lambda\,\gamma\,\beta\,(v^{2r})^3\left(a z_{ijt}^{mr} - w_{ijt}^{2r}\right) \tag{2.3}$$

$$+\sum_{i\in C\cup D}\sum_{j\in C\cup D}\sum_{r\in R}\sum_{t\in T} f_t^c\,\lambda\,\gamma\,\alpha l_{ij} d_{ij}(\mu\, x_{ijt} + f_{ijt}) \tag{2.4}$$

$$+\sum_{i\in D}\sum_{t\in T} dc_t s_{it} \tag{2.5}$$

$$+\sum_{i\in C\cup D}\sum_{j\in C\cup D}\sum_{t\in T} c_{ij} y_{ijt} \tag{2.6}$$

Formula (3) stands for the second objective function which represents the customers' satisfaction through maximizing the average amount of products transported to cross-dock centers.

$$\text{maximize TR} = \sum_{i\in P}\sum_{j\in C}\sum_{t\in T} e^{-\sigma_{it}.\tau} f_{ijt} \quad (3)$$

3.2.Formulation of the constraints

The constraints of model are divided into two groups. Eqs. (4)-(15) are related to the delivery section and the governing routing conditions. Eqs. (16)-(23) are further related to the pick-up sections and features governing supply through the supplier.

Eq. (4) describes that in each period, the used delivery vehicles should be less than or equal to the number of available vehicles. Eqs. (5)-(6) guarantee that each customer is visited by vehicles exactly once. Eq. (7) states that delivery section vehicles do not move between different cross-docks.

$$\sum_{i\in C}\sum_{j\in D} x_{ijt} \le k_2 \quad \forall t \in T, \quad (4)$$

$$\sum_{i\in C\cup D} x_{ijt} = 1 \quad \forall j \in D, \forall t \in T, \quad (5)$$

$$\sum_{j\in C\cup D} x_{ijt} = 1 \quad \forall i \in D, \forall t \in T, \quad (6)$$

$$\sum_{i\in C}\sum_{j\in C} x_{ijt} = 0 \quad \forall t \in T, \quad (7)$$

Eq. (8) implies that the load difference of the load entered at one customer and load exiting from that customer should be equal to the demand of that customer. In other words, customer demands should be supplied during each period. Eq. (9) describes load limits transported between two points. The minimum load which enters customer *j* is equal to its demand. The maximum load transported through route traversing from point *i* to *j* is equal to the case in which the vehicle carries products in its full capacity. Here, *i* is its first customer and its demand is only supplied at point *i*. Eq. (10) indicates that vehicles should deliver all products of a cross-dock to customers.

$$\sum_{j\in D\cup C} f_{jit} - \sum_{j\in D\cup C} f_{ijt} = q_{it} \quad \forall i \in D, \forall t \in T, \quad (8)$$

$$q_{jt} x_{ijt} \le f_{ijt} \le x_{ijt}(Q_2 - q_{it}) \quad \forall i, j \in D, \forall t \in T, \quad (9)$$

$$\sum_{j\in D} f_{ijt} = Lo_{it} \quad \forall i \in C, \forall t \in T, \quad (10)$$

Eq. (11) shows the starting time from *i* to *j* after finishing serving *i*. Eq. (12) indicates a hard time window for each customer. Eq. (13) shows the return time of each customer towards existing cross-docks.

$$\sum_{i\in C\cup D}\sum_{r\in R}\sum_{m\in\{1,2,3\}} w_{ijt}^{mr} \geq \varphi_{it} + h_{it} \quad \forall i \in D, \forall t \in T, \tag{11}$$

$$l_{it} \leq \varphi_{it} \leq u_{it} \quad \forall i \in D, \forall t \in T, \tag{12}$$

$$s_{ij} \geq \sum_{r\in R}\sum_{m\in\{1,2,3\}}\sum_{j\in C}\left(w_{ijt}^{mr} + \theta^{mr} w_{ijt}^{mr} + \eta_{ij}^{mr} z_{ijt}^{mr}\right) \quad \forall i \in D, \forall t \in T, \tag{13}$$

Eq. (14) describes the relationship between the speed variable at each route and the route variable. Eq. (15) guarantees that no sub-tour is formed in the delivery section.

$$\sum_{r\in R}\sum_{m\in\{1,2,3\}} z_{ijt}^{mr} = x_{ijt} \quad \forall i,j \in C \cup D, \forall t \in T, \tag{14}$$

$$u_i + 1 \leq u_j + M\left(1 - x_{ijt}\right) \quad \forall i,j \in D, \forall t \in T, \tag{15}$$

Eqs. (16)-(17) represent that each supplier is met only during the pick-up section. Eq. (18) indicates that pick-up vehicles do not move between cross-docks. Eq. (19) indicates that total constituted routes at each period should be less than pick-up vehicles. It is not necessary to use the maximum number of vehicles in the pick-up section.

$$\sum_{i\in C\cup P} y_{ijt} = 1 \quad \forall j \in P, \forall t \in T, \tag{16}$$

$$\sum_{j\in C\cup P} y_{ijt} = 1 \quad \forall i \in P, \forall t \in T, \tag{17}$$

$$\sum_{i\in C}\sum_{j\in C} y_{ijt} = 0 \quad \forall t \in T, \tag{18}$$

$$\sum_{i\in C}\sum_{j\in P} y_{ijt} \leq K_2 \quad \forall t \in T, \tag{19}$$

Eq. (20) calculates the total load received at each cross-dock by pick-up vehicles in each period. The total load is delivered to customers using delivery vehicles. Eq. (21) describes that load which can be received from each supplier at each period is equal to the maximum capacity of that supplier. Eq. (22) considers maximum pick-up at each period from one supplier equal to the free capacity of that vehicle. Eq. (23) ensures the elimination of the possible sub-tours in pick-up routes.

$$\sum_{j \in P} f_{jit} = Lo_{it} \qquad \forall i \in C, t \in T, \tag{20}$$

$$f_{ijt} \leq scap_i \, y_{ijt} \qquad \forall i, j \in P, \forall t \in T, \tag{21}$$

$$\sum_{j \in C \cup P} f_{ijt} \leq Q_2 - \sum_{j \in C \cup P} f_{ijt} \qquad \forall i \in P, \forall t \in T, \tag{22}$$

$$au_{ij} + 1 \leq au_{ji} + M\left(1 - y_{ijt}\right) \qquad \forall i, j \in P, \forall t \in T, \tag{23}$$

4. Solution Methods

Since VRP is a well-known NP-hard problem which was presented by Dantzig and Ramser (1959), and the proposed PRP-CDS is an extended problem of VRP, PRP-CDS is known as an NP-hard problem. So, heuristic and meta-heuristic methods (like Evolutionary Algorithms (EAs)) are required to solve these problems. Many researchers have been working on similar optimization problems and the applications of novel and high-performance solution methods such as heuristics and meta-heuristics in order to optimize them (Wang et al., 2017b; Tirkolaee et al., 2019a, 2020b, 2020c; Goli et al., 2019; Dalla Torre et al., 2019). For this purpose, two efficient multi-objective meta-heuristic algorithms, namely, MOSA and NSGA-II, are presented to solve the problem. Furthermore, the ε-constraint technique is proposed to evaluate the performance of the algorithms in small-sized problems.

4.1.ε-constraint

The ε-constraint method is one of the most famous exact solution techniques for coping with multi-objective problems, which solving this kind of problems by keeping one of the objective functions as the main one and moving the other objective functions into the constraints as the subsidiary objectives (Bérubé et al., 2009).

The Pareto fronts can be generated using the ε-constraint method as it is defined in Eq. (24):

$$\begin{array}{l} \min\ (f_1(X), f_2(X), \ldots, f_n(X)) \\ \text{such that} \\ x \in X. \end{array} \quad \rightarrow \quad \begin{array}{l} \min\ f_1(X) \\ \text{such that} \\ x \in X, \\ f_2(X) \leq \varepsilon_2, \\ \ldots \\ f_n(X) \leq \varepsilon_n. \end{array} \tag{24}$$

The execution steps of the ε-constraint method are as follows:

Step 1: Modify the proposed mathematical model based on Eq. (24),

Step 2: Specify the range of ε values based on the lowest and highest possible value for the second objective function,

Step 3: Optimize the mathematical model and save the solution as one of the Pareto solutions,

Step 4: Change the ε value and go to Step 3,

Step 5: Report the Pareto optimal solutions,

The flowchart of the proposed ε-constraint method is illustrated in Figure 3.

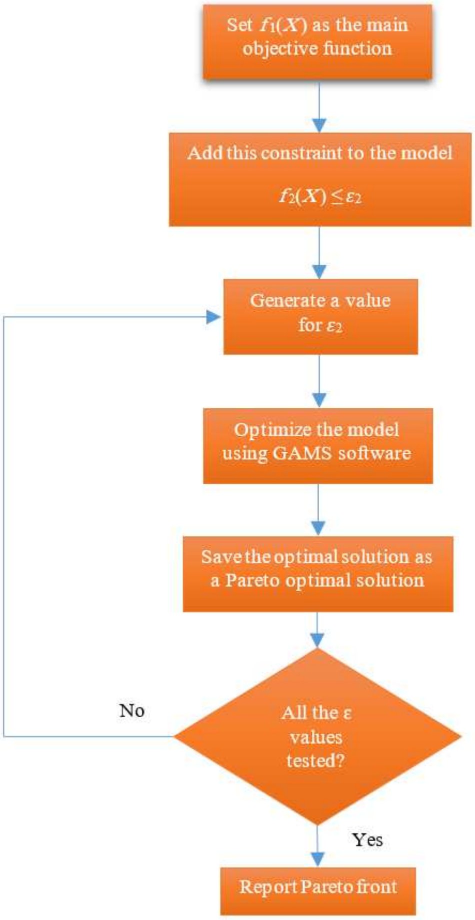


Fig. 3. Flowchart of the ε-constraint method.

According to Figure 3, in the proposed model, the first objective function is taken into account as the main objective and the second objective function is introduced as a subsidiary objective. Then, 10 breakpoints are considered for the objective functions (Tirkolaee et al., 2019a). The associated formulation of the proposed model is shown in Eq. (25):

$$\begin{aligned}&\min f_1(X)\\&\text{such that}\\&x \in X,\\&f_2(X) \geq \varepsilon_2.\end{aligned} \tag{25}$$

4.2. Non-dominated Sorting Genetic Algorithm II

Among Multi-Objective Evolutionary Algorithms (MOEA), NSGA-II is one of the most widely used optimization algorithms in multi-objective optimization contexts (Wang et al., 2018a, 2018b; Tirkolaee et al., 2019a; Sangaiah et al. 2020). The basis of NSGA-II is combining both Pareto concept and GA; therefore, it is suitable for multi-objective problems. The Non-dominated Sorting Genetic Algorithm (NSGA) was first introduced by Srinivas and Deb (1994); it divided the evolutionary group into several levels to provide a dominant relation for selection and solution. Deb et al. (2002) optimized an operational NSGA scale, such that the elite mechanism was employed instead of sharing coefficient of density function; they called this version of NSGA as NSGA-II. Along with all efficiencies of NSGA-II, it can be regarded as a formation basis of many multi-objective optimization algorithms. This algorithm and its unique method have been employed in handling multi-objective optimization problems by many researchers to generate novel multi-objective optimization algorithms. Undoubtedly, NSGA-II is one of the basic MOEAs which can be regarded as the second generation of these methods.

In order to describe NSGA-II, determining solution density in search space, selection of solution for generating the next generation is described in the following, respectively.

4.1.1 Fast Sorting Method for Searching Dominant Solution

The computational complexity of MOEA methods like NSGA for population dimension N and number of fitness functions m is equal to O(mN^3), which becomes O(mN^2) using this algorithm. It should be mentioned that this advantage is achieved by increasing storage space from O(N) to O(N^2). For each solution, two important factors are calculated: 1. domination count n_p, the number of solutions which dominate the solution p, and 2. S_p, a set of solutions that the solution p dominates. Calculating these two features is followed by O(mN^2) comparisons. Solutions with n_p=0 are the first Pareto or F_1. Now, the non-dominated set S_p is considered for each member F_i, and n_j of the j^{th} member is reduced by one unit. Solutions for which n_j=0 belong to H. after completing H for all members of F_1, it can be said that H is the second Pareto front. In the following, F_1 is set aside and H is repeated as first Pareto front for the remaining members. Figure 4 reflects the assignment of different ranks to existing solutions. Accordingly, we have several types of ranking such that the solutions related to Rank 1 are the best and have the high priority, then the solutions related to Rank 2 have the second priority, and so on.

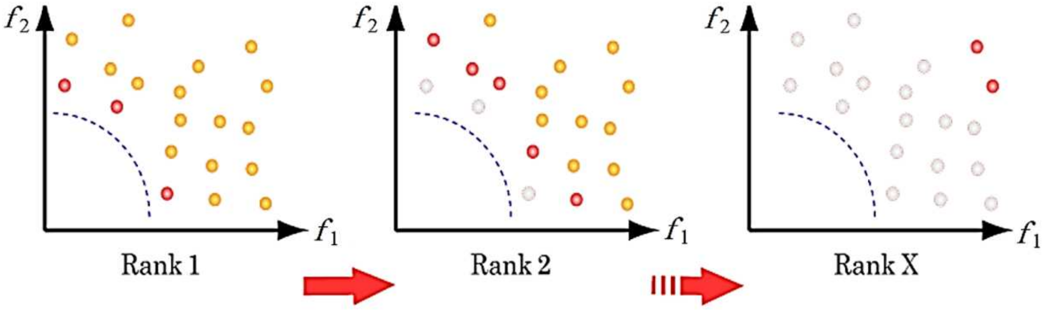


Fig. 4. Assigning different ranks to existing solutions in NSGA-II (Deb et al., 2002).

4.1.2 Calculating Solution Density Index

To determine solution density around a specific point which is a measure for adjusting the variety of population, the average nearest solution at two sides of the point is considered for all fitness functions. Here, quantity $i_{distance}$ stands for a size estimation of the largest cuboid surrounding the point i without including the other points in the population which is called crowding distance. Figure 5 illustrates this concept for two fitness functions (Deb et al. (2002)). Accordingly, the crowding distance of the solution i in its front (marked with solid circles) represents the average side length of the cuboid (shown with a dashed box). Although Figure 5 depicts the crowding-distance calculation for two objectives, the approach is applicable to more than two objectives as well.

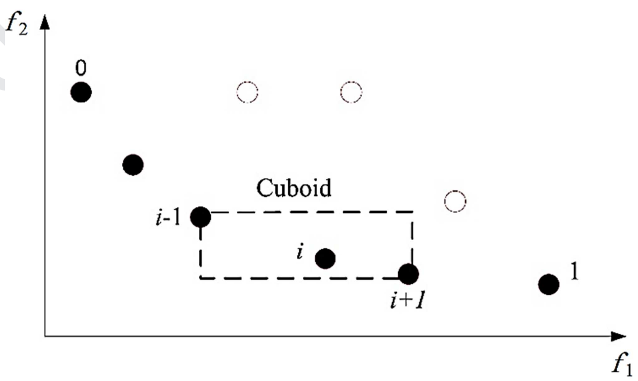


Fig. 5. Calculating Crowding Distance (Deb et al., 2002).

4.1.3 Selecting Solution for Generating Next Generation

In order to implement NSGA-II algorithm, a first initial parent population P is generated. The population is sorted based on the sorting algorithm and the Pareto front is assigned to each solution. Now, the optimization problem is converted into a simple problem of minimizing the

Pareto front fitness function. Selection operators of the binary tournament, mutation, and crossover are employed to create an offspring Q of size N. The method of this generation is different due to the elitism mechanism. In the elitism mechanism, first, a hybrid population of parents and children is formed. Then, the hybrid population is sorted based on the crowding comparison operator, and N best solutions are considered as future population P_t +1. Then, with population size N of P_t+1 and by applying mutation, crossover, and selection operators, the population with size N of Q_t+1 is built. In this algorithm, the population variety of each generation is guaranteed by applying the crowding comparison operator while selecting the binary tournament in which the sharing parameter is not required. Thus, this method does not have the shortcoming of other methods like NSGA. Moreover, the crowding distance is calculated in fitness functions space, which can also be calculated using parameters' space. Another point in building the population of each generation is that a+b selection method is used instead of (a, b). This increases the stability of the method and guarantees that the elite population of the previous generation is not eliminated in the new generation. Implementing NSGA-II according to (Deb et al. (2002)) is depicted in Figure 6. Accordingly, F_1 denotes the best non-dominated set includes the best solutions in the combined population. A combined population is defined by $R_t = Q_t \cup P_t$, which has the size of 2N. Then, solutions from the set F_2 have the second priority to be chosen, followed by solutions from the set F_3, and so on.

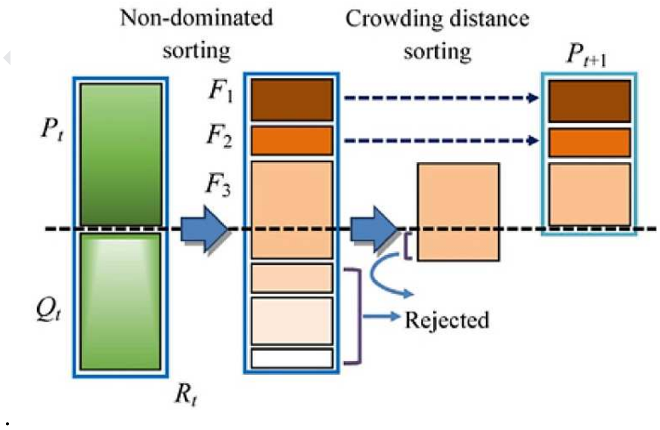


Fig. 6. Implementation of NSGA-II.

Finally, to summarize the structure of the proposed NSGA-II algorithm, the flowchart of this algorithm is provided and presented in Figure 7.

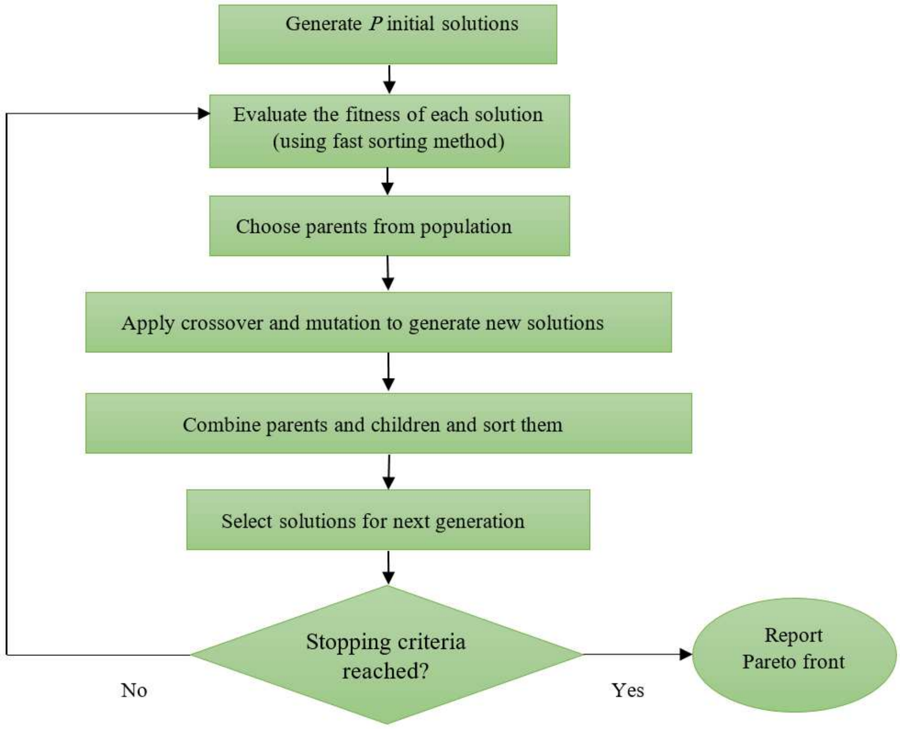


Fig. 7. Flowchart of NSGA-II algorithm.

4.2 Multi-objective Simulated-annealing Algorithm

SA is a local search meta-heuristic algorithm which can prevent from being trapped in local optima. This algorithm is very efficient in solving nonconvex or discrete problems. Thus, SA is used to solve integer programming problems (Glover and Kochenberger, 2006). Ever since its introduction, SA has demonstrated a high performance in large combinatorial optimization problems (Kubotani and Yoshimura, 2003). Moreover, implementation simplicity, convergence, and hill-climbing for avoiding local optima make use of this algorithm to improve the best result. MOSA tries to generate non-dominated solutions by applying a simple probability function that attempts to generate solutions on the Pareto optimal front. This probability function is different so that the total space of objective is covered uniformly obtaining as many possible non-dominated and well-dispersed solutions (Varadharajan and Rajendran, 2005). These aforementioned characteristics have made MOSA a fast reliable algorithm in comparison with the other existing multi-objective algorithms with a wide application in optimization problems.

The overall framework of this algorithm is that it begins with one solution, initial parameters include the number of iterations at each temperature (M), initial temperature (T_0), temperature reduction rate (α), final temperature (T_{end}) and Boltzmann constant (K), which are initialized after starting the search operations. Then, a neighborhood is taken into account for the initial solution, and if the objective function value of the neighbor is superior to the current solution objective value, the neighboring solution is replaced with the current solution. Otherwise, the difference value of the neighboring solution and the initial solution is calculated and a randomly-generated number in [0, 1] is regarded and compared with the probability of $\exp\left(-\frac{\Delta}{KT}\right)$, if the random number is less than $\exp\left(-\frac{\Delta}{KT}\right)$, the worse solution is accepted. A given number of iterations are executed at each temperature, and then the temperature is reduced. Annealing or temperature reduction equation is stated as $\alpha T \rightarrow T$. The termination condition is defined by reaching the final temperature. The mechanism of the proposed MOSA is adapted from that introduced by Kubotani and Yoshimura (2003). MOSA algorithm is established on the basis of SA algorithm with this difference that the non-dominance concept as in NSGA-II is used. The advantage of MOSA over other EAs is that high memory is not required to keep population information (Nam and Park, 2000). The flowchart of the MOSA algorithm is presented in Figure 8.

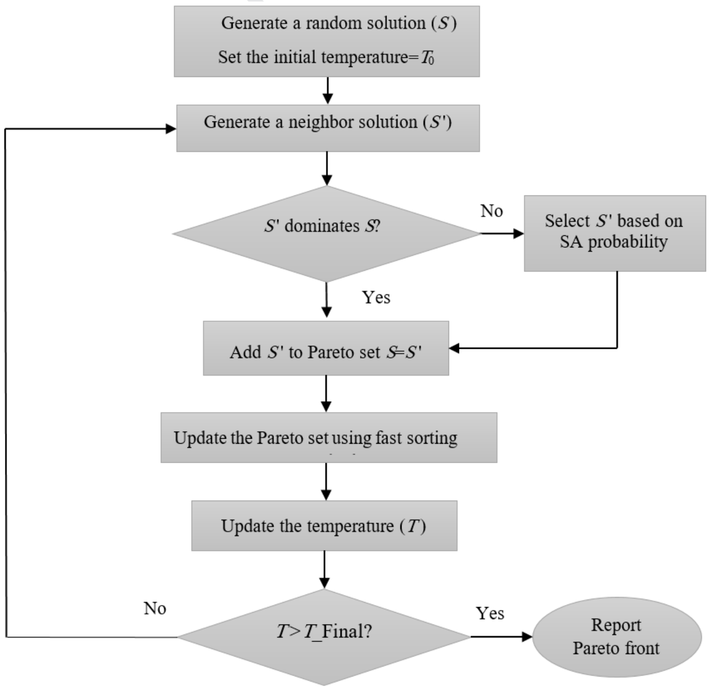


Fig. 8 Flowchart of MOSA algorithm.

4.3 Solution Representation

One of the most important components of meta-heuristic algorithms is to find an efficient way of representing the solution or provide the coding system of mathematical modeling. In this study, a solution representation for the PRP-CDS problem is designed which includes two main parts. The first part is a vector with a length of P+3*K_1, where it contains three elements. The first P cells represent the priority order of suppliers. The first K_1 cells show the number of suppliers that each vehicle visits. The second K_1 cells identify the origin of each vehicle in the pick-up phase. Finally, the last K_1 cells display the destination of each vehicle. Figure 9 provides an example for the first part of the solution representation, where P=5, K_1= 3 and C=2.

4	2	3	1	5	3	2	0	6	7	7	7	7	7
Supplier permutation					Number of visited points			Origin cross-dock			Destination cross-dock		

Fig. 9. An example of the first part of solution representation.

As shown in Figure 9, the first vehicle moves from the cross-dock 6 and then visits suppliers 4, 2 and 3, respectively, and eventually moves to cross-dock 7. The second vehicle moves from cross-dock 7 to suppliers 1 and 5, respectively, and finally moves back to cross-dock 7. The vehicle car is also not used.

The second part of the solution representation displays the routes of vehicles in the delivery phase. It contains a vector of length 2*D+3*K_2. This vector is approximately the same as that proposed in the first part of the solution, however, the difference is that it has one more element including the speed level of vehicles. Figure 10 gives an example for the second part of the solution representation, where D=5, C=2 and K_2=3 and three speed levels are considered.

9	13	10	14	8	3	2	1	2	3	1	2	2	7	6	6	6	7	6
Customer permutation					Speed level					Number of visited points			Origin cross-dock			Destination cross-dock		

Fig. 10. An example of the second part of solution representation.

In Figure 10, the first vehicle starts from cross-dock 7, then moves to customer 9 and finally moves to cross-dock 6. This vehicle has three pre-defined speed levels in its routes. Similarly, other routes are constructed. In each route, the amount of products should be picked up or delivered in such a way that the capacity constraint of vehicles is not violated. Moreover, if the total picked-up products from suppliers are less than the amounts required for customers, the penalty M (the optional large number) is added to the system.

5. Parameters Tuning and Comparison Measures

In this section, the Taguchi design method is used to determine optimal values for the proposed algorithms; to compare the performances, four different measures, namely, MID, SM, DM, and CPU-T are considered which are described in the following. It should be mentioned that these two algorithms are simulated in MATLAB R2015 on a system with CPU Core i7 and 8G RAM.

5.1 Taguchi Design Method

Performances of meta-heuristic algorithms depend on input parameters. In fact, if the parameters of an efficient algorithm are not adjusted correctly, then this algorithm becomes inefficient. The increase of research costs allows for using experimental methods which give exact results with a minimum number of experiments necessary. In the previous two decades, different techniques have been proposed to design experiments, where one of the first methods in this context is the factorial method in which the number of experiments is obtained by $N=L^m$. The main disadvantage of this method is that if the number of variables is high, the number of experiments becomes high also which is not cost-effective in terms of time and cost. However, the Taguchi design method compared to other methods is used widely due to being more comprehensive. In this method, the effect of uncontrollable factors on the solution is reduced using a parameter called Signal to Noise ratio (S/N). It shows the dispersion around a given value. In fact, it implies how the obtained solutions have altered among several experiments. In this paper, Eq. (26) is employed to measure S/N (*the smaller the better*) which is used for minimizing objective functions; here, y_i is the solution point in i^{th} experiment (i=1,2,...,n) and n is the number of experiments (Taguchi, 1986):

$$(S/N)_s = -10 \log_{10}\left(\frac{\sum_{i=1}^{n} y_i^2}{n}\right). \tag{26}$$

Finally, utilizing Minitab ® software by the Taguchi method, L_9 design for NSGA-II and L_{27} design for MOSA are proposed.

5.1.1 Adjusting Parameters of NSGA-II

As mentioned previously, GA has a large number of factors and parameters which affect the final solution and the performance of the algorithm. Thus, achieving a suitable combination of these factors might improve the efficiency of the algorithm to a great extent. In order to adjust parameters of NSGA-II, three factors at low, moderate and high levels are proposed. Table 6 describes the parameters and search domain of input parameter levels of NSGA-II for the suggested model.

Table 6. Parameters of the proposed NSGA-II.

Algorithm	Parameter	Range	Low (1)	Medium (2)	High (3)
NSGA-II	**Initial population (A)**	200-300	200	250	300
	Percentage of Crossover (B)	0.7-0.9	0.7	0.8	0.9
	Percentage of Mutation (C)	0.1-0.3	0.1	0.2	0.3

Now, one of the MCDM methods called Simple Additive Weighting (SAW) is applied to convert measurements to one solution for selecting optimal levels of the parameters. SAW technique can be represented in multiple steps (Zitzler et al., 2000):

***Step* 1:** Nature of each index should be determined based on its being positive or negative.

***Step* 2:** Values obtained for measures in the decision matrix are descaled in Eqs. (27) - (28):

For negative criteria: $$n_{ij} = \frac{r_j^{min}}{r_{ij}}, \qquad i=1,\ldots,m; \; j=1,\ldots,n, \tag{27}$$

For positive criteria: $$n_{ij} = \frac{r_{ij}}{r_j^{max}}, \qquad i=1,\ldots,m; \; j=1,\ldots,n. \tag{28}$$

Here, r_{ij} is the positive normalized weight of experiment i for measure j.

Step* 3*: Considering importance coefficients or weight of measures and normalized values of the decision matrix, SAW can be calculated for each experiment according to Eqs. (29) - (30):

$$\sum_{j=1}^{n} w_j = 1, \tag{29}$$

$$SAW_i = \sum_{j=1}^{n} w_j n_{ij}, \qquad i=1,\ldots,m. \tag{30}$$

The importance of all measures is considered to be uniform and positive. Since normalized weights of experiments for measures are positive too, higher values of SAW are used in the decision-making process. Therefore, considering the above discussion, at every 3^3=27 levels, a sample problem should be solved. However, after executing the Taguchi design method, L_9 array is selected as a suitable experimental scheme for adjusting parameters. The *S*/*N* plot is depicted in Figure 11. Accordingly, the best combination of the parameters is obtained considering maximum values of the mean of S/N ratios. For example, the second levels of the parameters A and B are the best ones in NSGA-II. Therefore, the best values for the number of initial population and percentage of crossover operator are 250 and 0.8, respectively.

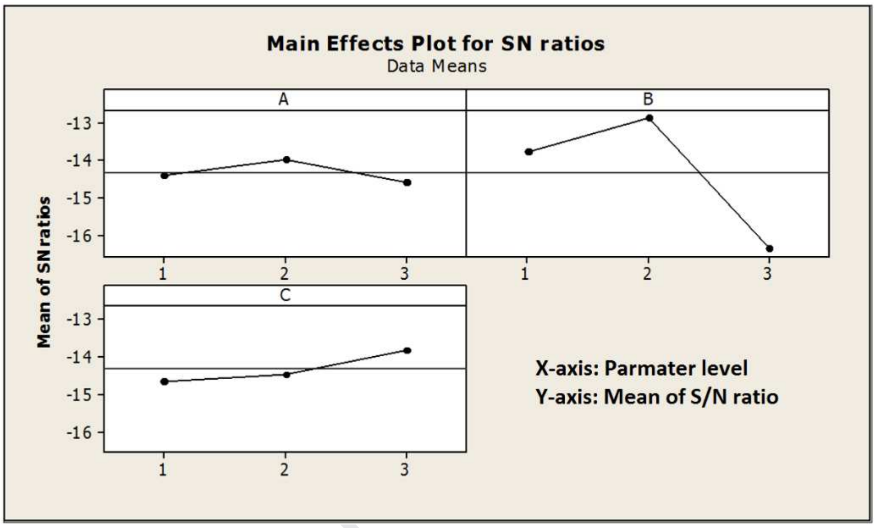


Fig. 11. Average *S*/*N* ratio plot at different levels for NSGA-II.

According to Figure 11, a suitable value for each parameter is the one which has the minimum *S*/*N* value. Thus, the values of Table 7 are optimal for the NSGA-II algorithm while the number of required generation for improving algorithm is considered to be 50. So, the mechanism of the proposed algorithm is constructed.

Table 7. Optimal levels of the parameters for NSGA-II.

Algorithm	Parameter	Optimum
NSGA-II	**Initial population (A)**	250
	Percentage of crossover (B)	0.8
	Percentage of mutation (C)	0.2
	Maximum iteration (Max-iteration)	50

5.1.2 Adjusting Parameters of MOSA Algorithm

To adjust the parameters of the MOSA algorithm, 5 factors are determined at 3 levels. Table 8 describes the search domain at input parameter levels of the MOSA algorithm for the suggested model.

Table 8. Parameters of the proposed MOSA.

Algorithm	Parameter	Range	Low (1)	Medium (2)	High (3)
MOSA	Maximum number of iterations (A)	1-5	1	2	5
	Initial temperature (B)	150-300	150	200	300
	Temperature reduction rate (C)	0.8-0.9	0.8	0.85	0.9
	Boltzman constant (D)	0.2-0.7	0.2	0.5	0.7
	Final temperature (E)	1-5	1	2	5

Considering the Taguchi standard orthogonal arrays, L_{27} array is selected as a suitable experimental scheme for adjusting parameters of the proposed MOSA algorithm. Here, L_{27} array is the experimental design with 27 experiments. If experiments are performed completely, 3^5=243 experiments are required to adjust the parameters of the MOSA algorithm. After executing the Taguchi design method, the S/N plot is given in Figure 12. Similarly, the best combination of the parameters is obtained considering maximum values of the mean of S/N ratios. For example, the third levels of the parameters A and B are the best ones in MOSA. Therefore, the best values for the maximum number of iterations and initial temperature are 5 and 300, respectively.

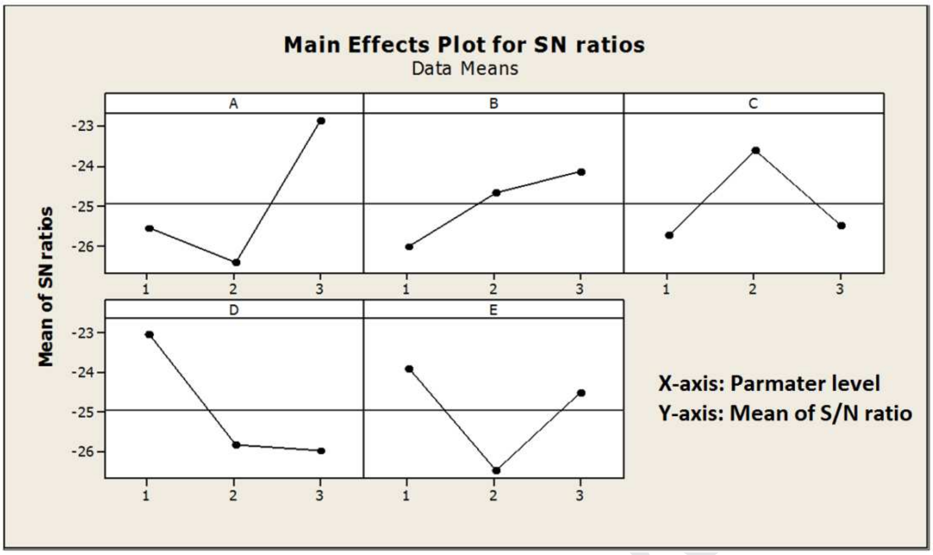


Fig. 12. Average *S*/*N* ratio plot at different levels in MOSA.

Now, based on the output presented in Figure 12, the optimal value of each parameter is specified in Table 9, and other examples are executed using the parameters of this algorithm.

Table 9. Optimal levels of the parameters for MOSA.

Algorithm	Parameter	Optimum
MOSA	**Maximum number of Iterations (A)**	5
	Initial Temperature (B)	300
	Temperature reduction rate (C)	0.85
	Boltzmann constant (D)	0.2
	Final temperature (E)	1

5.2. Performance Comparison of Multi-Objective Algorithms

To evaluate the performance of the proposed methods, it is required to present some metrics to evaluate their outputs. For this purpose, 4 metrics, called MID, SM, DM and CPU-T, are presented according to Zitzler et al. (2000) which are explained in the following section.

5.2.1 MID: This measure is used to calculate the mean distance of Pareto solutions from the origin. In Eq. (31), c_i is the distance of Pareto solution from the so-called *ideal point*. Considering this relation, it is clear that the less is this metric, the efficiency of this algorithm would be higher:

$$MID = \frac{\sum_{i=1}^{n} c_i}{n}. \quad (31)$$

In multi-objective contexts based on the Pareto approach, one of the objectives is that boundaries should be closer to coordinate origin. Thus, this measure calculated the distance of fronts from the best population value. Figure 13 shows the MID metric schematically (Deb et al., 2002). Here, *n* is the number of Pareto solutions (marked with black circles) and their mean distance from the *ideal point* (marked with red circle) is obtained as the mentioned metric.

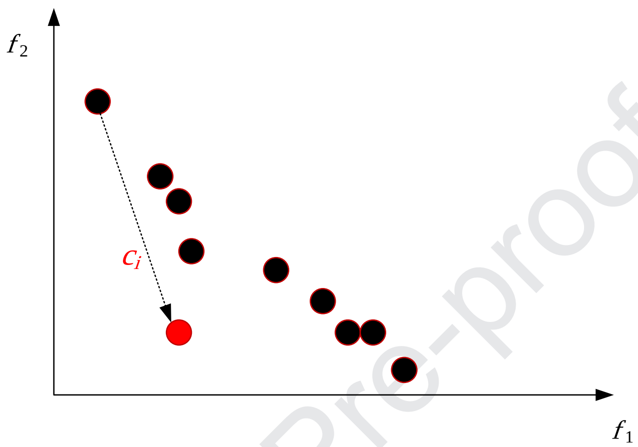


Fig. 13. MID criteria for evaluating multi-objective algorithms.

In this paper, the *ideal point* is considered to be the best value of each objective function in all algorithms. Furthermore, Eq. (32) is used to calculate MID:

$$MID = \sum_{i=1}^{n} \frac{\sqrt{\left(\frac{f_{1,i}-f_1^{best}}{f_{1.total}^{max}-f_{1.total}^{min}}\right)^2+\left(\frac{f_{2,i}-f_2^{best}}{f_{2.total}^{max}-f_{2.total}^{min}}\right)^2}}{n}, \tag{32}$$

where n is the number of Pareto points, $f_{1,i}$ and $f_{2,i}$ are the values of i^{th} Pareto solutions, and $f_{j.total}^{max}$ and $f_{j.total}^{min}$ are the maximum and minimum values of each objective function j among all non-dominated solutions obtained by an algorithm, respectively. Moreover, (f_1^{best}, f_2^{best}) consists of the coordinates which comprise the *ideal point*.

5.2.2 SM: This measure shows a uniform distribution of Pareto solutions in solution space. This measure is calculated as in Eq. (33), where d_i is the Euclidean distance between two marginal Pareto solutions in the solution space and $\bar{d}$ is average distances. The smaller the SM value is, the better the algorithm performs:

$$d_i = \sqrt{\left(f_{2,i+1} - f_{2,i}\right)^2 + \left(f_{1,i+1} - f_{1,i}\right)^2}, \qquad SM = \frac{\sum_{i=1}^{n-1}\left|\bar{d} - d_i\right|}{(n-1)\bar{d}}. \tag{33}$$

5.2.3 DM: This measure shows the variety of Pareto solutions which can be calculated using Eq. (34). The higher the DM value is, the better the algorithm performs:

$$DM = \sqrt{\sum_{i=1}^{n}\left(\frac{\max f_{1,i} - \min f_{1,i}}{f_{1.total}^{max} - f_{1.total}^{min}}\right)^2 + \left(\frac{\max f_{2,i} - \min f_{2,i}}{f_{2.total}^{max} - f_{2.total}^{min}}\right)^2}. \quad (34)$$

5.2.4 CPU-T: In large problems, one of the most important measures is the solution time. Hence, the algorithm execution time is considered as one of the most important quality evaluation indices.

6. Computational Results

This section provides the computational results of the study which are obtained by the proposed solution techniques. To this end, several problems are generated randomly in different sizes. Figure 14 represents an overall framework to demonstrate the relevance and the link between the numerical results and claimed contributions.

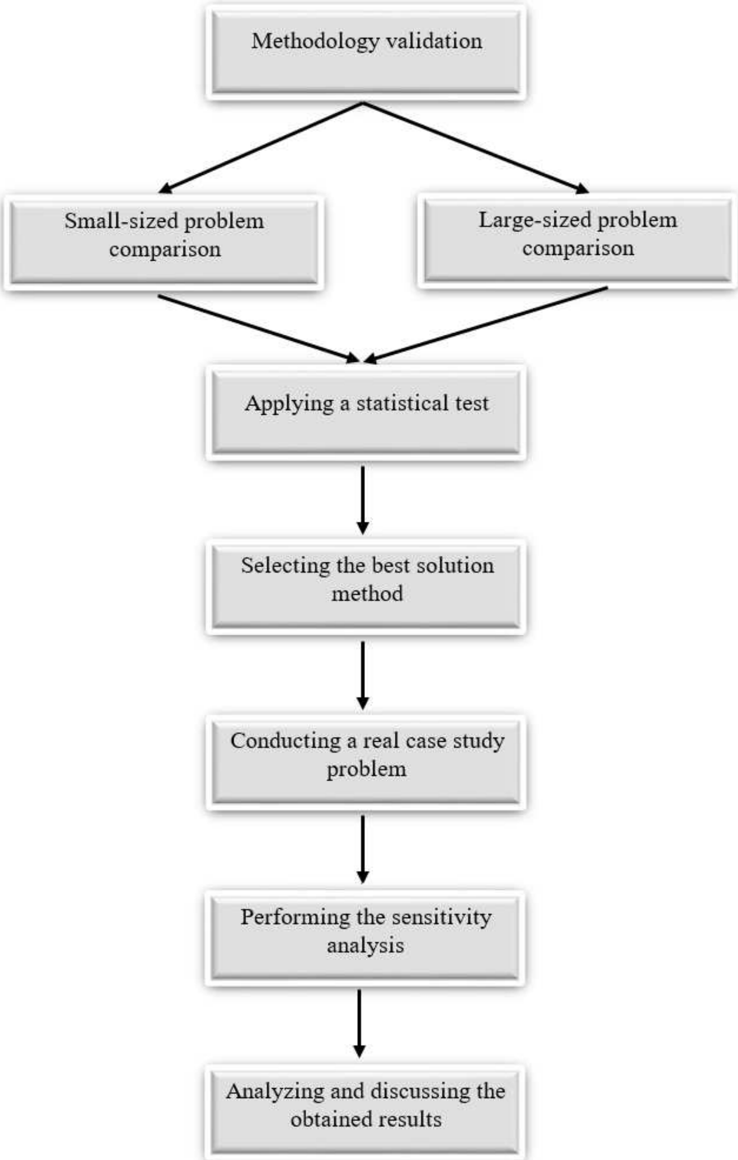


Fig. 14. Structure of the results analysis.

According to Figure 14, the methodology validation is provided by solving several different problems.

In the following, Table 5 presents the input data of the randomly-generated problems. Problems with medium and large size solved by NSGA-II and MOSA are compared with the ε-constraint method which is implemented in GAMS, and the efficiency of both of them is analyzed. Since the exact method cannot find a solution for large-sized problems in a reasonable computational time, the two proposed algorithms are applied to solve the problems. This is a good way to analyze the performance of the proposed algorithms in comparison with each other. It is notable that all the algorithms described in this paper are implemented in Matlab software. Details on the parameter values are shown in Tables 10 and 11. It should be noted that most of the parameters' values are adapted from Franceschetti et al. (2013). According to Table 10, problems P_1-P_4 are categorized into small and medium sizes due to observing a run time limit of 3600 seconds for the exact method.

Table 10. Input data for randomly-generated problems.

#Problem	Notations						
	C	P	D	T	R	K_1	K_2
P_1	2	3	5	1	1	1	1
P_2	5	6	12	4	3	2	2
P_3	10	8	20	6	3	2	3
P_4	15	12	30	12	4	3	4
P_5	20	14	35	12	4	3	5
P_6	25	16	40	18	5	4	5
P_7	30	25	50	24	5	5	6
P_8	35	30	60	24	6	5	7
P_9	40	40	80	32	8	6	7
P_{10}	45	45	90	32	8	7	8
P_{11}	50	50	100	48	10	7	9
P_{12}	70	60	120	48	15	8	10

Table 11. Considered values for the parameters.

Parameters	Value
Q_1	200
Q_2	150
v^{mr}	Uniform(5,100)
σ_{it}	1.5
q_{it}	Uniform(5,10)
$scap_i$	Uniform(1000,5000)
d_{ij}	Uniform(5,12)
l_{it}	Uniform(1,10)
u_{it}	Uniform(2,100)
h_{it}	Uniform(0.5,1)
dc_t	Uniform(100,120)
c_{ij}	Uniform(3,7)
ξ	1

Parameters	Value
κ	44
ψ	737
Γ	0.2
N_e	33
V	5
ρ	1.2041
c_d	0.7
A	3.912
μ	6350
ε	0.4
f_{ct}	1.4
$\overline{\varpi}$	0.9

As can be seen in Table 10, the columns indicate the problem number, number of cross-docks, number of suppliers, number of customers, number of time periods and the number of speed levels defined for each vehicle, respectively. Furthermore, Table 11 shows the values considered for the parameters of the model.

6.1 Efficiency Evaluation of the Solution Methods

The output results of different solution methods are reported in this section. The objective values for the algorithms and ε-constraint method are computed in the form of Pareto solutions by presenting the obtained values for the 1st and 2nd objectives. The steps of generating Pareto solutions by the ε-constraint method are represented by the obtained values in Tables 12 and 13.

Table 12. Obtained results of single-objective problem-solving.

Single-objective Problem	TC	TR
minimize TC	1335.02	338.98
maximize TR	1610.38	392.26

Table 13. Different epsilon's values obtained for 2nd objective function.

Breakpoints	ε_2
1	338.98

Breakpoints	ε_2
2	344.308
3	349.636
4	354.964
5	360.292
6	365.62
7	370.948
8	376.276
9	381.604
10	386.932

After determining all the 10 epsilons in Table 13, the Pareto solutions are reported in Table 14. To evaluate these Pareto solutions obtained by the proposed solution methods, the solution of the first instance problem is illustrated in Table 14.

Table 14. Obtained Pareto solutions for the proposed solution methods.

# Pareto Solution	ε-constraint		NSGA-II		MOSA	
	TC ($)	TR (load unit)	TC ($)	TR (load unit)	TC ($)	TR (load unit)
1	1379.04	354.06	1442.16	357.23	1483.54	360.96
2	1346.49	347.68	1490.39	366.78	1496.26	361.26
3	1406.56	359.31	1529.16	376.61	1603.08	385.84
4	1355.92	349.89	1397.52	349.21	1532.16	369.3
5	1467.62	375.95	1464.24	361.92	1643.2	392.16
6	1542.47	387.74	1507.05	369.09	1598.96	384.08
7	1456.58	372.18	1581.89	393.54	1562.76	380.2
8	1369.28	351.59	1355.77	336.55	1512.59	365.19
9	1442.75	366.9	1694.33	397.98	1474.34	359.57
10	1486.01	388.73	1482.6	365.78	1655.42	395.66

Table 14 shows the different objective values computed by NSGA-II, MOSA and ε-constraint method. It is obvious that by decreasing the 1st objective, the 2nd objective decreases, which indicates that there is a conflict between two objectives. Note that if you want to have a more reliable system, you should concurrently incur more costs and provide the required resources. This conflict is depicted in Figure 15.

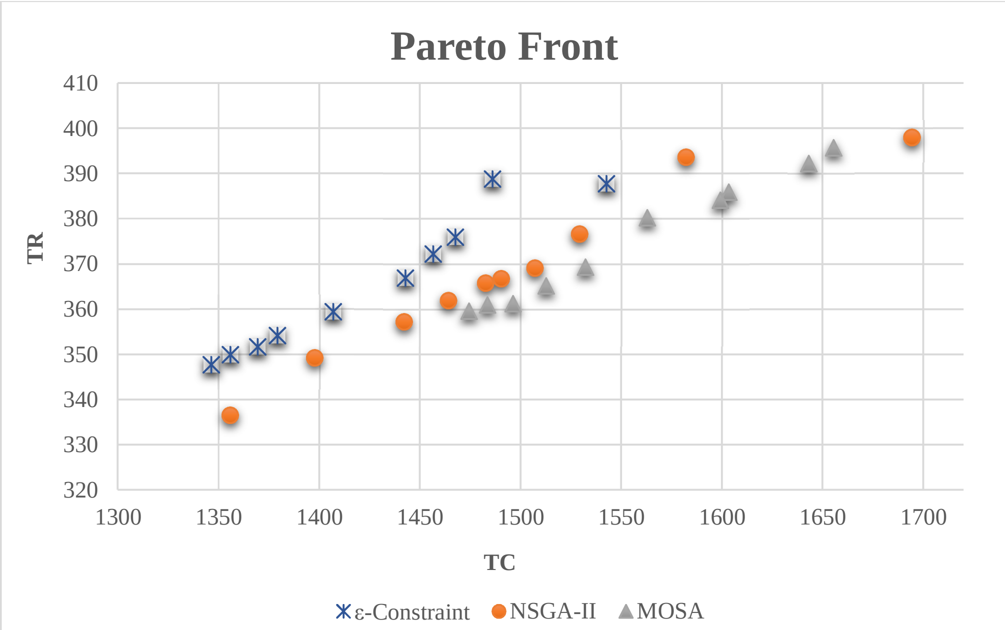


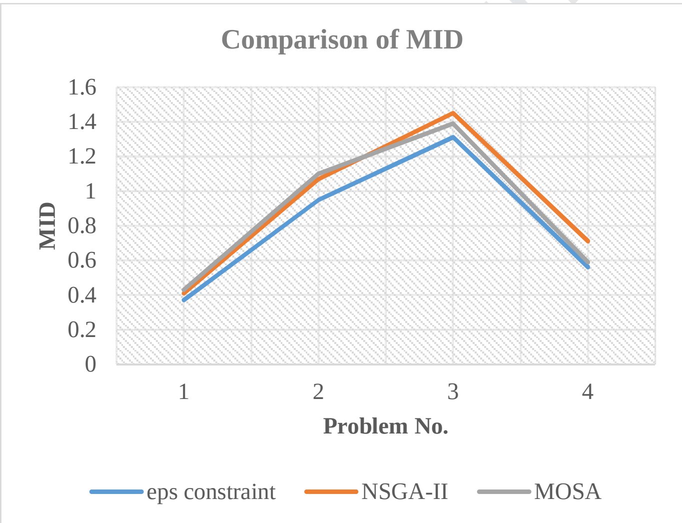
Fig. 15. Pareto fronts generated by different solution methods for P_1.

According to Figure 15, it can be concluded that Pareto fronts generated by the proposed solution methods are approximately similar to a small gap. As can be seen, all the Pareto points of the ε-constraint method are beyond the bisector line, meaning a higher 2nd objective and a lower 1st objective, which is a superiority. Hence, it is obvious that the proposed NSGA-II has an appropriate performance in comparison with the ε-constraint method due to its proximity to the bisector line and the Pareto points of the ε-constraint method. Moreover, MOSA has generated good solutions in comparison with NSGA-II, and it can be concluded that its efficiency is satisfactory. In fact, it provides some Pareto points above the bisector line and better than NSGA-II. Therefore, a high efficiency is proved for the proposed algorithms by analyzing Figure 12. In order to have a more exact evaluation and verification of the proposed algorithms, and to find out how much they are capable to detect optimal Pareto fronts, measures introduced in the previous section are applied. The efficiency of algorithms can be easily investigated by calculating three measures including MID, SM, and DM and considering the SAW values derived from these measures. These values are shown in Table 15 for four problems.

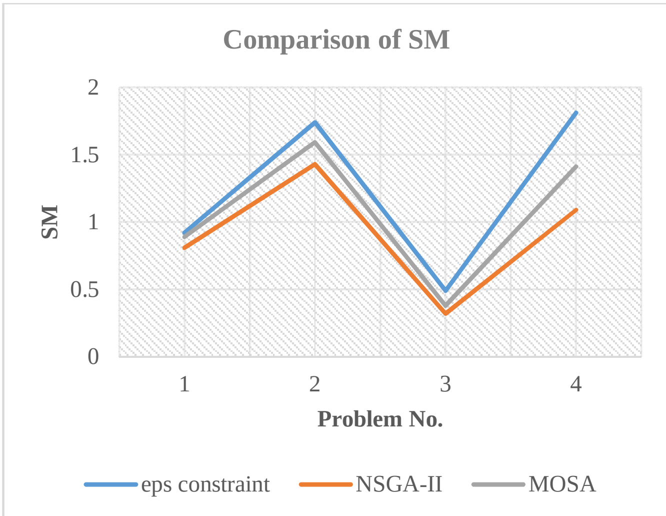
Table 15. Efficiency verification of the proposed solution methods in problems P_1- P_4.

#Problem	Solution method	Measures			
		MID	SM	DM	SAW
P_1	ε-constraint	0.49	0.37	0.92	1.89
	NSGA-II	0.54	0.41	0.81	1.7
	MOSA	0.72	0.43	0.89	1.53
P_2	ε-constraint	0.79	0.95	1.74	1.35
	NSGA-II	0.8	1.07	1.43	1.2
	MOSA	0.85	1.1	1.59	1.23
P_3	ε-constraint	0.58	1.31	0.49	0.99
	NSGA-II	0.61	1.45	0.32	0.88
	MOSA	0.73	1.39	0.38	0.82
P_4	ε-constraint	0.82	0.56	1.81	1.61
	NSGA-II	0.89	0.71	1.09	1.21
	MOSA	0.92	0.59	1.41	1.4

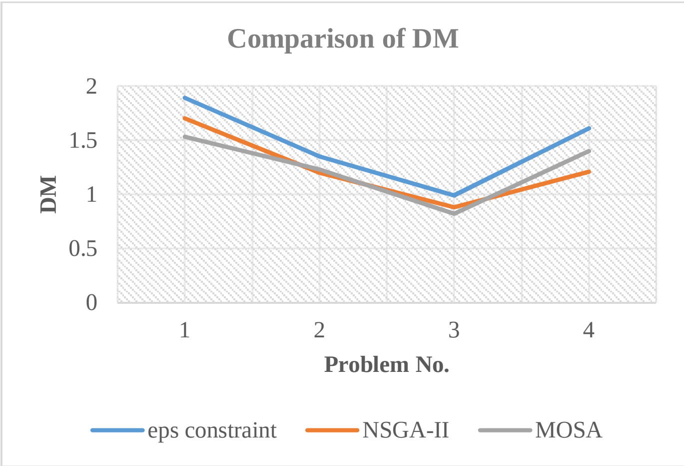
Furthermore, the solution methods can be schematically analyzed using Figure 16. As it is clear, the proposed algorithms can generate high-quality solutions very close to the suggested exact method. So, they can be employed as proper tools for solving the large-sized problems.



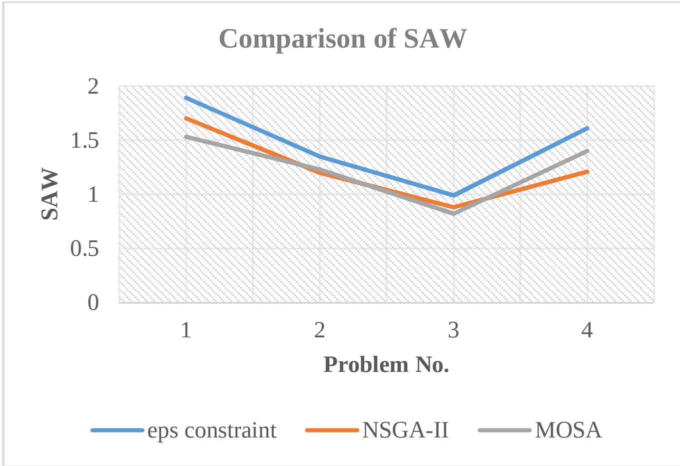
(a) MID values for three different solution methods in problems P_1 - P_4.



(b) SM values for three different solution methods in problems P_1 - P_4.



(c) DM values for three different solution methods in problems P_1 - P_4.



(d) Final comparison of three different solution methods in problems P_1 - P_4.

Fig. 16. Comparison of the different solution methods-problems P_1 - P_4.

As it can be seen in Figure 16, according to the optimal MID values, NSGA-II had certainly a better performance than MOSA. This means that NSGA-II generates more qualified solutions and its Pareto fronts are close to the ideal Pareto front. According to SM values, none of the algorithms have better performance than the others. In small-sized problems (problems P_1- P_2), MOSA seems to be a little better. However, NSGA-II has a superior performance for medium-sized problems (problems P_3- P_4), and with this increase in the problem size, NSGA-II seems to have a better performance than MOSA. In terms of DM values, MOSA has succeeded in finding a wider range of solutions to the NSGA-II, and this has occurred in all four problems. Finally, according to the SAW values, NSGA-II and MOSA performed very closely to each other for the problems with small and medium sizes and solved them optimally. However, NSGA-II is slightly better than MOSA. But what is important is that the output of both of these algorithms is valid and the obtained solutions are reliable. Therefore, they can be applied to solve large-sized problems (problems P_5 - P_{12}) to study the output and efficiency of the algorithms in comparison with each other more confidently. Table 16 shows the results.

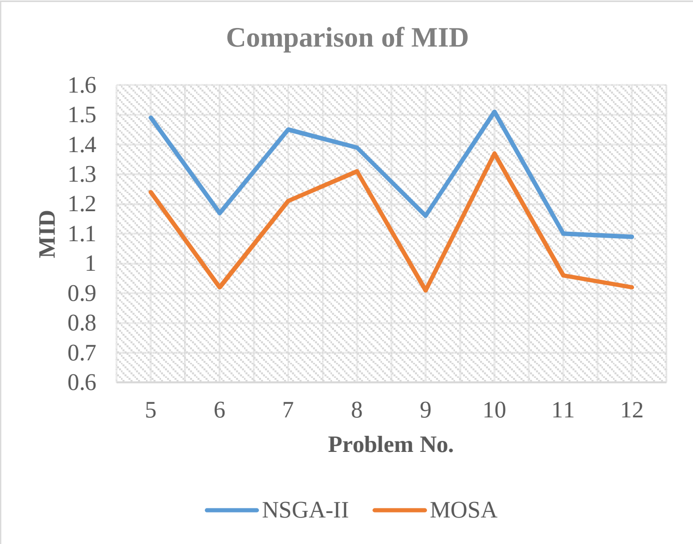
Table 16. The obtained results for large-sized problems.

#Problem	DM			MID			SM		
	ε-constraint	NSGA-II	MOSA	ε-constraint	NSGA-II	MOSA	ε-constraint	NSGA-II	MOSA
P_5	2.12	2.03	1.74	1.41	1.49	1.24	0.39	0.34	0.59
P_6	0.71	0.73	0.51	1.49	1.17	0.92	0.99	0.83	0.96
P_7	1.14	1.24	0.76	1.64	1.45	1.21	0.74	0.69	0.91
P_8	-	1.39	0.92	-	-	-	2.15	-	1.88
P_5	-	0.88	0.81	-	-	-	0.68	-	0.69
P_9	-	2.19	1.69	-	-	-	2.17	-	2.12
P_{10}	-	0.44	0.32	-	-	-	1.36	-	1.09
P_{11}	-	1.82	1.31	-	-	-	2.34	-	2.41

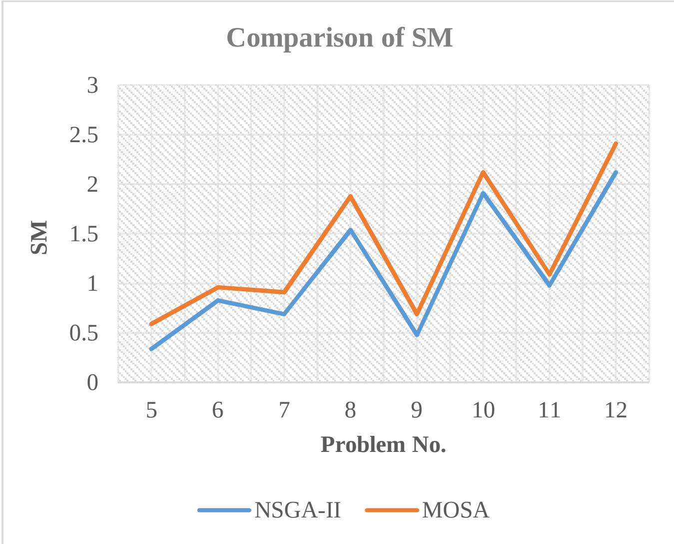
Table 17. SAW values for large-sized problems.

Solution methods	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}
ε-constraint	1.78	0.81	1.05	-	-	-	-	-
NSGA-II	1.41	0.88	0.9	0.74	1.12	0.96	0.76	0.94
MOSA	1.88	0.93	1.13	0.92	1.28	1.13	0.79	1.07

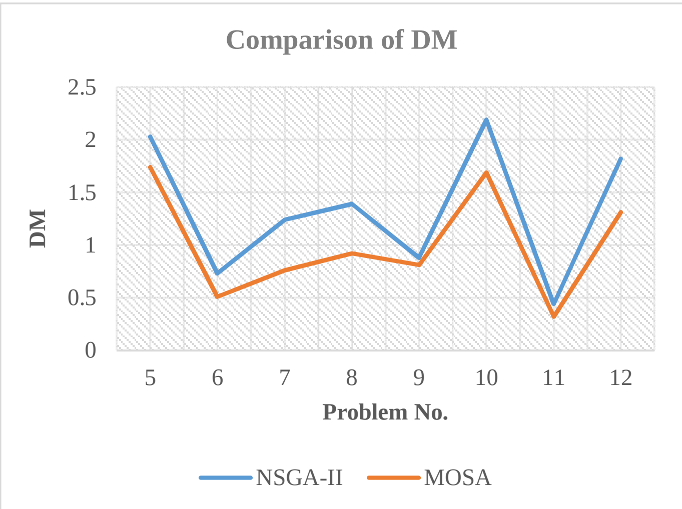
As can be seen from Table 17, the ε-constraint method has lost its efficiency in large-sized problems, and the meta-heuristic algorithms have shown a significant performance. Furthermore, GAMS software was unable to report a solution from P_8 onwards by applying the time limit of 3600 seconds. By comparing the proposed measures, it is found out that which algorithms have more appropriate performance. This comparison is shown more clearly in Figure 17.



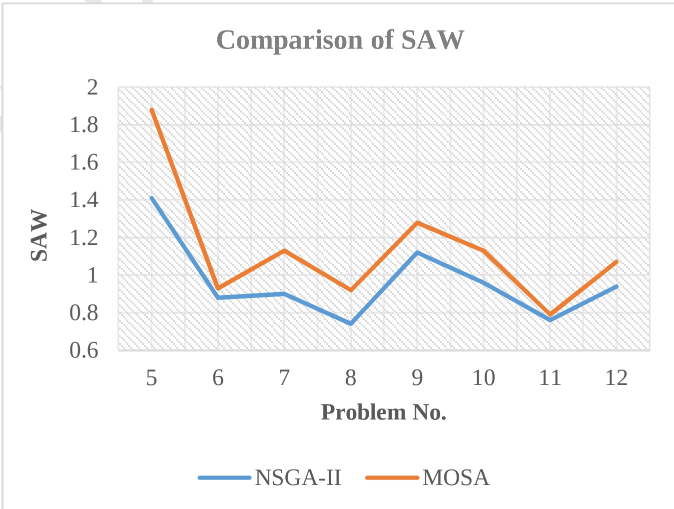
(a) MID values for two algorithms in problems P_5 - P_{12}.



(b) SM values for two algorithms in problems P_5 - P_{12}.



(c)DM values for two algorithms in problems P_5 - P_{12}.



(d) Final comparison of two algorithms in problems P_5 - P_{12}.

Fig. 17. Comparison of the two proposed algorithms-problems P_5 - P_{12}.

According to Figure 17, it is clear that with increasing the problem size, the performance gap between the two algorithms becomes more significant. NSGA-II has a better performance with respect to the MID criterion. MOSA has a superior performance with respect to the SM criterion in all problems. This difference between the algorithms for SM is far less than the MID. In terms of DM values, MOSA has succeeded in finding a wider range of solutions to the NSGA-II, and this has occurred in all problems. Finally, it is disclosed that NSGA-II has a better overall performance. Except for the two problems P_6- P_{11}, in the remaining problems, NSGA-II has a remarkable advantage over MOSA algorithm. However, in the two problems mentioned, the relative superiority continues with the NSGA-II algorithm. Therefore, NSGA-II can be introduced as a more efficient algorithm in this research. Furthermore, the Pareto front generated by the algorithms for P_{12} is illustrated in Figure 18 to understand the superiority of NSGA to MOSA visually.

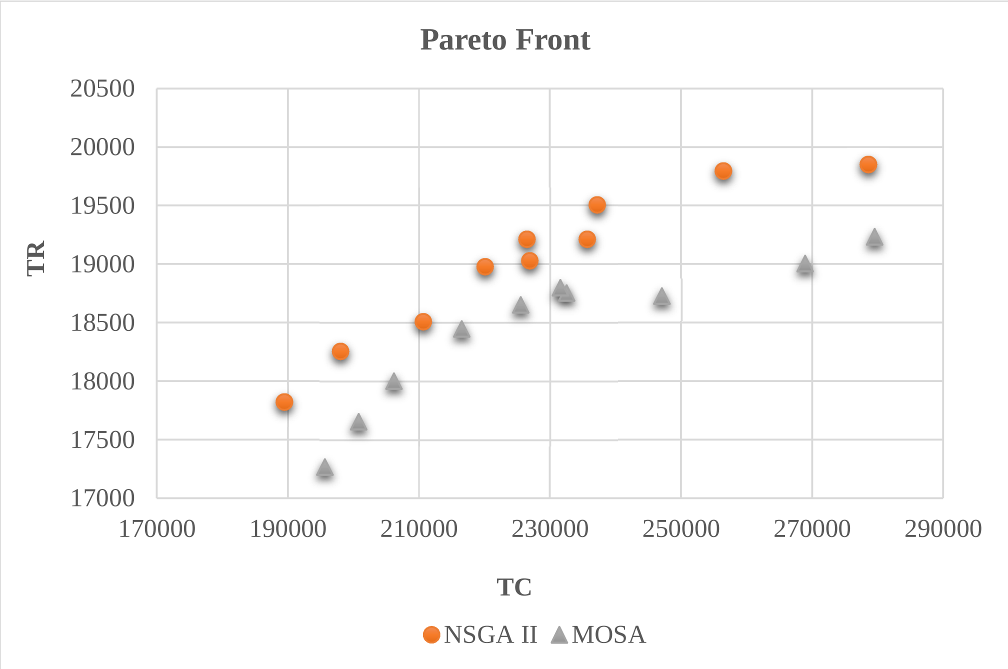


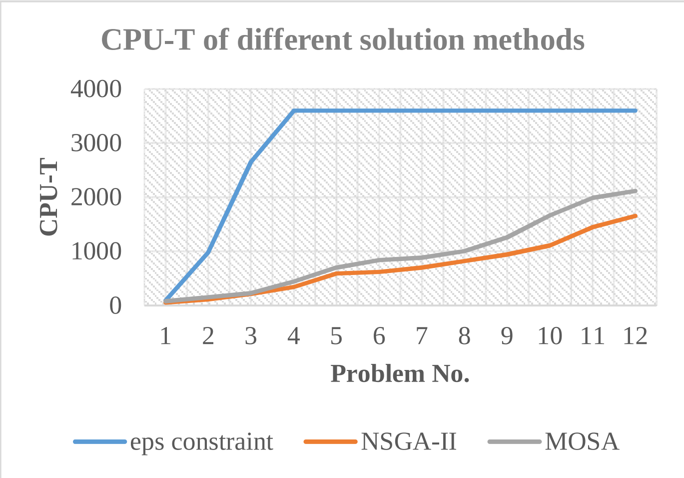
Fig. 18. Pareto front generated by NSGA-II and MOSA.

As it is clear in Figure 18, all Pareto solutions generated in NSGA-II are better than Pareto solutions generated in MOSA.

In terms of the other criteria for comparing these two algorithms, CPU-T is a good criterion. Table 18 and Figure 19 are related to this comparison.

Table 18. CPU-T for different solution methods.

Solution method	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}
ε-constraint	95.1	985.7	2652.5	3600	3600	3600	3600	3600	3600	3600	3600	3600
NSGA-II	58.6	112.5	214.4	345.2	588.9	621.4	698.3	819.7	945.4	1108.9	1449.1	1657.5
MOSA	77.4	154.7	229.6	442.2	701.3	841.6	884.2	1007.2	1258.9	1662.1	1989.6	2117.8



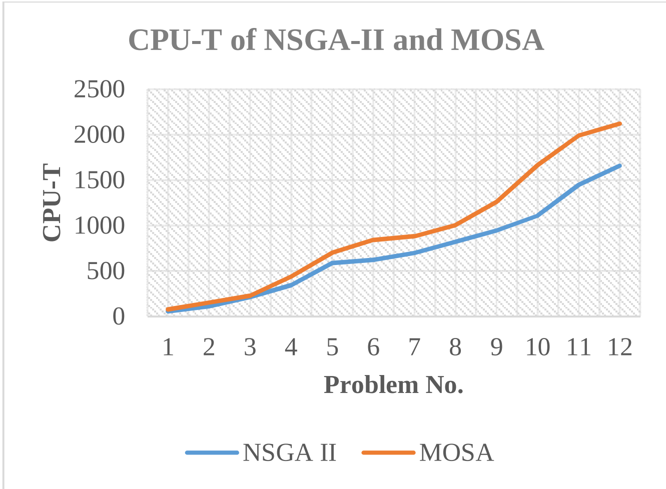


Fig. 19. CPU-T of the proposed solution methods.

According to Table 16 and Figure 19, with an increase in the problem size, CPU-T of the ε-constraint method increases significantly, so that it is unable to find a solution within the run time limitation of 3600 seconds from P_4 onwards. However, as it is clear, NSGA-II needs less time to find the Pareto front, and it can be among other advantages of this algorithm than MOSA. With these interpretations, the offered NSGA-II and MOSA algorithms can be excellent tools for solving large-sized problems. To complete the comparisons between MOSA and NSGA-II, a statistical test is applied to specify the performance of each algorithm on optimizing each objective function. Accordingly, an independent sample T-test is applied with α=5% using SPSS Software. The results are presented in Table 19.

Table 19. Statistical test of the proposed algorithm.

Obj.	**Comparison**	Paired Differences					*t*	*df*	**Sig. (2-tailed)**
		Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference				
					Lower	Upper			
TC	(NSGA-II)-(MOSA)	-61.7200	120.6188	38.14304	-148.0055	-24.56555	-1.618	9	.140
TR	(NSGA-II)-(MOSA)	7.95300	22.12637	6.99697	2.37812	7.87525	-1.137	9	.285

According to Table 19, the significance (Sig.) value is greater than α in both cases and it indicates that two algorithms has a similar performance in achieving suitable values for TC and TR simultaneously. On the other hand, the average difference of TC values is -61.72 which indicate that MOSA reports solutions with higher TC value. Moreover, this criteria is 7.95 for TR and it indicates that NSGA-II reports solutions with higher TR. Accordingly, it can be concluded that NSGA-II has a relative superiority over the MOSA algorithm.

7. Case Study

In this subsection, a case study is conducted in a dairy supply chain in Iran (*Kalleh Dairy Company*) with 4 suppliers, 33 customers, and 2 cross-docks such that the goal is to find the best planning for supplying Tehran and the countryside in order to justify the applicability of the model and superior solution method. Furthermore, 2 and 4 vehicles are available at the delivery and pick-up sections, respectively.

The study is investigated for milk as the main product. All the input information has been gathered from the company's experts at the main site which includes the historical data for 12 periods (months) (2016-2017). In fact, the aim is to implement the proposed model for this case and help the company's managers to identify the optimal policy, compare the current status with the optimal status which can be useful in their strategies implementing and future planning. The described supply chain network is depicted in Figure 20.

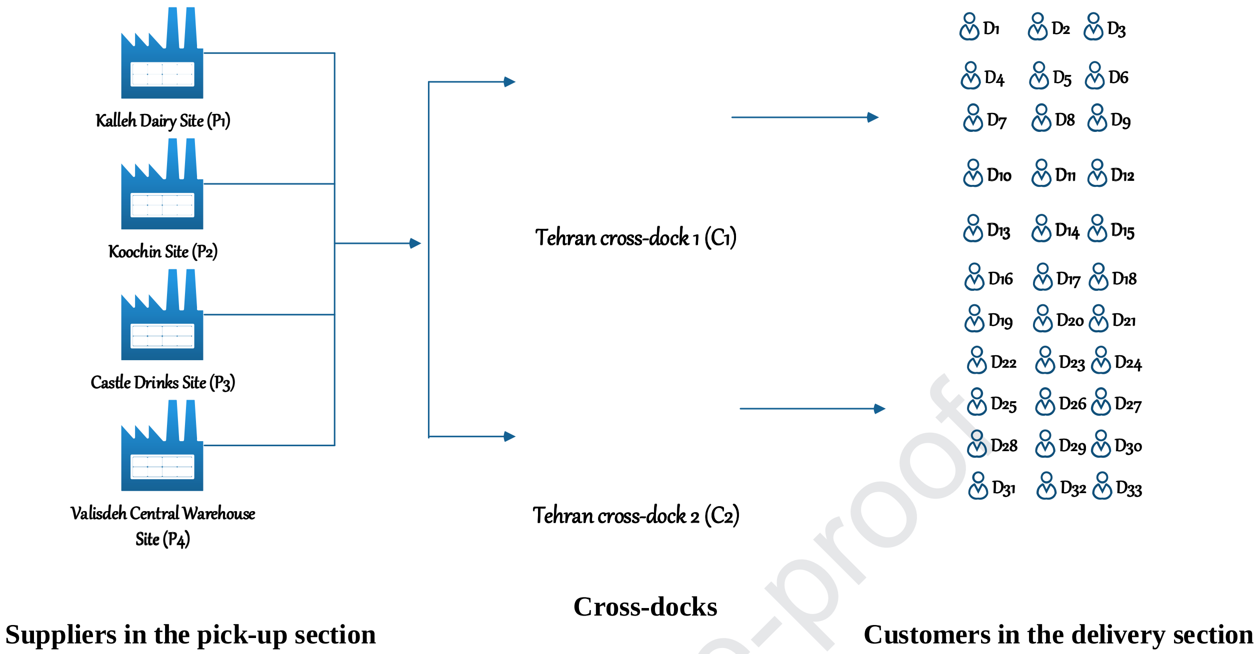


Fig. 20. The considered supply chain network of Kalleh Dairy Company.

After solving the problem using NSGA-II algorithm, the obtained Pareto front is illustrated in Figure 21.

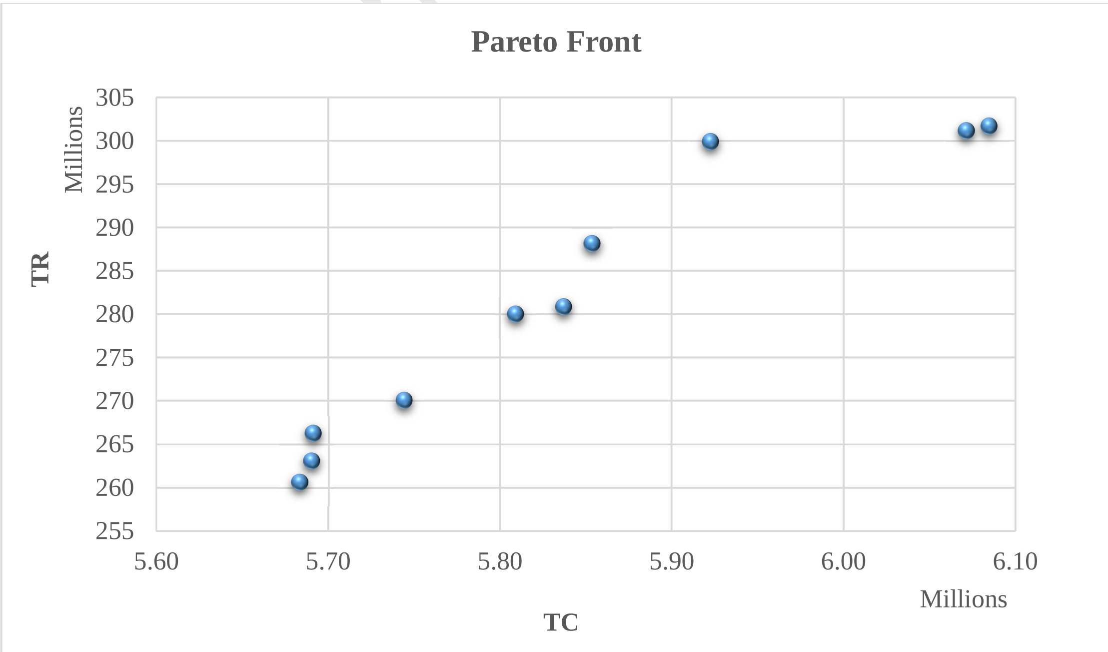


Fig. 21. Pareto front generated by NSGA-II algorithm for case study problem.

As it is shown in Figure 20, different Pareto solutions are proposed for the case study problem. All of these solutions are analyzed and the best one is chosen based on the experts' and managers' opinion using SAW method which has the value of 5922381.07 $ for TC and the value of 299951253.02 units for TR. The weights of 0.60 and 0.40 were set for TC and TR objectives, respectively. As an example, the best planning of the case study problem for the first time period is given by Table 20.

Table 20. Optimal route planning of the case study problem for the first time period.

Section	Route planning
Delivery	P1→C1→C2→P1 P2→P3→P4→C1→C2→P2
Pick-up	C1→D4→D7→D1→D8→D19→D25→D26→D30→D27→C1 C1→D29→D2→D28→D13→D15→D31→D20→D10→C1 C2→D3→D33→D6→D9→D14→D11→D16→D24→D29→C2 C2→D12→D17→D21→D18→D22→D19→D5→C2

In the following, in order to provide the management enlightenment, sensitivity analysis of the demand parameter is performed. Accordingly, the behavior of the objective functions is investigated against the changes of the parameter and useful managerial insights can be derived. Hence, q_{it} is considered as the demand of i^{th} customer for pick-up at t^{th} period. The change interval of the parameter is taken into account from -20% to +20% and the best Pareto solution determined by the SAW method is reported for each change interval. The obtained results are shown in Table 21 and Figure 22.

Table 21. Sensitivity analysis of the demand parameter.

Objective function	Change intervals of q_{it}				
	-20%	-10%	0%	+10%	+20%
TC	5837175.77	5915887.62	5922381.07	6061847.22	6285165.673
TR	290366448	290646727	299951253	307986947	300666128.2

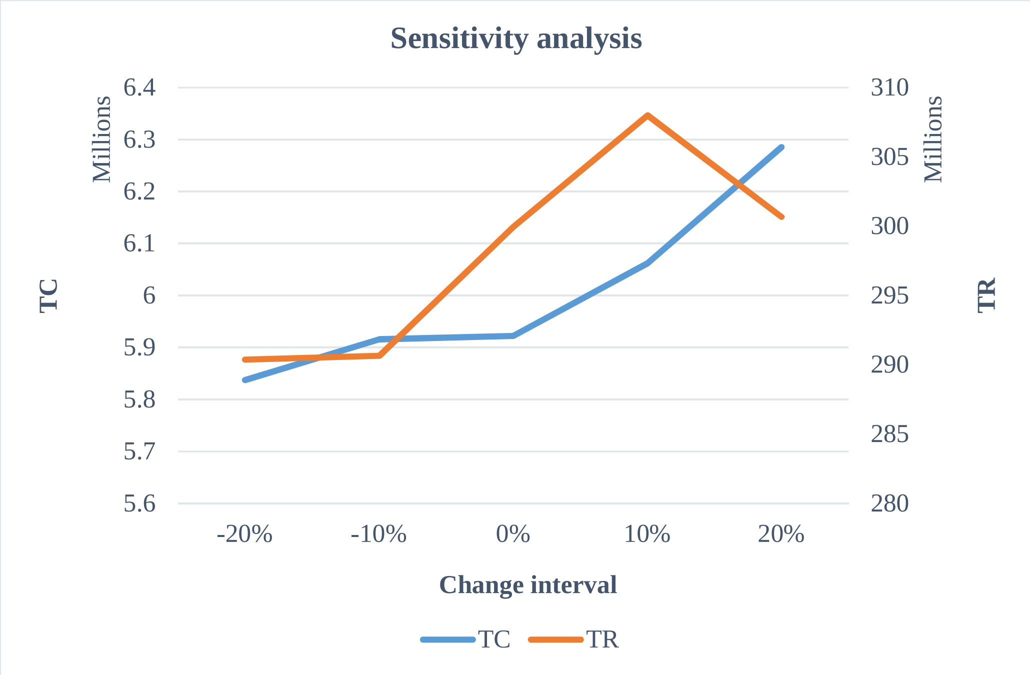


Fig. 22. Sensitivity analysis of the demand parameter.

According to the results obtained by the sensitivity analysis, it can be found that the objective functions have no regular behaviors against change intervals and different sensitivities are generated. For example, the most significant change in TC is related to the interval of 20% increase in the parameter, and the most significant change in TR is related to the interval of 10% decrease of the parameter. Management can determine the optimal policy by evaluating the behavior of the total cost and reliability values and according to the changes in the values of the parameters. Implementing the optimal policy in line with increasing the citizens' satisfaction through reliability maximization, minimizing the total network cost, and consequently, providing a special attention to the pollution will lead to a more sustainable supply chain.

8. Conclusion and Outlook

Cross-docking plays a significant role in augmenting the efficiency of large-scale supply chain distribution networks. Against the mechanism of traditional warehouses, cross-docks hold little or no inventory due to its nature. Instead, products from incoming vehicles are unloaded and directly transferred through the cross-dock to outgoing vehicles. Therefore, cross-docks lead to reducing inventory holding costs and shortening lead times from suppliers to customers. On the other hand, minimizing the transportation cost by providing optimal route planning, paying attention to the side-effect of vehicle transportation, namely, CO_2 emissions and increasing the customers' satisfaction by reliability maximization lead to a more sustainable supply chain management. These were the main reasons that motivated the authors to introduce and study the reliable PRP-CDS considering sustainable development. Accordingly, a BOMILP model was developed to formulate the problem based on real-world assumptions and to concurrently minimize the total cost (pollution and transportation costs) and maximize the supply reliability. Due to the problem's NP-hardness, two efficient algorithms were developed, namely, MOSA and NSGA-II to solve the problem. The Taguchi design method was applied to adjust the parameters of the proposed algorithm. The algorithms were tested by solving several different sized instance problems and compared with the ε-constraint method as an exact method, which was run using GAMS software with CPLEX solver in small-sized problems. The proposed algorithms were compared with each other through computing MID, SM, DM and CPU-T measures in different-sized problems. Moreover, a statistical test was applied to evaluate the performance of NSGA-II and MOSA on optimizing each objective. Finally, a real case study in Iran was investigated to assess the applicability of the proposed methodology.

The following important conclusions are yielded based on the computational results:

(1) In order to provide a link between the research contributions and numerical results, it can be claimed that sustainable development by optimizing TC and TR in the PRP-CDS is not found in the literature, the numerical results tried to demonstrate the relationship between these two important objectives by generating and analyzing Pareto solutions,

(2) Numerical results examined the performance of NSGA-II and MOSA against the ε-constraint method from different aspects. The obtained Pareto fronts in small-sized problems showed that the applied meta-heuristic algorithms can provide more scattered fronts and have acceptable quality which validates their performance,

(3) For better comparison, MID, SM, DM, SAW and CPU-T measures were examined in 4 different small-sized problems and 8 different large-sized problems. The results demonstrated that the suggested NSGA-II and MOSA algorithms are very close to each other in terms of different measures and both can provide more efficient Pareto fronts against the ε-constraint method,

(4) Statistical test was employed to choose the best algorithm of the study. The applied T-test revealed that the proposed NSGA-II algorithm was able to report Pareto solutions with lower TC and higher TR on average,

(5) Case study problem was implemented by the NSGA-II algorithm as the best solution tool and the results included the optimal route planning in delivery and pick-up section were determined and the values of objectives were reported,

(6) Based on the sensitivity analysis, it was revealed that the objective functions are directly dependent on the demand parameter and the management can evaluate the required available resources based on the fluctuations of demand in a real-world situation.

The main limitations of the study can be interpreted as the consideration of deterministic conditions and infinite capacity of cross-docks in the problem. Moreover, in the algorithms, the limitation is the use of random-based operators which causes to achieve a different final solution in each run. Accordingly, they need to be run several times for finding the best possible solutions. So, as one of the main future research directions, *uncertainty* can be considered in the problem for different parameters such as demand as the most possible parameter by applying known uncertainty tools including robust optimization, chance-constrained programming, stochastic programming and stochastic optimal control. On the other hand, capacitated cross-dock selection, multiple products and multiple types of vehicles with different features of pollution and locational decisions may be studied in the problem to make it more real and more valuable. Moreover, other multi-objective metaheuristics such as multi-objective particle swarm optimization (MOPSO) can be applied to test the proposed algorithms. Finally, Benders Decomposition Algorithm (BDA) can be applied to the model to resolve the randomness nature of the proposed metaheuristic algorithms.

Appendix

The computational equations of some input parameters are represented in Table A. The other parameters get assigned values directly.

Table A. Computational equations of parameters.

Parameters	Computational equations
λ	$\lambda = \frac{\xi}{\kappa\psi}$
γ	$\gamma = \frac{1}{1000\,\varepsilon\varpi}$

Parameters	Computational equations
β	$\beta = \frac{c_d\ A\ \rho}{2}$
θ^{mr}	$\theta^{mr} = \begin{cases} 0, & m = 1,2 \\ \frac{v^{2r} - v^{3r}}{v^{3r}}, & m = 3 \end{cases}$
η_{ij}^{mr}	$\eta_{ij}^{mr} = \begin{cases} \frac{d_{ij}}{v^{1r}}, & m = 1 \\ \frac{d_{ij}}{v^{3r}} + \frac{v^{3r} - v^{2r}}{v^{2r}} a, & m = 2 \\ \frac{d_{ij}}{v^{3r}}, & m = 3 \end{cases}$

References

Abad, H.K.E., Vahdani, B., Sharifi, M., and Etebari, F. A bi-objective model for pickup and delivery pollution-routing problem with integration and consolidation shipments in cross-docking system. Journal of Cleaner Production, 2018. 193: p. 784-801.

Ahmadizar, F., Zeynivand, M. and Arkat, J. Two-level vehicle routing with cross-docking in a three-echelon supply chain: A genetic algorithm approach. Applied Mathematical Modelling, 2015. 39(22): p. 7065-7081.

Baniamerian, A., Bashiri, M., and Tavakkoli-Moghaddam, R. Modified variable neighborhood search and genetic algorithm for profitable heterogeneous vehicle routing problem with cross-docking. Applied Soft Computing, 2019. 75: p. 441-460.

Bérubé, J.F., Gendreau, M., and Potvin, J.Y. An exact ε-constraint method for bi-objective combinatorial optimization problems: Application to the Traveling Salesman Problem with Profits. European Journal of Operational Research, 2009. 194(1): p. 39-50.

Boysen, N. and Fliedner, M. Cross-dock scheduling: Classification, literature review and research agenda. Omega, 2010. 38(6): p. 413-422.

Buijs, P., Vis, I.F., and Carlo, H.J. Synchronization in cross-docking networks: A research classification and framework. European Journal of Operational Research, 2014. 239(3): p. 593-608.

Chen, P., Guo, Y., Lim, A., and Rodrigues, B. Multiple crossdocks with inventory and time windows. Computers and operations research, 2006. 33(1): p. 43-63.

Dalla Torre, A., Gallo, P., Gubiani, D., Marshall, C., Montanari, A., Pittino, F., and Viel, A. A map-matching algorithm dealing with sparse cellular fingerprint observations. Geo-spatial Information Science, 2019. 22(2): p. 89-106.

Dantzig, G.B. and Ramser, J.H. The truck dispatching problem. Management science, 1959. 6(1): p. 80-91.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T.A.M.T., A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE transactions on evolutionary computation, 2002. 6(2): p. 182-197.

Dondo, R., Méndez, C.A. and J. Cerdá. The multi-echelon vehicle routing problem with cross-docking in supply chain management. Computers and Chemical Engineering, 2011. 35(12): p. 3002-3024.

Dondo, R. and Cerdá, J. A sweep-heuristic based formulation for the vehicle routing problem with cross-docking. Computers and Chemical Engineering, 2013. 48: p. 293-311.

Dondo, R. and Cerdá, J. A monolithic approach to vehicle routing and operations scheduling of a cross-dock system with multiple dock doors. Computers and Chemical Engineering, 2014. 63: p. 184-205.

Franceschetti, A., Honhon, D., Van Woensel, T., Bektaş, T., and Laporte, G. The time-dependent pollution-routing problem. Transportation Research Part B: Methodological, 2013. 56: p. 265-293.

Goli, A., Tirkolaee, E.B., Malmir, B., Bian, G.B., and Sangaiah, A.K. A multi-objective invasive weed optimization algorithm for robust aggregate production planning under uncertain seasonal demand. Computing, 2019. 101(6): p. 499-529.

Glover, F.W. and Kochenberger, G.A. Handbook of metaheuristics. Vol. 57. 2006: Springer Science and Business Media.

Hammer, M., Deep Change: how operational innovation can transform your company. Harvard Business Review, 2004. 82: 84-96.

Kinnear, E., Is there any magic in cross-docking? Supply Chain Management: An International Journal, 1997. 2(2): p. 49-52.

Kubotani, H., and Yoshimura, K. Performance evaluation of acceptance probability functions for multi-objective SA. Computers and Operations Research, 2003. 30(3): p. 427-442.

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

Liao, C.J., Lin, Y., and Shih, S.C. Vehicle routing with cross-docking in the supply chain. Expert Systems with Applications, 2010. 37(10): p. 6868-6873.

Ma, H., Miao, Z., Lim, A., and Rodrigues, B. Crossdocking distribution networks with setup cost and time window constraint. Omega, 2011. 39(1): p. 64-72.

Maknoon, Y. and Laporte, G. Vehicle routing with cross-dock selection. Computers and Operations Research, 2017. 77: p. 254-266.

Mardani, A., Kannan, D., Hooker, R.E., Ozkul, S., Alrasheedi, M., and Tirkolaee, E.B. Evaluation of green and sustainable supply chain management using structural equation modelling: A systematic review of the state of the art literature and recommendations for future research. Journal of Cleaner Production, 2020. 249: p. 119383.

Miao, Z., Yang, F., Fu, K., and Xu, D., Transshipment service through crossdocks with both soft and hard time windows. Annals of Operations Research, 2012. 192(1): p. 21-47.

Moghadam, S.S., Ghomi, S.F., and Karimi, B. Vehicle routing scheduling problem with cross-docking and split deliveries. Computers and chemical engineering, 2014. 69: p. 98-107.

Mousavi, S.M. and Vahdani, B. A robust approach to multiple vehicle location-routing problems with time windows for optimization of cross-docking under uncertainty. Journal of Intelligent and Fuzzy Systems, 2017. 32(1): p. 49-62.

Musa, R., Arnaout, J.P., and Jung, H. Ant colony optimization algorithm to solve for the transportation problem of cross-docking network. Computers and Industrial Engineering, 2010. 59(1): p. 85-92.

Nam, D. and Park, C.H. Multiobjective simulated annealing: A comparative study to evolutionary algorithms. International Journal of Fuzzy Systems, 2000. 2(2): p. 87-97.

Napolitano, M., Cross-dock fuels growth at Dots. Logistics Management (Highlands Ranch, Colo.: 2002), 2011. 50(2).

Nasiri, M.M., Rahbari, A., Werner, F., and Karimi, R. Incorporating supplier selection and order allocation into the vehicle routing and multi-cross-dock scheduling problem. International Journal of Production Research, 2018: p. 1-26.

Nikolopoulou, A.I., Repoussis, P.P., Tarantilis, C.D., and Zachariadis, E.E. Moving products between location pairs: Cross-docking versus direct-shipping. European Journal of Operational Research, 2017. 256(3): p. 803-819.

Pasandideh, S.H.R., Niaki, S.T.A., and Asadi, K. Optimizing a bi-objective multi-product multi-period three echelon supply chain network with warehouse reliability. Expert Systems with Applications, 2015. 42(5): p. 2615-2623.

Rahbari, A., Nasiri, M.M., Werner, F., Musavi, M., and Jolai, F. The vehicle routing and scheduling problem with cross-docking for perishable products under uncertainty: Two robust bi-objective models. Applied Mathematical Modelling, 2019. 70: p. 605-625.

Rajabi, M. and Shirazi, M.A. Truck scheduling in a cross-dock system with multiple doors and uncertainty in availability of trucks. Journal of Applied Environmental and Biological Sciences, 2016. 6(7S): p. 101-109.

Razmi, J., Zahedi-Anaraki, A., and Zakerinia, M. A bi-objective stochastic optimization model for reliable warehouse network redesign. Mathematical and Computer Modelling, 2017. 58(11-12): p. 1804-1813.

Sangaiah, A. K., Goli, A., Tirkolaee, E. B., Ranjbar-Bourani, M., Pandey, H. M., and Zhang, W. Big Data-Driven Cognitive Computing System for Optimization of Social Media Analytics. IEEE Access, 2020. 8: p. 82215-82226.

Santos, F.A., Mateus, G.R., and Da Cunha, A.S. The pickup and delivery problem with cross-docking. Computers and Operations Research, 2013. 40(4): p. 1085-1093.

Srinivas, N. and Deb, K. Muiltiobjective optimization using nondominated sorting in genetic algorithms. Evolutionary computation, 1994. 2(3): p. 221-248.

Stalk, G., Evans, P., and Shulman, L.E. Ompeting On Copobilities: The NeW RUleS Of COrpOfote Strotegy. Harvard business review, 1992.

Stephan, K. and Boysen, N. Cross-docking. Journal of Management Control, 2011. 22(1): p. 129.

Sung, C.S. and Song, S.H. Integrated service network design for a cross-docking supply chain network. Journal of the Operational Research Society, 2003. 54(12): p. 1283-1295.

Sung, C. and Yang, W. An exact algorithm for a cross-docking supply chain network design problem. Journal of the Operational Research Society, 2008. 59(1): p. 119-136.

Taguchi, G. Introduction to quality engineering: designing quality into products and processes. 1986.

Tirkolaee, E.B., Goli, A., Hematian, M., Sangaiah, A.K., and Han, T. Multi-objective multi-mode resource constrained project scheduling problem using Pareto-based algorithms. Computing, 2019a. 101(6): p. 1-24.

Tirkolaee, E.B., Mardani, A., Dashtian, Z., Soltani, M., and Weber, G.W. A novel hybrid method using fuzzy decision making and multi-objective programming for sustainable-reliable supplier selection in two-echelon supply chain design. Journal of Cleaner Production, 2020a. 250: p. 119517.

Tirkolaee, E. B., Goli, A., and Weber, G. W. Fuzzy Mathematical Programming and Self-Adaptive Artificial Fish Swarm Algorithm for Just-in-Time Energy-Aware Flow Shop Scheduling Problem with Outsourcing Option. IEEE Transactions on Fuzzy Systems, 2020b.

Tirkolaee, E. B., Mahdavi, I., Esfahani, M. M. S., and Weber, G. W. A robust green location-allocation-inventory problem to design an urban waste management system under uncertainty. Waste Management, 2020c. 102: p. 340-350.

Van Belle, J., Valckenaers, P., and Cattrysse, D. Cross-docking: State of the art. Omega, 2012. 40(6): p. 827-846.

Varadharajan, T.K., and Rajendran, C. A multi-objective simulated-annealing algorithm for scheduling in flowshops to minimize the makespan and total flowtime of jobs. European Journal of Operational Research, 2005. 167(3), p. 772-795.

Vincent, F.Y., Jewpanya, P., and Redi, A.P. Open vehicle routing problem with cross-docking. Computers and Industrial Engineering, 2016. 94: p. 6-17.

Wang, J., Jagannathan, A.K.R., Zuo, X., and Murray, C.C. Two-layer simulated annealing and tabu search heuristics for a vehicle routing problem with cross-docks and split deliveries. Computers and Industrial Engineering, 2017a. 112: p. 84-98.

Wang, Y., Ma, X. L., Li, Z. B., Liu, Y., Xu, M. Z., Wang, Y. H. Profit Distribution in Collaborative Multiple Centers Vehicle Routing Problem. Journal of Cleaner Production, 2017b, 144: 203-219.

Wang, Y., Assogba, K., Liu, Y., Ma, X. L., Xu, M. Z., Wang, Y. H. Two-echelon location-routing optimization with time windows based on customer clustering. Expert Systems with Applications, 2018a, 104: 244-260.

Wang, Y., Zhang, S. L., Assogba, K., Fan, J. X., Xu, M. Z., Wang, Y. H. Economic and environmental evaluations in the two-echelon collaborative multiple centers vehicle routing optimization. Journal of Cleaner Production, 2018b, 197: 443-461.

Wen, M., Larsen, J., Clausen, J., Cordeau, J.F., and Laporte, G. Vehicle routing with cross-docking. Journal of the Operational Research Society, 2009. 60(12): p. 1708-1718.

Zitzler, E., Deb, K., and Thiele, L. Comparison of multiobjective evolutionary algorithms: Empirical results. Evolutionary computation, 2000. 8(2): p. 173-195.

Highlights

- Developing a novel bi-objective MILP model to integrate reliable PRP and CDS,
- Studying the effects of traffic conditions in transportation and pollution emission,
- Designing high-quality algorithms including NSGA-II and MOSA to solve the problem,
- Applying the Taguchi design method to enhance the performance of the algorithms,
- Evaluating the applicability of the proposed methodology using a case study in Iran.

Declaration of interests

☒ 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.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: