

Multiple cross-docks scheduling with multiple doors using fuzzy approach and meta-heuristic algorithms

Mitra Movassaghi¹, Soroush Avakh Darestani^{1,2*}

¹ Department of Industrial Engineering, Faculty of Industrial and Mechanical Engineering, Islamic Azad University, Qazvin Branch, Qazvin, Iran

² Guildhall School of Business and Law, London Metropolitan University, London, United Kingdom

s.avakhdarestani@londonmet.ac.uk

Abstract

The issue of supply chain in today's world is a major competitive advantage in reducing costs. Supply chain includes procurement, logistics and transportation, marketing, organizational behavior, networking, strategic management, information systems management and operations management. One of the most important practices in logistics is Cross-Docking which sets its goals as inventory reduction and customer satisfaction increase. Customers receive goods through docks. Docks are responsible to provide a place for goods before being delivered to the customers. Then, these materials are directly loaded into outbound trucks with little or no storage in between to send to customers in the shortest possible time. This paper is mainly aimed at introducing a mixed integer linear programming model to solve scheduling several cross-docking problems. The proposed model is highly facilitated to allocate the optimal destinations to storage doors and truck scheduling in docks while selecting the collection and delivery routes. Using optimization approaches at uncertainty conditions is also of great importance. Mathematical programming techniques vividly fail to solve transportation problems that include fuzzy objective function coefficients. A fuzzy multi-objective linear programming model is proposed to solve the transportation decision-making with fuzzy objective function coefficients in this paper. On the other hand, the existences of computational complexities lead this model to be categorized as a NP-Hard one. Therefore, we applied Meta-heuristic Algorithms such as Genetic and Ant Colony in order to solve our proposed problem.

Keywords: Cross Docks Scheduling, Fuzzy Logic, Genetic and Ant Colony Meta-heuristic

1. Introduction

Existing technologies are hastily moving towards the specialization and globalization. In order to survive in this field, suppliers need to be capable of responding consumer demands at different conditions. In today's competitive environment, the significant role of distribution centers in timely delivery of goods and inventory cost reduction has attracted the attention of numerous supply-chain managers. Many companies use Cross-docking as a logistic strategy to ensure storage costs reduction and customer satisfaction improvement within a shorter delivery lead-time. Space requirements, inventory warehousing costs, labor intensive, and order picking tasks can be mentioned among the main reasons behind the high expense of goods storage. Cross-docking is the true tool to eliminate a large portion of such warehousing costs. A cross-dock is defined as an I-shaped facility with strip and stack dock doors located at opposite sides of the terminal and minimal storage space in between. Strip docks located in one side of the distribution terminal receive inbound shipments arriving at the cross-dock. As soon as the inbound trucks are unloaded, the freights are screened and sorted by destination. Then, a forklift or a conveyor belt is used to move them across the terminal via their designated stack dock doors and here is where the loads are charged into departing trucks to be carried to their destinations.

Since workers are responsible for unloading, sorting, and transferring a wide variety of loads from incoming trucks to outgoing trailers, freight handling is vividly considered as a labor intensive and costly task in a cross-dock terminal.

Products with the best matching to cross-docking include (a) products with a stable demand; (b) perishable bulk materials, consist of some chemical and food compounds, requiring immediate shipment; (c) frozen foods and other refrigerated products like pharmaceuticals that should be directly moved from cooled inbound to cooled outbound trucks to keep the cooling chain unbroken; (d) high-quality items of low quality inspection requirements during the receiving process and (e) ready pre-tagged products for being sold to the customers. Furthermore, hazardous chemicals drums and waste materials containers are aggregated at cross-dock facilities and immediately transferred to remedy sites for treatment and disposal. Cross docking is regarded as a tool for pharmaceutical, food and chemical industries day in day out in order to achieve more competitive advantages. In practical approach, successful implementation of cross-docking strategies are evident in chemical and manufacturing companies such as Eastman Kodak Co., Goodyear GB Ltd. and Toyota [1]. Comprehensive reviews on cross-docking was also offered by [1] and [2].

Both location and physical layout of a cross-dock facility have been subject to many researches so far regarding shape and number of dock doors, and related truck scheduling. Routing aspects of the problem were neglected, though. Operational issues at the cross-dock terminal are considered in the truck scheduling (TS) problem and are mainly addressed in assigning vehicles to dock doors, the processing sequence of trucks at every strip and stack door and transferring goods from inbound to outbound vehicles. The presence of a temporary storage is always necessary despite the fact that cross-docking is supposed to unload inbound trucks and immediately reload the freights into delivery vehicles. The absence and impossibility of a perfect synchronization in limited numbers of pickup and delivery trucks lead goods to fail in arriving at the cross-dock in the sequence they must be reloaded into the departing vehicles. Tsui and Chang [3] proposed a bilinear programming model to deal with the truck scheduling (TS) in an early work. In their research, the workers efficiency extremely depend on the cross-dock plan and how trailers are assigned to dock doors. Bartholdi and Gue [4] presented suitable attributes of an effective cross-dock plan and proposed a model that also considered the dock door assignment problem. In the model, the objective is to minimize the transfer time, material handling and compaction. Yu and Egbelu [5] presented two approaches for trucks scheduling at the dock and found a better items exchange between inbound and outbound trucks. One of the main features of their model is the concurrent designation of goods transfer among trucks and the docking sequences of inbound and outbound trucks. Li [6] also focused on truck scheduling and door assignment regarding a multi-door cross-dock and with more trucks than the doors and as a result the lines of trucks waiting for an empty door to start unload/offload operations. A mixed integer programming (MIP) model from a door scheduling standpoint, and a dependence ranking search (DRS) heuristic algorithm were presented as the two major approaches to solve the presented model and minimize the total cross-dock operation time. They concluded that DRS heuristic algorithm is able to find good solutions in a lot of shorter solution times while the MILP model cannot be used for practical instances due to the high computational cost.

Wisittipanich and Hengmeechai [7] proposed a mathematical model of mixed integer programming for door assigning and truck sequencing in a multi-door cross docking system. Their model aimed at minimizing total operational time or makespan. Serrano et al. [8] presented a mixed integer linear programming model to schedule inbound trucks' arrival times, shop-floor repackaging operations and outbound trucks' departure times. The objective of the model is to minimize penalty costs related to inbound trucks' arrival times and therefore unstable capacity of the repackaging factory. Molavi et al. [9] developed a mixed integer programming model with the hybrid genetic algorithm-reduced variable neighborhood search (HGARVNS) algorithm to solve the problem in medium and large sized scales. Amini and Tavakkoli-Moghaddam [10] presented a bi-objective linear mathematical model. A complete enumeration method is applied to find optimum solutions subject to the complication of large-scale problems, and they correct three multi-objective meta-heuristics; namely, Non-dominated Sorting Genetic Algorithm II (NSGA-II), Multi-Objective Simulated Annealing (MOSA) and Multi-Objective Differential Evolutionary (MODE). Keshtzari et al. [11] proposed a new mixed integer programming model for truck scheduling in cross-dock problems. Employing commercial optimization solvers, the efficiency of the presented model is analyzed with the available model. Ahkamiraad and

Wang [12] proposed a hybrid of the genetic algorithm and particle swarm optimization (HGP) to solve the formulated NP-hard problems. Small-size problems are solved by HGP. Major examination have been conducted for medium and large-size problems.

Bazgosha et al. [13] presented an integer linear programming model, which is solvable only for small-size examples by CPLEX solver in feasible times. Also, they extended two effective heuristic solution approaches, namely parallel and serial schedule generation schemes. They also extended three metaheuristic methods based on genetic algorithm, particle swarm optimization and cuckoo optimization algorithm. Hasani Goodarzi and Zegordi [14] proposed a location- routing problem for cross-docking system. The purpose is to specify the location of cross-docks, allocating suppliers to them and routing determination, so that the location cost and total shipping cost are minimized. The proposed model is NP-hard problems. Thus, a metaheuristic algorithm known as Biogeography-based optimization (BBO) is applied to solve the problem. Assadi and Bagheri [15] developed a mixed integer programming model. They solved optimally in small-sized cases. Also to solve medium to large-sized instances, two metaheuristics named Differential evolution and Population-based simulated annealing are applied. Maknoon et al. [16] presented a mathematical model for cross-docking problems. They developed a successive priority-based heuristic algorithm to deal with applied problems. Conclusion show the stability of the heuristic method for fairly large size problems. Azimi [17] developed a new approach known as "on-line docking" in an actual container port which is the principal contribution of the research. In the model, the objective is to minimize the average yearly system costs by assigning the best number of inbound-outbound docks and the fleet size for the interior transportations. Motaghedi-Larijani and Aminnayeri [18] presented a model to optimize the number of outbound doors based on minimizing the total costs, including the costs of adding a new outbound door and the expected waiting time of customers.

BolooriArabani et al. [19] evaluated five different meta-heuristics such as the genetic algorithm (GA), the tabu search (TS), the particle swarm optimization (PSO), the ant colony optimization (ACO) and the differential evolution (DE) algorithms, by testing them within a large number of samples. Taking into account delivery and pickup time, warehouse capacities and inventory-handling costs, Chen et al. [20] studied the truck scheduling problem for a network of cross-docks. Local search techniques like simulated annealing and Tabu search were applied to solve their proposed model and concluded that the heuristics outperform optimization models for providing good solutions in realistic time scales. Considering both cross-docking operations and vehicle routing problems, Lee et al. [21] developed an MILP formulation that assumes the arrival of all the vehicles from suppliers at the cross-dock aiming at avoiding vehicle waiting times at this point. Wen et al. [22] forced pickup and delivery tasks to be started within specific time windows using a mixed integer programming formulation for the VRPCD problem aiming at minimizing the travelled distance. The transportation requests were defined in terms of two locations: the pickup node where the freight is loaded and the delivery node to which is destined. Miao et al. [23] considered both soft and hard time windows using a multi cross-dock transshipment problem regarding fixed transportation schedules in the suppliers' flows toward the customers via the cross-docks. Cargoes can be delayed and consolidated in cross-docks, and both suppliers and customers may alternatively have hard time windows or less-restrictive soft time windows. The total cost of multi cross-dock distribution networks that includes transportation, inventory handling and penalty expenses is minimized using adaptive Tabu search and adaptive genetic algorithm as two solution methods for NP-Hard problems.

Logistic processes imply the presence of a large number of different types of risks, primarily in the fields of transport, transshipment, and storage of goods. The main reason for this fact is the presence of numerous participants of logistic systems, the existence of various interactions between large numbers of subsystems or subprocesses, which causes disturbances and uncertainties, both locally and at the system level [24]. Gajovic et al. [24] presented a developed fuzzy logic model based on the analytic hierarchy process (AHP) model and fuzzy analytic hierarchy process (FAHP). Avila-Torres et al. [25] proposed an integrated mathematical model for the frequency calculation and departure time problem considering uncertainty in demand and travel time.

Dondo and Cerdá [26] assumed an unlimited number of dock doors and proposed a monolithic formulation for VRPCD to determine the pickup and the delivery routes simultaneously with the truck scheduling at the cross-dock terminal. They defined a set of constraints to assign vehicles to pickup/delivery routes with the help of an incorporated version of sweep heuristic algorithm in the

MILP model. The results showed the merit of this algorithm in finding near optimal solutions to large problems at very acceptable CPU times. Nevertheless, the negligence of dock door assignments and queues of trucks in front of the dock doors is considered as its weak points that can be avoided using a precise coordination among pickup vehicle routes, cross-dock activities and delivery vehicle routes. If a limited number of dock doors is available, their assignment to incoming and outgoing trucks will be a criterion for the efficiency of the cross-dock operations.

To achieve this goal, we presented a new monolithic MILP formulation that integrates the pickup/delivery vehicle routing and scheduling with both the assignment of dock-doors to incoming and outgoing trucks and the managing of truck queues at strip/stack doors. Attracted to the surveys of Dondo and Cerdá [26], we avoided symmetrical solutions by embedding additional constraints imitating the sweeping algorithm to develop an efficient hybrid approach capable of solving medium-size problem instances at acceptable CPU times. In the mentioned cross dock operation planning and scheduling models, we assumed several docks and uncertainty of parameters as one of the decision making challenges. Not only, an efficient scheduling model are provided, but also, finding an optimal approach in the presence of uncertainty is of great importance. Mathematical programming techniques and equations have proved their disabilities in solving transportation decision making problems by fuzzy objective function coefficients. To overcome these problems, we provided a fuzzy-interactive multi-objective linear programming model for solving transportation decision problems by fuzzy objective function coefficients in the present research and concluded its computational flexibility and efficiency, at the end.

2. Problem description and formulation

VRPCD-TS (The Vehicle Routing Problem with Cross-Docking and The Truck Scheduling) problem which is defined as a combinational vehicle routing and cross-dock truck scheduling problem focuses on transporting a set of requests R from pickup to destination points passing through an intermediate cross-dock facility at minimum routing cost. A limited number of receiving (strip) doors RD and shipping (stack) doors SD are assumed in the cross-dock. In order to increase the cross-dock productivity and reduce the handling cost, the dock door to which an inbound (outbound) truck arrives (departs) at (from) the cross-dock, is determined from the very beginning. The truck scheduling (TS) problem seeks to find the optimal assignment of inbound/outbound trucks to dock doors. The majority of the studies on the VRPCD problem consider the same number of dock doors and trucks, so each truck will be assigned to a different door and truck scheduling aspects can be ignored. However, if this condition is not met, the dock doors will be seen as scarce resources that need to be scheduled overtime and lines of trucks waiting for service can arise at every dock door and this is the real so called truck scheduling problem. Sequential manner is offered for VRPCD and the truck scheduling (TS) problems simultaneous solving because of their complexities. However, we didn't categorize this combinational problem into two phases assuming a limited number of dock doors compared to other studies in this scope.

Different from Dondo and Cerdá [26] that studied vehicle routing and scheduling problem by a cross dock, we modeled the vehicle routing and scheduling problem using several cross docks in this paper. Due to the lack of data completeness and availability, decision-making of cross docks operation scheduling and planning generally face inaccurate data as well as transport planning [27].

One of the inevitable challenges that we encounter while making decisions on cross dock operation scheduling and planning problems is uncertainty of parameters. Therefore, there should be proper approaches to lead us to optimized solutions in uncertainty situations besides the presence of an efficient timing model. Parameters such as costs, demand, and production capacity are very likely to be uncertain in the cross dock scheduling problem [28].

Supply chain planning researches mainly focus on potential distribution relying on previous data to address uncertainties. Probable models may not be the best choice because of the lack of availability and also reliability of the previous statistical data [29]. In the contrary, fuzzy set and possibility theories proved to be superior to probability theories in facing an uncertain supply chain, besides their simplicity and no requirement for data collection [30]. Baykasoglu and Göçken [31] classified fuzzy mathematical programming problems detecting 15 different types of fuzzy mathematical programming models and provided different solution approaches for each type. Mathematical programming techniques and equations proved not to be capable of solving transportation decision-making problems by fuzzy

objective function coefficients. To get rid of this deficiency and respect computational flexibility and efficiency, we proposed a fuzzy-interactive multi-objective linear programming model for solving transportation decision problems by fuzzy objective function coefficients.

2.1. Problem assumptions

Inspired by Dondo and Cerdá [26] the mathematical formulation has been developed based on the following assumptions.

1. Goods are transported from suppliers to destinations by a homogeneous vehicle fleet through a single cross-dock terminal.
2. The well-known layout for cross-dock includes a specific number of strip and stack dock doors.
3. At the beginning of the planning horizon, all vehicles are assumed to be available, accomplish the required pickup tasks and subsequently perform the delivery tasks.
4. Dock doors are exclusively dedicated to either unloading or loading operations, e.g. they are designated as either strip or stack dock doors.
5. The number of strip/stack doors can be lower than the number of vehicles. Then, the dock doors can be regarded as scarce resources that should be scheduled over time.
6. Each P/D request must be serviced by a single vehicle, i.e. orders are not split table.
7. The loading/unloading of a truck at the cross-dock cannot be interrupted, i.e. no pre-emption is allowed.
8. The freights unloaded at the cross-dock are not interchangeable, i.e. each one must be sent to a specific destination.
9. The amounts of loaded or unloaded goods at supply/delivery locations are given.
10. Each vehicle is allowed to service more than one pick-up/delivery location.
11. The starting and ending point for the pickup and delivery routes are set to the cross-dock.
12. The total quantity of goods carried by a vehicle must not exceed its capacity.
13. The sum of a fixed stop time (ft_r^P/ft_r^D) and a variable component determines the service time at supply/delivery locations and is increased with the size of the cargo q_r to be picked-up/delivered at a rate l_r/u_r .
14. The goods picked up and delivered by the same truck are not unloaded at the cross-dock and remain inside the vehicle.
15. The total amount of goods unloaded on the receiving docks and the total freight loaded on trucks at the shipping doors must be equal at the end of the planning horizon. Therefore, there is no final inventory left at the cross-dock.

Sets:

N : unload events

R : requests

RD : receiving (strip) dock doors

SD : shipping (stack) dock doors

V : vehicles

W : cross-docks

Parameters:

In the real world making decisions on cross-dock operation scheduling and planning problems often involve inaccurate data due to incomplete information or unavailability of data. Uncertainty and changes such as fluctuations in production and demand flows, layout and arrangement of productions inside cross-docks and vehicles, volume of productions and their possible return and their temporary depot inside vehicles and cross-docks, performance of human resources, Traffic of freight routes and the use of alternative routes, etc. cause us to consider the parameters provided in this model fuzzy.

To generalize the proposed model by Dondo and Cerdá [26], we defined the following parameters for cross- docks.

$\widetilde{d}_{r,\hat{r}}^P/\widetilde{d}_{r,\hat{r}}^D$: distance between P/D locations r and \hat{r}

$\widetilde{d}_{r,w}^P/\widetilde{d}_{r,w}^D$: distance between the P/D location r and the cross-dock w

$\widetilde{ft}_r^P/\widetilde{ft}_r^D$: fixed stop time at the P/D site of request r

$\widetilde{ft}_w^P/\widetilde{ft}_w^D$: fixed stop time for P/D activities at the cross-dock terminal w

$\widetilde{lr}_r/\widetilde{ur}_r$: loading/unloading rate at P/D sites of request r
 $\widetilde{lr}_w/\widetilde{lr}_w$: loading/unloading rate at the cross-dock terminal w
 \widetilde{q}_r : shipment size for request r
 \widetilde{Q}_v : vehicle capacity
 \widetilde{Q}_w : cross-dock capacity
 \widetilde{sp}_v : vehicle travel speed
 $\widetilde{tt}_{d,\hat{d}}$: time spent in moving a vehicle from the unloading door $d \in RD$ to the shipping door $\hat{d} \in SD$
 \widetilde{uc}_v : unit distance cost for vehicle v
 μ : cost per unit of spent time for accomplishment of delivery and loading tasks.

Binary variables:

We also defined binary variables of $GP_{w,v}/GD_{w,v}$ for modeling the allocation of vehicles to cross dock as another aspect to generalize Dondo and Cerdá's [26] model.

$DP_{v,d}/DD_{v,d}$: denotes that vehicle v has been allocated to the strip/stack dock door d

$WP_{n,v}/WD_{n,v}$: denotes that the unloading (U) /loading (L) activity of vehicle v is associated to the time event n

$XP_{r,\hat{r}}/XD_{r,\hat{r}}$: establishes the sequencing of pickup (P)/delivery (D) nodes (r,\hat{r}) on the route of the assigned P/D vehicle

$YP_{r,v}/YD_{r,v}$: denotes that vehicle v visits the P/D location of request r

$ZP_{v,\hat{v}}/ZD_{v,\hat{v}}$: sequences vehicles (v,\hat{v}) waiting for service at the same strip/stack door

$GP_{w,v}/GD_{w,v}$: denotes that vehicle v visits the P/D location of cross-dock w

Nonnegative continuous variables:

AT_v^P/AT_v^D : P/D vehicle arrival times of vehicle v at the cross-dock facility

CP_r/CD_r : Cumulative travelling cost from the cross-dock to the P/D site of request r

$DRS_{v,d,\hat{d}}$: denotes that the receiving door $d \in RD$ and the shipping door $\hat{d} \in SD$ have been assigned to vehicle v

OC_v^P/OC_v^D : overall travelling cost for the P/D tour of vehicle v

RT_v^P : time at which vehicle v is released from its pickup duties

ST_v^P/ST_v^D : starting time for the P/D tour of vehicle v

TP_r/TD_r : vehicle arrival time at the P/D node of request r

TE_n : unload time-event n

$UR_{r,n,v}$: denotes that request r was unloaded from vehicle v before or exactly at time TE_n

$UT_{r,n}$: denotes that the request was unloaded on the cross-dock before or exactly at time event n

$YR_{r,v}$: states that the P/D locations of request r are both served by vehicle v

2.2. Problem formulation

Considering fuzzy parameters, mathematical model of problem was calculated as follows:

$$\text{Min } z_1 = \sum_{v \in V} [(OC_v^P + OC_v^D)] \quad (1)$$

$$\text{Min } z_2 = \sum_{v \in V} AT_v^D \quad (2)$$

$$\text{Min } z_3 = \mu \sum_{v \in V} AT_v^D + \sum_{v \in V} [(OC_v^P + OC_v^D)] \quad (3)$$

S.t:

$$\sum_{v \in V} GP_{w,v} = 1 \quad ; \forall w \in W \quad (4)$$

$$\sum_{v \in V} YP_{r,v} = 1 \quad ; \forall r \in R \quad (5)$$

$$CP_r \geq \widetilde{uc}_v \widetilde{d}_{w,r}^P YP_{r,v} GP_{w,v} \quad ; \forall r \in R . v \in V . w \in W \quad (6)$$

$$CP_{\acute{r}} \geq CP_r + \widetilde{u}\widetilde{c}_v \widetilde{d}_{r,\acute{r}}^P - M_C^P(1 - XP_{r,\acute{r}}) - M_C^P(2 - YP_{r,v} - YP_{\acute{r},v}) \quad (7)$$

$$; \forall r, \acute{r} \in R (r < \acute{r}), v \in V$$

(8)

$$CP_r \geq CP_{\acute{r}} + \widetilde{u}\widetilde{c}_v \widetilde{d}_{r,\acute{r}}^P - M_C^P XP_{r,\acute{r}} - M_C^P(2 - YP_{r,v} - YP_{\acute{r},v}); \forall r, \acute{r} \in R (\acute{r} < r), v \in V$$

$$OC_v^P \geq CP_r + \widetilde{u}\widetilde{c}_v \widetilde{d}_{r,w}^P - M_C^P(1 - YP_{r,v}) \quad ; \forall r \in R, v \in V, w \in W \quad (9)$$

$$TP_r \geq ST_v^P + \left(\frac{\widetilde{d}_{w,r}^P}{\widetilde{sp}}\right) YP_{r,v} GP_{w,v} \quad ; \forall r \in R, v \in V, w \in W \quad (10)$$

$$TP_{\acute{r}} \geq TP_r + \widetilde{f}t_r^P + \widetilde{l}r_r \widetilde{q}_r + \left(\frac{\widetilde{d}_{r,\acute{r}}^P}{\widetilde{sp}}\right) - M_T^P(1 - XP_{r,\acute{r}}) - M_T^P(2 - YP_{r,v} - YP_{\acute{r},v}) \quad (11)$$

$$; \forall r, \acute{r} \in R (r < \acute{r}), v \in V$$

$$TP_r \geq TP_{\acute{r}} + \widetilde{f}t_{\acute{r}}^P + \widetilde{l}r_{\acute{r}} \widetilde{q}_{\acute{r}} + \left(\frac{\widetilde{d}_{r,\acute{r}}^P}{\widetilde{sp}}\right) - M_T^P XP_{r,\acute{r}} - M_T^P(2 - YP_{r,v} - YP_{\acute{r},v}) \quad (12)$$

$$; \forall r, \acute{r} \in R (\acute{r} < r), v \in V$$

$$AT_v^P \geq TP_r + \widetilde{f}t_r^P + \widetilde{l}r_r \widetilde{q}_r + \left(\frac{\widetilde{d}_{r,w}^P}{\widetilde{sp}}\right) - M_T^P(1 - YP_{r,v}) \quad ; \forall r \in R, v \in V, w \in W \quad (13)$$

$$\sum_{r \in R} \widetilde{q}_r YP_{r,v} \leq \widetilde{Q}_v \quad ; \forall v \in V \quad (14)$$

$$\sum_{r \in R} \widetilde{q}_r YP_{r,v} GP_{w,v} \leq \widetilde{Q}_w \quad ; \forall v \in V, w \in W \quad (15)$$

$$YR_{r,v} \leq YP_{r,v} \quad ; \forall r \in R, v \in V \quad (16)$$

$$YR_{r,v} \leq YD_{r,v} \quad ; \forall r \in R, v \in V \quad (17)$$

$$YR_{r,v} \geq YP_{r,v} + YD_{r,v} - 1 \quad ; \forall r \in R, v \in V \quad (18)$$

$$\sum_{d \in RD} DP_{v,d} = 1 \quad ; \forall v \in V \quad (19)$$

$$RT_v^P \geq AT_v^P + \widetilde{f}t_w^P + \widetilde{u}\widetilde{r}_w GP_{w,v} [\sum_{r \in R} \widetilde{q}_r (YP_{r,v} - YR_{r,v})] \quad ; \forall r \in R, v \in V, w \in W \quad (20)$$

(21)

$$RT_{\acute{v}}^P \geq RT_v^P + \widetilde{f}t_w^P + \widetilde{u}\widetilde{r}_w GP_{w,\acute{v}} [\sum_{r \in R} \widetilde{q}_r (YP_{r,\acute{v}} - YR_{r,\acute{v}})] - M_T^P(1 - ZP_{v,\acute{v}}) - M_T^P(2 - DP_{v,d} - DP_{\acute{v},d}) \quad ; \forall d \in RD, v, \acute{v} \in V (v < \acute{v}), w \in W$$

(22)

$$RT_v^P \geq RT_{\acute{v}}^P + \widetilde{f}t_w^P + \widetilde{u}\widetilde{r}_w GP_{w,v} [\sum_{r \in R} \widetilde{q}_r (YP_{r,v} - YR_{r,v})] - M_T^P(1 - ZP_{v,\acute{v}}) - M_T^P(2 - DP_{v,d} - DP_{\acute{v},d}) \quad ; \forall d \in RD, v, \acute{v} \in V (v < \acute{v}), w \in W$$

$$\sum_{n \in N} WP_{n,v} = 1 \quad ; \forall v \in V \quad (23)$$

$$\sum_{v \in V} WP_{n,v} = 1 \quad ; \forall n \in N \quad (24)$$

$$TE_{\acute{n}} \geq TE_n \quad ; \forall n, \acute{n} \in N (n < \acute{n}) \quad (25)$$

$$TE_{\acute{n}} \geq RT_v^P + M_T^P(WP_{n,v} - 1) \quad ; \forall n, \acute{n} \in N (n < \acute{n}), v \in V \quad (26)$$

$$RT_v^P \leq TE_{\acute{n}} + M_T^P(1 - WP_{n,v}) \quad ; \forall n, \acute{n} \in N (n < \acute{n}), v \in V \quad (27)$$

$$\sum_{n \in N} TE_n = \sum_{v \in V} RT_v^P \quad (28)$$

$$TE_n \leq RT_v^P \quad ; \forall n = \text{first}(N), v \in V \quad (29)$$

$$TE_n \geq RT_v^P \quad ; \forall n = \text{last}(N), v \in V \quad (30)$$

$$UR_{r.n.v} \leq WP_{n.v} \quad ; \forall n \in N. r \in R. v \in V \quad (31)$$

$$\sum_{n \in N} UR_{r.n.v} \leq YP_{r.v} \quad ; \forall r \in R. v \in V \quad (32)$$

$$UR_{r.n.v} \geq (WP_{n.v} + YP_{r.v} - 1) \quad ; \forall n \in N. r \in R. v \in V \quad (33)$$

$$UT_{r.n} = \sum_{\substack{\hat{n} \in N \\ \hat{n} \leq n}} \sum_{v \in V} UR_{r.\hat{n}.v} \quad ; \forall n \in N. r \in R \quad (34)$$

$$ZP_{v.\acute{v}} \leq 2 - WP_{n.v} - \sum_{\substack{\hat{n} \in N \\ \hat{n} < n}} WP_{\hat{n}.\acute{v}} \quad ; \forall n \in N. v. \acute{v} \in V (v < \acute{v}) \quad (35)$$

$$ZP_{v.\acute{v}} \geq WP_{n.v} + \sum_{\substack{\hat{n} \in N \\ \hat{n} > n}} WP_{\hat{n}.\acute{v}} - 1 \quad ; \forall n \in N. v. \acute{v} \in V (v < \acute{v}) \quad (36)$$

$$\sum_{v \in V} GD_{w.v} = 1 \quad ; \forall w \in W \quad (37)$$

$$\sum_{v \in V} YD_{r.v} = 1 \quad ; \forall r \in R \quad (38)$$

$$\sum_{\acute{d} \in SD} DD_{v.\acute{d}} = 1 \quad ; \forall v \in V \quad (39)$$

$$DRS_{v.d.\acute{d}} \leq DP_{v.d} \quad ; \forall v \in V. d \in RD. \acute{d} \in SD \quad (40)$$

$$DRS_{v.d.\acute{d}} \leq DD_{v.\acute{d}} \quad ; \forall v \in V. d \in RD. \acute{d} \in SD \quad (41)$$

$$DRS_{v.d.\acute{d}} \geq DP_{v.d} + DD_{v.\acute{d}} - 1 \quad ; \forall v \in V. d \in RD. \acute{d} \in SD \quad (42)$$

$$\sum_{d \in RD} \sum_{\acute{d} \in SD} DRS_{v.d.\acute{d}} = 1 \quad ; \forall v \in V \quad (43)$$

$$\quad (44)$$

$$ST_v^D \geq RT_v^P + \sum_{d \in RD} \sum_{\acute{d} \in SD} \widetilde{tt}_{d.\acute{d}} DRS_{v.d.\acute{d}} + \widetilde{ft}_w^D + \widetilde{ur}_w GD_{w.v} [\sum_{r \in R} \widetilde{qr} (YD_{r.v} - YR_{r.v})] \quad ; \forall v \in V. w \in W \quad (45)$$

$$ST_{\acute{v}}^D \geq ST_v^D + \sum_{d \in RD} \sum_{\acute{d} \in SD} \widetilde{tt}_{d.\acute{d}} DRS_{\acute{v}.d.\acute{d}} + \widetilde{ft}_w^D + \widetilde{ur}_w GD_{w.\acute{v}} [\sum_{r \in R} \widetilde{qr} (YD_{r.\acute{v}} - YR_{r.\acute{v}})] - M_T^D (1 - ZD_{v.\acute{v}}) - M_T^D (2 - DD_{v.\acute{d}} - DD_{\acute{v}.\acute{d}}) \quad ; \forall v \in V. w \in W \quad (46)$$

$$ST_v^D \geq ST_{\acute{v}}^D + \sum_{d \in RD} \sum_{\acute{d} \in SD} \widetilde{tt}_{d.\acute{d}} DRS_{v.d.\acute{d}} + \widetilde{ft}_w^D + \widetilde{ur}_w GD_{w.v} [\sum_{r \in R} \widetilde{qr} (YD_{r.v} - YR_{r.v})] - M_T^D ZD_{v.\acute{v}} - M_T^D (2 - DD_{v.\acute{d}} - DD_{\acute{v}.\acute{d}}) \quad ; \forall v \in V. w \in W$$

$$\sum_{n \in N} WD_{n.v} = 1 \quad ; \forall v \in V \quad (47)$$

$$UT_{r.n} \geq (WD_{n.v} + YD_{r.v} - 1) \quad ; \forall n \in N. r \in R. v \in V \quad (48)$$

$$\quad (49)$$

$$ST_v^D \geq TE_n + \sum_{d \in RD} \sum_{\acute{d} \in SD} \widetilde{tt}_{d.\acute{d}} DRS_{v.d.\acute{d}} + \widetilde{ft}_w^D + \widetilde{ur}_w GD_{w.v} [\sum_{r \in R} \widetilde{qr} (YD_{r.v} - YR_{r.v})] - M_T^D (1 - WD_{n.v}) \quad ; \forall n \in N. v \in V. w \in W$$

$$WD_{n.v} \leq \sum_{\substack{\hat{n} \in N \\ \hat{n} \leq n}} WP_{\hat{n}.v} \quad ; \forall n \in N. v \in V \quad (50)$$

$$CD_r \geq \widetilde{uc}_v \widetilde{d}_{w.r}^D YD_{r.v} GD_{w.v} \quad ; \forall r \in R. v \in V. w \in W \quad (51)$$

$$CD_{\acute{r}} \geq CD_r + \widetilde{uc}_v \widetilde{d}_{r.\acute{r}}^D - M_C^D (1 - XD_{r.\acute{r}}) - M_C^D (2 - YD_{r.v} - YD_{\acute{r}.v}) \quad (52)$$

$$; \forall r. \acute{r} \in R (r < \acute{r}). v \in V$$

$$CD_r \geq CD_{\acute{r}} + \widetilde{uc}_v \widetilde{d}_{r.\acute{r}}^D - M_C^D XD_{r.\acute{r}} - M_C^D (2 - YD_{r.v} - YD_{\acute{r}.v}); \forall r. \acute{r} \in R (\acute{r} < r). v \in V \quad (53)$$

$$OC_v^D \geq CD_r + \widetilde{uc}_v \widetilde{d}_{r,w}^D - M_C^D (1 - YD_{r,v}) \quad ; \forall r \in R. v \in V. w \in W \quad (54)$$

$$TD_r \geq ST_v^D + \left(\frac{\widetilde{d}_{w,r}^D}{\widetilde{sp}} \right) YD_{r,v} GD_{w,v} \quad ; \forall r \in R. v \in V. w \in W \quad (55)$$

$$TD_{\acute{r}} \geq TD_r + \widetilde{ft}_r^D + \widetilde{ur}_r \widetilde{q}_r + \left(\frac{\widetilde{d}_{r,\acute{r}}^D}{\widetilde{sp}} \right) - M_T^D (1 - XP_{r,\acute{r}}) - M_T^{PD} (2 - YD_{r,v} - YD_{\acute{r},v}) \quad (56)$$

; $\forall r. \acute{r} \in R (r < \acute{r}). v \in V$

$$TD_r \geq TD_{\acute{r}} + \widetilde{ft}_{\acute{r}}^D + \widetilde{ur}_{\acute{r}} \widetilde{q}_{\acute{r}} + \left(\frac{\widetilde{d}_{\acute{r},r}^D}{\widetilde{sp}} \right) - M_T^D XP_{r,\acute{r}} - M_T^D (2 - YD_{r,v} - YD_{\acute{r},v}) \quad (57)$$

; $\forall r. \acute{r} \in R (\acute{r} < r). v \in V$

$$AT_v^D \geq TD_r + \widetilde{ft}_r^D + \widetilde{ur}_r \widetilde{q}_r + \left(\frac{\widetilde{d}_{r,w}^D}{\widetilde{sp}} \right) - M_T^D (1 - YD_{r,v}) \quad ; \forall r \in R. v \in V. w \in W \quad (58)$$

$$\sum_{r \in R} \widetilde{q}_r YD_{r,v} \leq \widetilde{Q}_v \quad ; \forall v \in V \quad (59)$$

$$\sum_{r \in R} \widetilde{q}_r YD_{r,v} GD_{w,v} \leq \widetilde{Q}_w \quad ; \forall v \in V. w \in W \quad (60)$$

$$ZD_{v,\acute{v}} \leq 2 - WD_{n,v} - \sum_{\substack{\acute{v} \in N \\ \acute{v} < n}} WD_{\acute{v},v} \quad ; \forall n \in N. v. \acute{v} \in V (v < \acute{v}) \quad (61)$$

$$ZD_{v,\acute{v}} \geq WD_{n,v} + \sum_{\substack{\acute{v} \in N \\ \acute{v} > n}} WD_{\acute{v},v} - 1 \quad ; \forall n \in N. v. \acute{v} \in V (v < \acute{v}) \quad (62)$$

$$AT_v^P \geq (1 - \eta_P) \left[ST_v^P + \left(\frac{OC_v^P}{\widetilde{uc}_v \widetilde{sp}} \right) + \sum_{r \in R} \left(\widetilde{ft}_r^P + \widetilde{lr}_r \widetilde{q}_r \right) YP_{r,v} \right] \quad ; \forall v \in V \quad (63)$$

$$AT_v^P \leq (1 + \eta_P) \left[ST_v^P + \left(\frac{OC_v^P}{\widetilde{uc}_v \widetilde{sp}} \right) + \sum_{r \in R} \left(\widetilde{ft}_r^P + \widetilde{lr}_r \widetilde{q}_r \right) YP_{r,v} \right] \quad ; \forall v \in V \quad (64)$$

$$AT_v^D \geq (1 - \eta_D) \left[ST_v^D + \left(\frac{OC_v^D}{\widetilde{uc}_v \widetilde{sp}} \right) + \sum_{r \in R} \left(\widetilde{ft}_r^D + \widetilde{lr}_r \widetilde{q}_r \right) YD_{r,v} \right] \quad ; \forall v \in V \quad (65)$$

$$AT_v^D \leq (1 + \eta_D) \left[ST_v^D + \left(\frac{OC_v^D}{\widetilde{uc}_v \widetilde{sp}} \right) + \sum_{r \in R} \left(\widetilde{ft}_r^D + \widetilde{lr}_r \widetilde{q}_r \right) YD_{r,v} \right] \quad ; \forall v \in V \quad (66)$$

$$\sum_{\substack{d \in RD \\ d < n}} DP_{v,d} \geq \sum_{\substack{\acute{v} \in N \\ \acute{v} < n}} WP_{\acute{v},v} \quad ; \forall v \in V. n \in N (n \leq |RD|) \quad (67)$$

Objective function z_1 tries to minimize cumulative routing cost of vehicle; and objective function z_2 aims at minimizing cumulative distribution time. Objective function z_3 seeks to minimize coordinated composition of the first and second objectives. It should be noted that coefficient μ indicates costs per unit time spent on accomplishment of delivery and loading tasks in the third objective function.

Eq.(4) refers to the allocation of loading vehicle to docks. Each vehicle should be allocated to a dock. If the allocation variable $GP_{w,v}$ is equal to 1, vehicle v will serve dock w .

Eq. (5) assigns requests to pickup vehicles. The pickup location of each request must be allocated to a single vehicle. The pickup node of request r will be served by the inbound vehicle v provided that the assignment variable $YP_{r,v}$ is equal to 1. Eq.(6) defines the routing cost from the cross-dock up to the first visited node on a pickup route and provides a lower bound on the routing cost from the cross-dock to any pickup node served by vehicle v , including the first visited location. The parameter \widetilde{uc}_v represents the routing cost per unit distance and $\widetilde{d}_{w,r}^D$, denotes the distance between the cross-docks, identified by the subscript w , and the pickup site of request r .

Both constraints (7) and (8) try to relate the cumulative routing costs from the cross-docks to the pickup sites of a pair of requests $(r. \acute{r} \in R)$ served by the same vehicle v (i.e. $YP_{r,v} = YP_{\acute{r},v} = 1$). A single binary variable $XP_{r,\acute{r}}$ (with $r < \acute{r}$) to select the relative order of any pair of pick-up nodes (r, \acute{r}) located on the same inbound route in this formulation. If $XP_{r,\acute{r}} = 1 (r < \acute{r})$, then the request r is served earlier than \acute{r} . By Eq. (7), therefore, $CP_{\acute{r}}$ must be larger than CP_r by at least the routing cost along the path directly connecting both locations, i.e. the shortest route between the pickup sites of r and \acute{r} .

Otherwise, $XP_{r,\hat{r}} = 0$ and node \hat{r} are seen before node r . Consequently, $CP_{\hat{r}}$ should be lower than CP_r by at least the cost term ($uc_v d_{r,\hat{r}}^P$) which is met by Eq.(8). It is worth mentioning that parameter M_C^P is a relatively large number.

Eq.(9) indicates overall routing cost for the tour allocated to pickup vehicle v . each pickup route should end at the cross-dock facility. Since there is unknown string of nodes on the route before solving the model, Eq. (9) provides a lower bound on the total routing cost for the vehicle tour OC_v^P considering any node on the route as the last visited one. The value of OC_v^P determined by the largest bound is set by the pickup location that is actually last visited by vehicle v . pickup node visiting times and vehicle arrival times at the cross-docks are presented in Eqs (10)-(13). These equations provide the opportunity to determine both the visiting time for the pickup location r (TP_r) and the v th-vehicle arrival time (AT_v^P) at the cross-dock. Vehicle v should wait its turn on the queue of the assigned strip dock door till the end of unloading operations. The timing constraints (11)-(12) present the same mathematical structures as Eqs (7)-(8). These sequencing constraints consider routing time parameters instead of routing cost coefficients. The service time at any pickup node r is the sum of two terms: a fixed preparation time ft_r^P plus the variable loading time that directly increases with the load size q_r . The proportionality constant lr_r stands for the loading rate at the pickup node r . Moreover, the routing time along the path connecting the pickup nodes r and \hat{r} is given by the ratio between the distance $d_{r,\hat{r}}^P$ and the vehicle speed sp_v . If all pickup routes are started at time $t = 0$, then $ST_v^P = 0$ for all $v \in V$, the continuous variable ST_v^P stands for the starting time of the v th-pickup route.

Eq. (14) doesn't allow the load transported by vehicle v to exceed its maximum capacity (Q_v). Eq.(15) doesn't allow the load transported by cross-dock w cannot exceed its maximum capacity (Q_w). Eqs (16)-(18) represent pickup node visiting times and vehicle arrival times at the cross-docks. When pickup and delivery sites of the request r are both served by the same vehicle, the related transshipment operations at the cross-dock are not required. In such a case, $YP_{r,v} = YD_{r,v} = 1$ for some vehicle v and the load of request r is not discharged on the receiving dock, i.e. it remains into the vehicle v . define $YR_{r,v}$ be a non-negative continuous variable with a domain $[0, 1]$ to identify requests fully served by vehicle v . Eqs (16)-(18) drives $YR_{r,v}$ to one while $YP_{r,v} = YD_{r,v} = 1$, and drops $YR_{r,v}$ to zero if either of such variables are null.

Eq. (19), causes a vehicle returning to the cross dock from its pick-up trip to perform the unloading operations in just one receiving dock door ($d \in RD$). Let us define the binary variable $DP_{v,d}$ to denote that the pickup vehicle v has been assigned to the strip dock door d whenever $DP_{v,d} = 1$. In Eq. (19), the set RD includes all the receiving doors available at the receiving dock.

Eq. (20) indicates sequencing pickup vehicles assigned to the same strip dock door. The trucks leave the cross-dock after all freight has been unloaded. Eq. (20) defines a lower bound for the release time (RT_v^P) at which the pickup vehicle v completes the off-load operations at the cross-dock and is ready to perform delivery tasks. We need to this bound to set the value of (RT_v^P) for the vehicle first served at any receiving dock door. In turn, constraints (21) and (22) relate the times at which vehicles (v, \hat{v}) $\in V$ ($v < \hat{v}$) end their unloading tasks just in case both vehicles have been assigned to the same strip door d ($DP_{v,d} = DP_{\hat{v},d} = 1$). The relative order of a pair of vehicles v and \hat{v} on the queue of the common assigned door d is defined by a single variable $ZP_{v,\hat{v}}$ (with $v < \hat{v}$). If $ZP_{v,\hat{v}} = 1$, provided that vehicle v is served before. Otherwise, $ZP_{v,\hat{v}} = 0$ and truck \hat{v} are unloaded earlier. After being serviced at different strip dock doors, the constraints (21) and (22) become redundant. The service time is the sum of two components at every door including a fixed preparation time (ft_w^P) and a variable service-time contribution which directly increases with the cargo to be unloaded given by $\sum_{r \in R} q_r (YP_{r,v} - YR_{r,v})$.

Eqs (23)-(24) are representatives for sequencing unloads events at the cross-docks. An unload event n occurs at the cross-dock whenever a pickup vehicle v just completes the discharge of the cargoes to be delivered by other vehicles. Therefore, there will be as many unloads events in the set N as the number of pickup vehicles on duty. N is an ordered event set with the element n occurring before event \hat{n} ($n < \hat{n}$). Let us define the binary variable $WP_{n,v}$ allocating pickup vehicles to unloads events, and the continuous variable TE_n representing the time at which the event n occurs. The event-time TE_n will be set by the release time of vehicle v from its pickup assignments (RT_v^P) only if $WP_{n,v} = 1$. Eqs (23)-(24) force an inbound vehicle to be exactly assigned to a single time event and an inbound vehicle to

be allocated to only one event. Dummy events are those assigned to unused vehicles that will never occur.

Furthermore, Eq. (25) proves the occurrence of event n prior to event $(n < \hat{n})$. Through Eq (25), the pickup vehicles should be assigned to unloads events in the same order that they complete their pickup duties. If the event n has been allocated to vehicle v ($WP_{n,v} = 1$), then $TE_n = RT_v^P$. Eq (26) sets the value of RT_v^P as a lower bound for TE_n whenever vehicle v has been assigned to either an earlier event ($\hat{n} < n$) or to event n itself. The equality condition is met by Eqs (27)-(30).

The subset of requests already unloaded at the cross-dock at the event time TE_n is involved in Eqs (31)-(33). Let $UR_{r,n,v}$ be a continuous variable with domain (0,1) denoting that request r collected by vehicle v is available for delivery on the cross-dock at the event time TE_n only if $UR_{r,n,v} = 1$. When the request r is not collected by vehicle v ($YP_{r,v} = 0$) or is assigned to an event $n \neq \hat{n}$ ($WP_{n,v} = 0$), Eqs (31) and (32) drive $UR_{r,n,v}$ to zero. If the reverse situation holds, $UR_{r,n,v}$ is set equal to one by Eq (33).

Continuous variable $UT_{r,n}$ with domain (0,1) provides the subset of requests already unloaded on the receiving dock at time TE_n . If $UT_{r,n} = 1$, then the request r has been discharged from the pickup vehicle at a time earlier than or equal to TE_n . In case the request r still remains on the cross dock at TE_n , it will be available for delivery at that time. The value of $UT_{r,n}$ is defined by Eq (34).

There are normally some loads temporarily stored in front of the stack doors waiting for the arrival of the other goods to be also delivered by the assigned outbound truck.

Eqs (35)-(36) address the further queuing constraints for vehicles assigned to the same receiving door. When the inbound vehicles v and \hat{v} (with $v < \hat{v}$) have been allocated to the same receiving door $d \in RD$ and vehicle v features an earlier unload event ($WP_{n,v} = WP_{\hat{n},\hat{v}} = 1$ with $n < \hat{n}$), then by Eqs. (35) and (36) vehicle v must be served before \hat{v} and $ZP_{v,\hat{v}} = 1$. Otherwise, vehicle \hat{v} is unloaded before and $ZP_{v,\hat{v}} = 0$ by Eq. (35). When vehicles v and \hat{v} fail to share the same strip dock door, Eqs (35)-(36) become redundant.

Eq. (37) allocates unloading vehicle to docks. Each vehicle should be allocated to a dock. If the allocation variable $GD_{w,v}$ is equal to 1, vehicle v serves dock w . As stated by Eq. (38), each transportation request must be allocated to a single outbound vehicle. A binary variable $YD_{r,v}$ should be defined to denote the allocation of request r to the outbound vehicle v only if $YD_{r,v} = 1$.

Eq. (38) allocates delivery vehicles to shipping dock doors. We set $DD_{v,\hat{d}}$ as a binary variable allocating outbound vehicles to shipping doors. If $DD_{v,\hat{d}} = 1$, then the loading operations for vehicle v will take place at the shipping door $\hat{d} \in SD$. As stated by Eq. (39), an outbound vehicle on duty must be loaded at just one stack dock door. The set SD comprises the shipping doors available at the cross-dock.

Eqs (40)-(43) aim at identifying the strip and stack dock doors assigned to each vehicle. The continuous variable $DRS_{v,d,\hat{d}}$ with domain (0, 1) has been introduced to indicate that vehicle v should move from the strip door $d \in RD$ to the stack door $\hat{d} \in SD$ before starting the loading operations. Eqs (40)-(43) drive the variable $DRS_{v,d,\hat{d}}$ to one whenever $DP_{v,d} = DD_{v,\hat{d}} = 1$, and drops $DRS_{v,d,\hat{d}}$ to zero if either of such variables are null.

Eqs (44)-(46) clarify the sequence of outbound vehicles assigned to the same shipping door. The continuous variable ST_v^D denotes the time at which the delivery vehicle v starts the loading of the assigned requests at the cross-dock. Considering the same fleet of vehicles for pickup and delivery tasks, a pair of constraints are essential to be defined on the value of ST_v^D :

- (a) Pickup assignments need to be completed to let the loading of a delivery vehicle v start, i.e. it shouldn't be earlier than RT_v^P ; and (b) all the preceding trucks on the queue of the assigned stack dock door $\hat{d} \in SD$ (i.e. $DD_{v,\hat{d}} = 1$) should be served to let the loading of vehicle v begin. Eq. (44) accounts for constraint (a) while Eqs (45)-(46) mathematically describe the condition (b) by relating the times ST_v^D and $ST_{\hat{v}}^D$ at which the pair of vehicles $(v, \hat{v}) \in V$ (with $v < \hat{v}$) assigned to the same shipping door \hat{d} ($DD_{v,\hat{d}} = DD_{\hat{v},\hat{d}} = 1$) finish their loading activities at the cross-dock. If vehicle v precedes \hat{v} on the queue of door \hat{d} , then the sequencing variable $ZD_{v,\hat{v}}$ will be equal to one as explained in Eq. (45) applies. Otherwise, $ZD_{v,\hat{v}} = 0$ and Eq. (46) becomes the relevant constraint. When two vehicles are allocated to different stack dock doors, constraints (45)-(46) both become redundant. The total loading

time is equal to the sum of a fixed preparation time ft_w^D plus a variable time contribution that directly increases with the load size regarding Eqs (44) and (45)-(46). Furthermore, $tt_{d,\hat{d}}$ states the time spent by a vehicle to move from the receiving door $d \in RD$ to the shipping door $\hat{d} \in SD$. Constraint (44) should be omitted when the fleets of inbound and outbound vehicles are different. If the vehicles are either inbound or outbound trucks, the model will still be applied.

- (b) Since there is small travel time between the docks in comparison with the time during which the freights should temporarily remain on the cross-dock, the constraint (44) will keep redundant.

Eqs (47)-(50) assign delivery vehicles to unloads events. All of the requests should be delivered by an available truck at the cross-dock to let an outbound vehicle start its loading. The main reason is that the loading sequence is generally determined by: (a) the need of having the loads tightly packed into the truck and putting the fragile goods on the top, and (b) the ordering of the delivery nodes on the vehicle route [1]. The binary variable $WD_{n,v}$ is defined to denote that the outbound vehicle v has been assigned to the unload event $n \in ND$ only if $WD_{n,v} = 1$. Allocating the outbound vehicle v to event n ($WD_{n,v} = 1$) states that the requests assigned to vehicle v ($YD_{r,v} = 1$) have already been unloaded on the cross-dock at a time earlier than or equal to TE_n . Such requests all feature $UT_{r,n} = 1$ and, therefore, the condition $WD_{n,v} + YD_{r,v} = 2$ implies that $UT_{r,n} = 1$ and the loading of vehicle v is forced to begin after TE_n .

Eq. (47) insures the assignment of each outbound vehicle on duty to a single unload event $n \in N$. Several delivery vehicles can be allocated to the same unload event, though. Eq. (48) will be able to set $WD_{n,v} = YD_{r,v} = 1$ only if the variable $UT_{r,n}$ is equal to one. In this way, Eq. (48) avoids the allocation of event n to an outbound vehicle v if $UT_{r,n} = 0$ for some requests r with $YD_{r,v} = 1$.

Furthermore, Eq. (49) doesn't let an outbound vehicle v allocated to event n to start the loading operations prior to time TE_n . In addition, Eq. (49) will drive the variable $WD_{n,v}$ to zero if the unload event for vehicle v occurs at some later event ($\hat{n} > n$), i.e. $WP_{\hat{n},v} = 0$ for some ($\hat{n} \leq n$). If every truck is either inbound or outbound, we should omit Eq. (50).

We apply similar constraint sets with mathematical structures to the proposed ones for the pickup phase for delivery routes. Replacing the assignment variable $YP_{r,v}$ by $YD_{r,v}$, the routing cost CP_r by CD_r , the visiting time TP_r by TD_r , the sequencing variable $XP_{r,\hat{r}}$ by $XD_{r,\hat{r}}$ ($r < \hat{r}$), and the superscript P by D, formulations can be derived from Eqs (6)-(15).

Eqs (51)-(54) define sequencing constraints providing the outbound routing costs from the cross-docks up to the delivery site of request r are defined by Eqs (51)-(54). The parameter M_C^D is a relatively large number.

Eqs (55)-(58) elaborate on the set of constraints providing lower bounds for the vehicle stop times at delivery locations.

Eq. (59) forces the load transported by vehicle v not to exceed its maximum capacity (Q_v). Eqs (60) states that the load transported by cross-dock w cannot exceed its maximum capacity (Q_w).

Eqs (61)-(62) demonstrate the requirement of further queuing constraints for vehicles sharing the same shipping door. If delivery vehicles v and \hat{v} are loaded at the same stack dock door and vehicle v is allocated to an earlier event, then vehicle v will be served before and $ZD_{v,\hat{v}} = 1$ regarding Eqs. (61)-(62). In the contrary, vehicle \hat{v} is loaded earlier and $ZD_{v,\hat{v}} = 0$. The value of $ZD_{v,\hat{v}}$ can also be meaningless when the vehicles have been allocated to different shipping doors.

Related constraints to the total routing cost and the vehicle arrival times are considered as additional constraints to speed up the solution process.

If there is a relationship between arrival time AT_v^P and the total routing cost for the pickup tour of vehicle v , lower and upper bounds on the value of AT_v^P will be obtained through Eqs (63) and (64), respectively. Estimating AT_v^P as the sum of the starting time ST_v^P plus the total service time at the visited locations and the total traveling time, such bounds will be obtained. Regardless of the time windows for the service start at the P/D locations, the parameter η will be equal to zero. Nonetheless, it has been chosen $\eta_p = 0 \cdot 001$ to account for round off errors. Since the pickup vehicles sometimes are required to wait for the opening of the time window at some visiting sites, the value of η_p should be increased to 0.1-0.3 for problems with narrow time windows.

Constraints (65)-(66) that are similar to Eqs (63)-(64) are dedicated to the delivery phase.

Eq. (67) explains the valid inequality constraints for allocating received dock doors to the vehicles. Constraint (67) is incorporated into the mathematical model to solve large problems and eliminate symmetric solutions. If the set RD comprises three elements $\{rd_1, rd_2, rd_3\}$, then constraints (67) allocates the dock door rd_1 to the vehicle v^* that first unloads the cargo on the cross-dock terminal (e.g., $WP_{n1,v^*} = 1$), the dock door rd_2 to the vehicle $v^\#$ completing the unloading operations in the second place (e.g., $WP_{n2,v^\#} = 1$) and rd_3 to the truck finishing the pickup duties on third place. The optimal solution is not excluded from the feasible region by Constraint (67) but avoids symmetrical assignments.

3. Solution methodology

3.1. Fuzzy-interactive approach

Defuzzification is an important step in fuzzy systems. In fuzzy systems, the results of an approximate argument are usually obtained in the form of one or more fuzzy sets. In these cases, the fuzzy output of the system needs to be converted to a normal (non-fuzzy) number. There are several methods in this field, which in this article uses a two-step approach.

Mathematical programming techniques and equations vividly fail to solve transportation decision-making problems by fuzzy objective function coefficients. Accordingly, a fuzzy-interactive multi-objective linear programming model is presented to overcome this deficiency while facing a transportation decision problems with fuzzy objective function coefficients. The proposed model has proved to be far better than the previous ones in terms of computational flexibility and efficiency.

As it is obvious from the presented model, the majority of parameters are fuzzy and include correct values and technological coefficients. Moreover, objective functions and main deterministic variables are set as the constraints. We use a two-phase approach in order to solve the proposed fuzzy model. In the first phase, the initial fuzzy model is altered to a deterministic equivalent auxiliary model. In the second phase, a fuzzy method is applied to obtain the final preferred compromise solution.

3.1.1. Deterministic equivalent model

We apply Jimenez et al.'s (2007) method to convert possible model to a deterministic equivalent model which contains the inaccurate coefficients in constraints. Fixing the number of objective functions and unequal constraint, this method proved to be highly efficient when it comes to face uncertainty. Aiming at modeling the vague nature of inaccurate parameters to achieve computational efficiency and simplicity in data acquisition, a fuzzy trigonometric distribution is applied. Considering $\tilde{c} = (c^p, c^m, c^o)$ as a triangular fuzzy number. The reason for this choice is that these numbers are often used in applications of fuzzy controllers, managerial decision making, business and finance, etc., and graphical representations and operations with triangular numbers are very easy and they also have a linear membership function and all parameters are assumed to be triangular fuzzy numbers.

The membership function ($\mu_{\tilde{c}}(x)$) will be as follows:

$$\mu_{\tilde{c}}(x) = \begin{cases} f_c(x) = \frac{x-c^p}{c^m-c^p} & \text{if } c^p \leq x \leq c^m \\ 1 & \text{if } x = c^m \\ g_c(x) = \frac{c^o-x}{c^o-c^m} & \text{if } c^m \leq x \leq c^o \\ 0 & \text{if } x < c^p \text{ or } x > c^o \end{cases} \quad (68)$$

The following equation represents the expected interval (EI) and expected value (EV) of fuzzy number \tilde{c} [32]:

$$EI(\tilde{c}) = [E_1^c, E_2^c] = \left[\int_0^1 f_c^{-1}(x) dx, \int_0^1 g_c^{-1}(x) dx \right] \quad (69)$$

$$EV(\tilde{c}) = \frac{E_1^c + E_2^c}{2} \quad (70)$$

Display parameters are obtained using a triangular fuzzy distribution as follows:

$$EI(\tilde{c}) = \left[\frac{1}{2}(c^p + c^m), \frac{1}{2}(c^m + c^o) \right] \quad (71)$$

$$EV(\tilde{c}) = \frac{c^p + 2c^m + c^o}{4} \quad (72)$$

A fuzzy mathematical programming model with fuzzy parameters is considered:

$$\begin{aligned} \text{Min } Z &= \tilde{c}x \\ \text{s.t. } \tilde{a}_i x &\geq \tilde{b}_i \quad i = 1, \dots, l \\ \tilde{a}_i x &= \tilde{b}_i \quad i = l + 1, \dots, m \\ x &\geq 0 \end{aligned} \tag{73}$$

Feasibility and optimality are two major issues to compare fuzzy numbers and address the non-deterministic and uncertain nature of problem parameters. Furthermore, it is essential to answer the following two questions [33]:

- 1) Regarding the existence of fuzzy numbers in constraints, how we can define the feasibility of decision vector x .
- 2) How we can measure the optimality of objective function can be defined by fuzzy coefficients.

Inspired by Jimenez et al.'s ranking method [33], a degree is defined for each pair of fuzzy numbers \tilde{a} and \tilde{b} in which \tilde{a} is larger than \tilde{b} as follows: (74)

$$\mu_M(\tilde{a}, \tilde{b}) = \begin{cases} 0 & \text{if } E_2^a - E_1^a > 0 \\ \frac{E_2^a - E_1^b}{E_2^a - E_1^b - (E_1^a - E_2^b)} & \text{if } 0 \in [E_1^a - E_2^b, E_2^a - E_1^b] \\ 1 & \text{if } E_1^a - E_2^b > 0 \end{cases}$$

If $\mu_M(\tilde{a}, \tilde{b}) \geq \alpha$, then \tilde{a} is larger than or equal to \tilde{b} at least in degree α . Regarding [19], decision vector $x \in \mathcal{R}$ is feasible in degrees α if $\min_{i=1, \dots, m} \{\mu_M(\tilde{a}_i x, \tilde{b}_i)\} = \alpha$. Accordingly, problem constraints are defined as follows:

$$\frac{E_2^{a_i x} - E_1^{b_i}}{E_2^{a_i x} - E_1^{a_i x} + E_1^{b_i} - E_2^{b_i}} \geq \alpha \quad i = 1, \dots, l \tag{75}$$

Equation (75) will be changed to the following equation after simplification:

$$[(1 - \alpha)E_2^{a_i} + \alpha E_1^{a_i}]x \geq \alpha E_2^{b_i} + (1 - \alpha)E_1^{b_i} \tag{76}$$

For equal state, we also have:

$$\tilde{a} \geq_{\frac{\alpha}{2}} \tilde{b} \quad . \quad \tilde{a} \leq_{\frac{\alpha}{2}} \tilde{b} \tag{77}$$

The above-mentioned equation can be rewritten as follows:

$$\frac{\alpha}{2} \leq \mu_M(\tilde{a}, \tilde{b}) \leq 1 - \frac{\alpha}{2} \tag{78}$$

Feasibility solution of x^0 will be an acceptable optimal solution for model (73) provided that the following condition is true:

$$\mu_M(\tilde{c}x, \tilde{c}x^0) \geq \frac{1}{2} \tag{79}$$

We can conclude that x^0 provides better solution than feasibility vectors (with minimization goal) at least in $\frac{1}{2}$ degree; we also have:

$$\tilde{c}x \geq_{\frac{1}{2}} \tilde{c}x^0 \tag{80}$$

Regarding previous equations, we will have:

$$\frac{E_2^{c x} - E_1^{c x^0}}{E_2^{c x} - E_1^{c x} + E_1^{c x^0} - E_2^{c x^0}} \geq \frac{1}{2} \tag{81}$$

Or

$$\frac{E_2^{cx} + E_1^{cx}}{2} \geq \frac{E_2^{cx^0} + E_1^{cx^0}}{2} \quad (82)$$

We place the equations (76), (78) and (82) in model (73) to obtain its α -parameter model as follows:

$$\text{Min } Z = EV(\tilde{c})x \quad (83)$$

s.t.

$$[(1 - \alpha)E_2^{a_i} + \alpha E_1^{a_i}]x \geq \alpha E_2^{b_i} + (1 - \alpha)E_1^{b_i} \quad i = 1 \dots m$$

$$\left[\left(1 - \frac{\alpha}{2}\right)E_2^{a_i} + \frac{\alpha}{2}E_1^{a_i} \right]x \geq \frac{\alpha}{2}E_2^{b_i} + \left(1 - \frac{\alpha}{2}\right)E_1^{b_i} \quad i = l + 1 \dots m$$

$$\left[\frac{\alpha}{2}E_2^{a_i} + \left(1 - \frac{\alpha}{2}\right)E_1^{a_i} \right]x \geq \left(1 - \frac{\alpha}{2}\right)E_2^{b_i} + \frac{\alpha}{2}E_1^{b_i} \quad i = l + 1 \dots m$$

$$x \geq 0$$

We can formulate an auxiliary deterministic equivalent model of the original problem using the method above:

$$\text{Min } z_1 = \sum_{v \in V} [(OC_v^P + OC_v^D)] \quad (84)$$

$$\text{Min } z_2 = \sum_{v \in V} AT_v^D \quad (85)$$

$$\text{Min } z_3 = \sum_{v \in V} AT_v^D + \sum_{v \in V} [(OC_v^P + OC_v^D)] \quad (86)$$

s.t.

$$\sum_{v \in V} GP_{w,v} = 1 \quad ; \forall w \in W \quad (87)$$

$$\sum_{v \in V} YP_{r,v} = 1 \quad ; \forall r \in R \quad (88)$$

$$(89)$$

$$CP_r \geq \left[\alpha \left(\frac{uc_v^m + uc_v^o}{2} \right) + (1 - \alpha) \left(\frac{uc_v^p + uc_v^m}{2} \right) \right] \left[\alpha \left(\frac{(d_{w,r}^p)^m + (d_{w,r}^p)^o}{2} \right) + (1 - \alpha) \left(\frac{(d_{w,r}^p)^p + (d_{w,r}^p)^m}{2} \right) \right] YP_{r,v} GP_{w,v} \quad ; \forall r \in R . v \in V . w \in W \quad (90)$$

$$CP_{\hat{r}} \geq CP_r + \left[\alpha \left(\frac{uc_v^m + uc_v^o}{2} \right) + (1 - \alpha) \left(\frac{uc_v^p + uc_v^m}{2} \right) \right] \left[\alpha \left(\frac{(d_{r,\hat{r}}^p)^m + (d_{r,\hat{r}}^p)^o}{2} \right) + (1 - \alpha) \left(\frac{(d_{r,\hat{r}}^p)^p + (d_{r,\hat{r}}^p)^m}{2} \right) \right] - M_C^P (1 - XP_{r,\hat{r}}) - M_C^P (2 - YP_{r,v} - YP_{\hat{r},v}) \quad ; \forall r, \hat{r} \in R (r < \hat{r}) . v \in V \quad (91)$$

$$CP_r \geq CP_{\hat{r}} + \left[\alpha \left(\frac{uc_v^m + uc_v^o}{2} \right) + (1 - \alpha) \left(\frac{uc_v^p + uc_v^m}{2} \right) \right] \left[\alpha \left(\frac{(d_{r,\hat{r}}^p)^m + (d_{r,\hat{r}}^p)^o}{2} \right) + (1 - \alpha) \left(\frac{(d_{r,\hat{r}}^p)^p + (d_{r,\hat{r}}^p)^m}{2} \right) \right] - M_C^P XP_{r,\hat{r}} - M_C^P (2 - YP_{r,v} - YP_{\hat{r},v}) \quad ; \forall r, \hat{r} \in R (\hat{r} < r) . v \in V \quad (92)$$

$$OC_v^P \geq CP_r + \left[\alpha \left(\frac{uc_v^m + uc_v^o}{2} \right) + (1 - \alpha) \left(\frac{uc_v^p + uc_v^m}{2} \right) \right] \left[\alpha \left(\frac{(d_{w,r}^p)^m + (d_{w,r}^p)^o}{2} \right) + (1 - \alpha) \left(\frac{(d_{w,r}^p)^p + (d_{w,r}^p)^m}{2} \right) \right] - M_C^P (1 - YP_{r,v}) \quad ; \forall r \in R . v \in V . w \in W$$

$$TP_r \geq ST_v^P + \left(\frac{\left[\alpha \left(\frac{(d_{w,r}^P)^m + (d_{w,r}^P)^o}{2} \right) + (1-\alpha) \left(\frac{(d_{w,r}^P)^p + (d_{w,r}^P)^m}{2} \right) \right]}{\left[\alpha \left(\frac{sp_v^m + sp_v^o}{2} \right) + (1-\alpha) \left(\frac{sp_v^p + sp_v^m}{2} \right) \right]} \right) YP_{r,v} GP_{w,v} ; \forall r \in R. v \in V. w \in W \quad (95)$$

$$(94)$$

$$TP_{\hat{r}} \geq TP_r + \left[\alpha \left(\frac{(ft_r^P)^m + (ft_r^P)^o}{2} \right) + (1-\alpha) \left(\frac{(ft_r^P)^p + (ft_r^P)^m}{2} \right) \right] + \left[\alpha \left(\frac{lr_r^m + lr_r^o}{2} \right) + (1-\alpha) \left(\frac{lr_r^p + lr_r^m}{2} \right) \right] \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1-\alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] + \left(\frac{\left[\alpha \left(\frac{(d_{r,\hat{r}}^P)^m + (d_{r,\hat{r}}^P)^o}{2} \right) + (1-\alpha) \left(\frac{(d_{r,\hat{r}}^P)^p + (d_{r,\hat{r}}^P)^m}{2} \right) \right]}{\left[\alpha \left(\frac{sp_v^m + sp_v^o}{2} \right) + (1-\alpha) \left(\frac{sp_v^p + sp_v^m}{2} \right) \right]} \right) - M_T^P (1 - XP_{r,\hat{r}}) - M_T^P (2 - YP_{r,v} - YP_{\hat{r},v}) ; \forall r, \hat{r} \in R (r < \hat{r}). v \in V \quad (95)$$

$$TP_r \geq TP_{\hat{r}} + \left[\alpha \left(\frac{(ft_{\hat{r}}^P)^m + (ft_{\hat{r}}^P)^o}{2} \right) + (1-\alpha) \left(\frac{(ft_{\hat{r}}^P)^p + (ft_{\hat{r}}^P)^m}{2} \right) \right] + \left[\alpha \left(\frac{lr_{\hat{r}}^m + lr_{\hat{r}}^o}{2} \right) + (1-\alpha) \left(\frac{lr_{\hat{r}}^p + lr_{\hat{r}}^m}{2} \right) \right] \left[\alpha \left(\frac{q_{\hat{r}}^m + q_{\hat{r}}^o}{2} \right) + (1-\alpha) \left(\frac{q_{\hat{r}}^p + q_{\hat{r}}^m}{2} \right) \right] + \left(\frac{\left[\alpha \left(\frac{(d_{\hat{r},r}^P)^m + (d_{\hat{r},r}^P)^o}{2} \right) + (1-\alpha) \left(\frac{(d_{\hat{r},r}^P)^p + (d_{\hat{r},r}^P)^m}{2} \right) \right]}{\left[\alpha \left(\frac{sp_v^m + sp_v^o}{2} \right) + (1-\alpha) \left(\frac{sp_v^p + sp_v^m}{2} \right) \right]} \right) - M_T^P XP_{r,\hat{r}} - M_T^P (2 - YP_{r,v} - YP_{\hat{r},v}) ; \forall r, \hat{r} \in R (\hat{r} < r). v \in V \quad (96)$$

$$AT_v^P \geq TP_r + \left[\alpha \left(\frac{(ft_r^P)^m + (ft_r^P)^o}{2} \right) + (1-\alpha) \left(\frac{(ft_r^P)^p + (ft_r^P)^m}{2} \right) \right] + \left[\alpha \left(\frac{lr_r^m + lr_r^o}{2} \right) + (1-\alpha) \left(\frac{lr_r^p + lr_r^m}{2} \right) \right] \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1-\alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] + \left(\frac{\left[\alpha \left(\frac{(d_{w,r}^P)^m + (d_{w,r}^P)^o}{2} \right) + (1-\alpha) \left(\frac{(d_{w,r}^P)^p + (d_{w,r}^P)^m}{2} \right) \right]}{\left[\alpha \left(\frac{sp_v^m + sp_v^o}{2} \right) + (1-\alpha) \left(\frac{sp_v^p + sp_v^m}{2} \right) \right]} \right) - M_T^P (1 - YP_{r,v}) ; \forall r \in R. v \in V. w \in W$$

$$\sum_{r \in R} \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1-\alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] YP_{r,v} \leq \left[\alpha \left(\frac{Q_v^m + Q_v^o}{2} \right) + (1-\alpha) \left(\frac{Q_v^p + Q_v^m}{2} \right) \right] ; \forall v \in V \quad (97)$$

$$\sum_{r \in R} \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1-\alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] YP_{r,v} GP_{w,v} \leq \left[\alpha \left(\frac{Q_w^m + Q_w^o}{2} \right) + (1-\alpha) \left(\frac{Q_w^p + Q_w^m}{2} \right) \right] \quad (98)$$

; $\forall v \in V. w \in W$

$$YR_{r,v} \leq YP_{r,v} ; \forall r \in R. v \in V \quad (99)$$

$$YR_{r,v} \leq YD_{r,v} ; \forall r \in R. v \in V \quad (100)$$

$$YR_{r,v} \geq YP_{r,v} + YD_{r,v} - 1 ; \forall r \in R. v \in V \quad (101)$$

$$\sum_{d \in RD} DP_{v,d} = 1 ; \forall v \in V \quad (102)$$

$$(103)$$

$$RT_v^P \geq AT_v^P + \left[\alpha \left(\frac{(ft_w^P)^m + (ft_w^P)^o}{2} \right) + (1-\alpha) \left(\frac{(ft_w^P)^p + (ft_w^P)^m}{2} \right) \right] + \left[\alpha \left(\frac{ur_w^m + ur_w^o}{2} \right) + (1-\alpha) \left(\frac{ur_w^p + ur_w^m}{2} \right) \right] GP_{w,v} \left[\sum_{r \in R} \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1-\alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] (YP_{r,v} - YR_{r,v}) \right] ; \forall r \in R. v \in V. w \in W$$

(104)

$$RT_v^P \geq RT_v^P + \left[\alpha \left(\frac{(ft_w^P)^m + (ft_w^P)^o}{2} \right) + (1 - \alpha) \left(\frac{(ft_w^P)^p + (ft_w^P)^m}{2} \right) \right] + \left[\alpha \left(\frac{ur_w^m + ur_w^o}{2} \right) + (1 - \alpha) \left(\frac{ur_w^p + ur_w^m}{2} \right) \right] GP_{w.v} \left[\sum_{r \in R} \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1 - \alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] (YP_{r.v} - YR_{r.v}) \right] - M_T^P (1 - ZP_{v.v}) - M_T^P (2 - DP_{v.d} - DP_{v.d}) \quad ; \forall d \in RD. v, v' \in V (v < v'). w \in W$$

(105)

$$RT_v^P \geq RT_v^P + \left[\alpha \left(\frac{(ft_w^P)^m + (ft_w^P)^o}{2} \right) + (1 - \alpha) \left(\frac{(ft_w^P)^p + (ft_w^P)^m}{2} \right) \right] + \left[\alpha \left(\frac{ur_w^m + ur_w^o}{2} \right) + (1 - \alpha) \left(\frac{ur_w^p + ur_w^m}{2} \right) \right] GP_{w.v} \left[\sum_{r \in R} \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1 - \alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] (YP_{r.v} - YR_{r.v}) \right] - M_T^P (1 - ZP_{v.v}) - M_T^P (2 - DP_{v.d} - DP_{v.d}) \quad ; \forall d \in RD. v, v' \in V (v' < v). w \in W$$

$$\sum_{n \in N} WP_{n.v} = 1 \quad ; \forall v \in V \quad (106)$$

$$\sum_{v \in V} WP_{n.v} = 1 \quad ; \forall n \in N \quad (107)$$

$$TE_{\hat{n}} \geq TE_n \quad ; \forall n, \hat{n} \in N (n < \hat{n}) \quad (108)$$

$$TE_{\hat{n}} \geq RT_v^P + M_T^P (WP_{n.v} - 1) \quad ; \forall n, \hat{n} \in N (n < \hat{n}). v \in V \quad (109)$$

$$RT_v^P \leq TE_{\hat{n}} + M_T^P (1 - WP_{n.v}) \quad ; \forall n, \hat{n} \in N (n < \hat{n}). v \in V \quad (110)$$

$$\sum_{n \in N} TE_n = \sum_{v \in V} RT_v^P \quad (111)$$

$$TE_n \leq RT_v^P \quad ; \forall n = \text{first}(N). v \in V \quad (112)$$

$$TE_n \geq RT_v^P \quad ; \forall n = \text{last}(N). v \in V \quad (113)$$

$$UR_{r.n.v} \leq WP_{n.v} \quad ; \forall n \in N. r \in R. v \in V \quad (114)$$

$$\sum_{n \in N} UR_{r.n.v} \leq YP_{r.v} \quad ; \forall r \in R. v \in V \quad (115)$$

$$UR_{r.n.v} \geq (WP_{n.v} + YP_{r.v} - 1) \quad ; \forall n \in N. r \in R. v \in V \quad (116)$$

$$UT_{r.n} = \sum_{\substack{\hat{n} \in N \\ \hat{n} \leq n}} \sum_{v \in V} UR_{r.\hat{n}.v} \quad ; \forall n \in N. r \in R \quad (117)$$

$$ZP_{v.v'} \leq 2 - WP_{n.v} - \sum_{\substack{\hat{n} \in N \\ \hat{n} < n}} WP_{\hat{n}.v'} \quad ; \forall n \in N. v, v' \in V (v < v') \quad (118)$$

$$ZP_{v.v'} \geq WP_{n.v} + \sum_{\substack{\hat{n} \in N \\ \hat{n} > n}} WP_{\hat{n}.v'} - 1 \quad ; \forall n \in N. v, v' \in V (v < v') \quad (119)$$

$$\sum_{v \in V} GD_{w.v} = 1 \quad ; \forall w \in W \quad (120)$$

$$\sum_{v \in V} YD_{r.v} = 1 \quad ; \forall r \in R \quad (121)$$

$$\sum_{\hat{d} \in SD} DD_{v.\hat{d}} = 1 \quad ; \forall v \in V \quad (122)$$

$$DRS_{v.d.\hat{d}} \leq DP_{v.d} \quad ; \forall v \in V. d \in RD. \hat{d} \in SD \quad (123)$$

$$DRS_{v.d.\hat{d}} \leq DD_{v.\hat{d}} \quad ; \forall v \in V. d \in RD. \hat{d} \in SD \quad (124)$$

$$DRS_{v.d.\hat{d}} \geq DP_{v.d} + DD_{v.\hat{d}} - 1 \quad ; \forall v \in V. d \in RD. \hat{d} \in SD \quad (125)$$

$$\sum_{d \in RD} \sum_{\hat{d} \in SD} DRS_{v.d.\hat{d}} = 1 \quad ; \forall v \in V \quad (126)$$

(127)

$$\begin{aligned}
ST_v^D &\geq RT_v^P + \sum_{d \in RD} \sum_{\acute{d} \in SD} \left[\alpha \left(\frac{tt_{\acute{d}\acute{d}}^m + tt_{\acute{d}\acute{d}}^o}{2} \right) + (1 - \alpha) \left(\frac{tt_{\acute{d}\acute{d}}^p + tt_{\acute{d}\acute{d}}^m}{2} \right) \right] DRS_{v.d.\acute{d}} + \left[\alpha \left(\frac{(ft_w^D)^m + (ft_w^D)^o}{2} \right) + \right. \\
&(1 - \alpha) \left. \left(\frac{(ft_w^D)^p + (ft_w^D)^m}{2} \right) \right] + \left[\alpha \left(\frac{ur_w^m + ur_w^o}{2} \right) + (1 - \alpha) \left(\frac{ur_w^p + ur_w^m}{2} \right) \right] GD_{w.v} \left[\sum_{r \in R} \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1 - \right. \right. \\
&\left. \left. \alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] (YD_{r.v} - YR_{r.v}) \right] ; \forall v \in V. w \in W
\end{aligned}$$

(128)

$$\begin{aligned}
ST_{\acute{v}}^D &\geq ST_v^D + \sum_{d \in RD} \sum_{\acute{d} \in SD} \left[\alpha \left(\frac{tt_{\acute{d}\acute{d}}^m + tt_{\acute{d}\acute{d}}^o}{2} \right) + (1 - \alpha) \left(\frac{tt_{\acute{d}\acute{d}}^p + tt_{\acute{d}\acute{d}}^m}{2} \right) \right] DRS_{\acute{v}.d.\acute{d}} + \left[\alpha \left(\frac{(ft_w^D)^m + (ft_w^D)^o}{2} \right) + \right. \\
&(1 - \alpha) \left. \left(\frac{(ft_w^D)^p + (ft_w^D)^m}{2} \right) \right] + \left[\alpha \left(\frac{ur_w^m + ur_w^o}{2} \right) + (1 - \alpha) \left(\frac{ur_w^p + ur_w^m}{2} \right) \right] GD_{w.\acute{v}} \left[\sum_{r \in R} \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1 - \right. \right. \\
&\left. \left. \alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] (YD_{r.\acute{v}} - YR_{r.\acute{v}}) \right] - M_T^D (1 - ZD_{v.\acute{v}}) - M_T^D (2 - DD_{v.\acute{d}} - DD_{\acute{v}.\acute{d}}) \\
&; \forall v, \acute{v} \in V (v < \acute{v}). w \in W
\end{aligned}$$

(129)

$$\begin{aligned}
ST_v^D &\geq ST_{\acute{v}}^D + \sum_{d \in RD} \sum_{\acute{d} \in SD} \left[\alpha \left(\frac{tt_{\acute{d}\acute{d}}^m + tt_{\acute{d}\acute{d}}^o}{2} \right) + (1 - \alpha) \left(\frac{tt_{\acute{d}\acute{d}}^p + tt_{\acute{d}\acute{d}}^m}{2} \right) \right] DRS_{v.d.\acute{d}} + \left[\alpha \left(\frac{(ft_w^D)^m + (ft_w^D)^o}{2} \right) + \right. \\
&(1 - \alpha) \left. \left(\frac{(ft_w^D)^p + (ft_w^D)^m}{2} \right) \right] + \left[\alpha \left(\frac{ur_w^m + ur_w^o}{2} \right) + (1 - \alpha) \left(\frac{ur_w^p + ur_w^m}{2} \right) \right] GD_{w.v} \left[\sum_{r \in R} \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1 - \right. \right. \\
&\left. \left. \alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] (YD_{r.v} - YR_{r.v}) \right] - M_T^D ZD_{v.\acute{v}} - M_T^D (2 - DD_{v.\acute{d}} - DD_{\acute{v}.\acute{d}}) ; \forall v, \acute{v} \in V (\acute{v} < v). w \in W
\end{aligned}$$

$$\sum_{n \in N} WD_{n.v} = 1 ; \forall v \in V \quad (130)$$

$$UT_{r.n} \geq (WD_{n.v} + YD_{r.v} - 1) ; \forall n \in N. r \in R. v \in V \quad (131)$$

(132)

$$\begin{aligned}
ST_v^D &\geq TE_n + \sum_{d \in RD} \sum_{\acute{d} \in SD} \left[\alpha \left(\frac{(ft_w^D)^m + (ft_w^D)^o}{2} \right) + (1 - \alpha) \left(\frac{(ft_w^D)^p + (ft_w^D)^m}{2} \right) \right] DRS_{v.d.\acute{d}} + \\
&\left[\alpha \left(\frac{(ft_w^D)^m + (ft_w^D)^o}{2} \right) + (1 - \alpha) \left(\frac{(ft_w^D)^p + (ft_w^D)^m}{2} \right) \right] + \left[\alpha \left(\frac{ur_w^m + ur_w^o}{2} \right) + (1 - \right. \\
&\left. \alpha) \left(\frac{ur_w^p + ur_w^m}{2} \right) \right] GD_{w.v} \left[\sum_{r \in R} \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1 - \alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] (YD_{r.v} - YR_{r.v}) \right] - M_T^D (1 - \\
&WD_{n.v}) ; \forall n \in N. v \in V. w \in W
\end{aligned}$$

(133)

$$\begin{aligned}
CD_r &\geq \left[\alpha \left(\frac{uc_v^m + uc_v^o}{2} \right) + (1 - \alpha) \left(\frac{uc_v^p + uc_v^m}{2} \right) \right] \left[\alpha \left(\frac{(d_{r.w}^D)^m + (d_{r.w}^D)^o}{2} \right) + (1 - \right. \\
&\left. \alpha) \left(\frac{(d_{r.w}^D)^p + (d_{r.w}^D)^m}{2} \right) \right] YD_{r.v} GD_{w.v} ; \forall r \in R. v \in V. w \in W
\end{aligned}$$

(134)

$$\begin{aligned}
CD_{\acute{r}} &\geq CD_r + \left[\alpha \left(\frac{uc_v^m + uc_v^o}{2} \right) + (1 - \alpha) \left(\frac{uc_v^p + uc_v^m}{2} \right) \right] \left[\alpha \left(\frac{(d_{r.\acute{r}}^D)^m + (d_{r.\acute{r}}^D)^o}{2} \right) + (1 - \alpha) \left(\frac{(d_{r.\acute{r}}^D)^p + (d_{r.\acute{r}}^D)^m}{2} \right) \right] - \\
&M_C^D (1 - XD_{r.\acute{r}}) - M_C^D (2 - YD_{r.v} - YD_{\acute{r}.v}) ; \forall r, \acute{r} \in R (r < \acute{r}). v \in V
\end{aligned}$$

(135)

$$CD_r \geq CD_{\acute{r}} + \left[\alpha \left(\frac{uc_v^m + uc_v^o}{2} \right) + (1 - \alpha) \left(\frac{uc_v^p + uc_v^m}{2} \right) \right] \left[\alpha \left(\frac{(d_{r,\acute{r}}^D)^m + (d_{r,\acute{r}}^D)^o}{2} \right) + (1 - \alpha) \left(\frac{(d_{r,\acute{r}}^D)^p + (d_{r,\acute{r}}^D)^m}{2} \right) \right] - M_C^D XD_{r,\acute{r}} - M_C^D (2 - YD_{r,v} - YD_{\acute{r},v}) ; \forall r, \acute{r} \in R (\acute{r} < r), v \in V$$

(136)

$$OC_v^D \geq CD_r + \left[\alpha \left(\frac{uc_v^m + uc_v^o}{2} \right) + (1 - \alpha) \left(\frac{uc_v^p + uc_v^m}{2} \right) \right] \left[\alpha \left(\frac{(d_{r,w}^D)^m + (d_{r,w}^D)^o}{2} \right) + (1 - \alpha) \left(\frac{(d_{r,w}^D)^p + (d_{r,w}^D)^m}{2} \right) \right] - M_C^D (1 - YD_{r,v}) ; \forall r \in R, v \in V, w \in W$$

(137)

$$TD_r \geq ST_v^D + \left(\frac{\left[\alpha \left(\frac{(d_{r,w}^D)^m + (d_{r,w}^D)^o}{2} \right) + (1 - \alpha) \left(\frac{(d_{r,w}^D)^p + (d_{r,w}^D)^m}{2} \right) \right]}{\left[\alpha \left(\frac{sp_v^m + sp_v^o}{2} \right) + (1 - \alpha) \left(\frac{sp_v^p + sp_v^m}{2} \right) \right]} \right) YD_{r,v} GD_{w,v} ; \forall r \in R, v \in V, w \in W$$

(138)

$$TD_{\acute{r}} \geq TD_r + \left[\alpha \left(\frac{(ft_r^D)^m + (ft_r^D)^o}{2} \right) + (1 - \alpha) \left(\frac{(ft_r^D)^p + (ft_r^D)^m}{2} \right) \right] + \left[\alpha \left(\frac{ur_r^m + ur_r^o}{2} \right) + (1 - \alpha) \left(\frac{ur_r^p + ur_r^m}{2} \right) \right] \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1 - \alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] + \left(\frac{\left[\alpha \left(\frac{(d_{r,\acute{r}}^D)^m + (d_{r,\acute{r}}^D)^o}{2} \right) + (1 - \alpha) \left(\frac{(d_{r,\acute{r}}^D)^p + (d_{r,\acute{r}}^D)^m}{2} \right) \right]}{\left[\alpha \left(\frac{sp_v^m + sp_v^o}{2} \right) + (1 - \alpha) \left(\frac{sp_v^p + sp_v^m}{2} \right) \right]} \right) - M_T^D (1 - XP_{r,\acute{r}}) - M_T^{PD} (2 - YD_{r,v} - YD_{\acute{r},v}) ; \forall r, \acute{r} \in R (r < \acute{r}), v \in V$$

(139)

$$TD_r \geq TD_{\acute{r}} + \left[\alpha \left(\frac{(ft_{\acute{r}}^D)^m + (ft_{\acute{r}}^D)^o}{2} \right) + (1 - \alpha) \left(\frac{(ft_{\acute{r}}^D)^p + (ft_{\acute{r}}^D)^m}{2} \right) \right] + \left[\alpha \left(\frac{ur_{\acute{r}}^m + ur_{\acute{r}}^o}{2} \right) + (1 - \alpha) \left(\frac{ur_{\acute{r}}^p + ur_{\acute{r}}^m}{2} \right) \right] \left[\alpha \left(\frac{q_{\acute{r}}^m + q_{\acute{r}}^o}{2} \right) + (1 - \alpha) \left(\frac{q_{\acute{r}}^p + q_{\acute{r}}^m}{2} \right) \right] + \left(\frac{\left[\alpha \left(\frac{(d_{r,\acute{r}}^D)^m + (d_{r,\acute{r}}^D)^o}{2} \right) + (1 - \alpha) \left(\frac{(d_{r,\acute{r}}^D)^p + (d_{r,\acute{r}}^D)^m}{2} \right) \right]}{\left[\alpha \left(\frac{sp_v^m + sp_v^o}{2} \right) + (1 - \alpha) \left(\frac{sp_v^p + sp_v^m}{2} \right) \right]} \right) - M_T^D XP_{r,\acute{r}} - M_T^D (2 - YD_{r,v} - YD_{\acute{r},v}) ; \forall r, \acute{r} \in R (\acute{r} < r), v \in V$$

(140)

$$AT_v^D \geq TD_r + \left[\alpha \left(\frac{(ft_r^D)^m + (ft_r^D)^o}{2} \right) + (1 - \alpha) \left(\frac{(ft_r^D)^p + (ft_r^D)^m}{2} \right) \right] + \left[\alpha \left(\frac{ur_r^m + ur_r^o}{2} \right) + (1 - \alpha) \left(\frac{ur_r^p + ur_r^m}{2} \right) \right] \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1 - \alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] + \left(\frac{\left[\alpha \left(\frac{(d_{r,w}^D)^m + (d_{r,w}^D)^o}{2} \right) + (1 - \alpha) \left(\frac{(d_{r,w}^D)^p + (d_{r,w}^D)^m}{2} \right) \right]}{\left[\alpha \left(\frac{sp_v^m + sp_v^o}{2} \right) + (1 - \alpha) \left(\frac{sp_v^p + sp_v^m}{2} \right) \right]} \right) - M_T^D (1 - YD_{r,v}) ; \forall r \in R, v \in V, w \in W$$

(141)

$$\sum_{r \in R} \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1 - \alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] YD_{r,v} \leq \left[\alpha \left(\frac{Q_v^m + Q_v^o}{2} \right) + (1 - \alpha) \left(\frac{Q_v^p + Q_v^m}{2} \right) \right] ; \forall v \in V, r \in R$$

(142)

$$\sum_{r \in R} \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1 - \alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] YD_{r,v} GD_{w,v} \leq \left[\alpha \left(\frac{Q_w^m + Q_w^o}{2} \right) + (1 - \alpha) \left(\frac{Q_w^p + Q_w^m}{2} \right) \right] ; \forall v \in V, w \in W$$

$$ZD_{v,\hat{v}} \leq 2 - WD_{n,v} - \sum_{\hat{n} < n} WD_{\hat{n},\hat{v}} ; \forall n \in N, v, \hat{v} \in V (v < \hat{v}) \quad (143)$$

$$ZD_{v,\hat{v}} \geq WD_{n,v} + \sum_{\hat{n} > n} WD_{\hat{n},\hat{v}} - 1 ; \forall n \in N, v, \hat{v} \in V (v < \hat{v}) \quad (144)$$

(145)

$$AT_v^P \geq (1 - \eta_P) \left[ST_v^P + \left(\frac{OC_v^P}{\left[\alpha \left(\frac{uc_v^m + uc_v^o}{2} \right) + (1 - \alpha) \left(\frac{uc_v^p + uc_v^m}{2} \right) \right] \left[\alpha \left(\frac{sp_v^m + sp_v^o}{2} \right) + (1 - \alpha) \left(\frac{sp_v^p + sp_v^m}{2} \right) \right]} \right) \right] + \sum_{r \in R} \left(\left[\alpha \left(\frac{(ft_r^P)^m + (ft_r^P)^o}{2} \right) + (1 - \alpha) \left(\frac{(ft_r^P)^p + (ft_r^P)^m}{2} \right) \right] + \left[\alpha \left(\frac{lr_r^m + lr_r^o}{2} \right) + (1 - \alpha) \left(\frac{lr_r^p + lr_r^m}{2} \right) \right] \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1 - \alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] \right) YP_{r,v} ; \forall r \in R, v \in V, w \in W$$

(146)

$$AT_v^P \leq (1 + \eta_P) \left[ST_v^P + \left(\frac{OC_v^P}{\left[\alpha \left(\frac{uc_v^m + uc_v^o}{2} \right) + (1 - \alpha) \left(\frac{uc_v^p + uc_v^m}{2} \right) \right] \left[\alpha \left(\frac{sp_v^m + sp_v^o}{2} \right) + (1 - \alpha) \left(\frac{sp_v^p + sp_v^m}{2} \right) \right]} \right) \right] + \sum_{r \in R} \left(\left[\alpha \left(\frac{(ft_r^P)^m + (ft_r^P)^o}{2} \right) + (1 - \alpha) \left(\frac{(ft_r^P)^p + (ft_r^P)^m}{2} \right) \right] + \left[\alpha \left(\frac{lr_r^m + lr_r^o}{2} \right) + (1 - \alpha) \left(\frac{lr_r^p + lr_r^m}{2} \right) \right] \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1 - \alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] \right) YP_{r,v} ; \forall r \in R, v \in V, w \in W$$

(147)

$$AT_v^D \geq (1 - \eta_D) \left[ST_v^D + \left(\frac{OC_v^D}{\left[\alpha \left(\frac{uc_v^m + uc_v^o}{2} \right) + (1 - \alpha) \left(\frac{uc_v^p + uc_v^m}{2} \right) \right] \left[\alpha \left(\frac{sp_v^m + sp_v^o}{2} \right) + (1 - \alpha) \left(\frac{sp_v^p + sp_v^m}{2} \right) \right]} \right) \right] + \sum_{r \in R} \left(\left[\alpha \left(\frac{(ft_r^D)^m + (ft_r^D)^o}{2} \right) + (1 - \alpha) \left(\frac{(ft_r^D)^p + (ft_r^D)^m}{2} \right) \right] + \left[\alpha \left(\frac{ur_r^m + ur_r^o}{2} \right) + (1 - \alpha) \left(\frac{ur_r^p + ur_r^m}{2} \right) \right] \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1 - \alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] \right) YD_{r,v} ; \forall r \in R, v \in V, w \in W$$

(148)

$$AT_v^D \leq (1 + \eta_D) \left[ST_v^D + \left(\frac{OC_v^D}{\left[\alpha \left(\frac{uc_v^m + uc_v^o}{2} \right) + (1 - \alpha) \left(\frac{uc_v^p + uc_v^m}{2} \right) \right] \left[\alpha \left(\frac{sp_v^m + sp_v^o}{2} \right) + (1 - \alpha) \left(\frac{sp_v^p + sp_v^m}{2} \right) \right]} \right) \right] + \sum_{r \in R} \left(\left[\alpha \left(\frac{(ft_r^D)^m + (ft_r^D)^o}{2} \right) + (1 - \alpha) \left(\frac{(ft_r^D)^p + (ft_r^D)^m}{2} \right) \right] + \left[\alpha \left(\frac{ur_r^m + ur_r^o}{2} \right) + (1 - \alpha) \left(\frac{ur_r^p + ur_r^m}{2} \right) \right] \left[\alpha \left(\frac{q_r^m + q_r^o}{2} \right) + (1 - \alpha) \left(\frac{q_r^p + q_r^m}{2} \right) \right] \right) YD_{r,v} ; \forall r \in R, v \in V, w \in W$$

$$\sum_{\substack{d \in RD \\ d < n}} DP_{v,d} \geq \sum_{\substack{\hat{n} \in N \\ \hat{n} < n}} WP_{\hat{n},v} \quad ; \forall v \in V . n \in N (n \leq |RD|) \quad (149)$$

3.1.2. Fuzzy solution approach

Zimmermann [34] firstly introduced fuzzy solution approach in order to solve multi-objective programming problems. The related literature review indicates the appearance of several different approaches for investigating possibility models [29]; [35]; [36]. Inspired by [37], the present research tries to solve the presented deterministic model. The above-mentioned methods steps are as follows:

Step 1: Determine appropriate possibility triangular or trapezoidal distributions for problem parameters, and then formulate the problem model.

Step 2: Convert the inaccurate objective function of model into the deterministic function model through expected value according to inaccurate parameters.

Step 3: Calculate α -value (minimum acceptable feasibility degree of decision vector) and convert fuzzy constraints of problem into the deterministic constraints, and then formulate auxiliary deterministic equivalent model of problem.

Step 4: Determine α -positive ideal solution (α -PIS) and α -negative ideal solution (α -NIS) for each objective function and feasible α -level.

Step 5: Determine linear membership function for each objective function as follows:

$$\mu_1(x) = \begin{cases} 1 & \text{if } Z_1 \leq Z_1^{\alpha-Pis} \\ \frac{Z_1^{\alpha-Nis} - Z_1}{Z_1^{\alpha-Nis} - Z_1^{\alpha-Pis}} & \text{if } Z_1^{\alpha-Pis} \leq Z_1 \leq Z_1^{\alpha-Nis} \\ 0 & \text{if } Z_1 > Z_1^{\alpha-Nis} \end{cases} \quad (150)$$

$$\mu_2(x) = \begin{cases} 1 & \text{if } Z_2 \leq Z_2^{\alpha-Pis} \\ \frac{Z_2^{\alpha-Nis} - Z_2}{Z_2^{\alpha-Nis} - Z_2^{\alpha-Pis}} & \text{if } Z_2^{\alpha-Pis} \leq Z_2 \leq Z_2^{\alpha-Nis} \\ 0 & \text{if } Z_2 > Z_2^{\alpha-Nis} \end{cases} \quad (151)$$

$$\mu_3(x) = \begin{cases} 1 & \text{if } Z_3 \leq Z_3^{\alpha-Pis} \\ \frac{Z_3^{\alpha-Nis} - Z_3}{Z_3^{\alpha-Nis} - Z_3^{\alpha-Pis}} & \text{if } Z_3^{\alpha-Pis} \leq Z_3 \leq Z_3^{\alpha-Nis} \\ 0 & \text{if } Z_3 > Z_3^{\alpha-Nis} \end{cases} \quad (152)$$

In which $\mu_h(x)$ is the satisfaction degree with objective function h.

Step 6: Convert the three-objective deterministic equivalent model into a mixed integer linear programming (MILP) through a cumulative function. The above-mentioned cumulative function is as follows:

$$\begin{aligned} \max \lambda(x) &= \gamma \lambda_0 + (1 - \gamma) \sum_h \theta_h \mu_h(x) \\ \text{s.t. } \lambda_0 &\leq \mu_H(x) \quad . \quad h = 1,2,3 \\ x &\in F(x) \quad . \quad \lambda_0 \text{ and } \lambda \in [0,1] \end{aligned} \quad (153)$$

$F(x)$ represents that the feasibility section contains the deterministic equivalent model variables; and θ and γ indicate the importance of objective function h and the correction factor, respectively. It is worth mentioning that $\lambda_0 = \min_h \{\mu_h(x)\}$ is obtained by objective functions as the minimum degree of satisfaction.

Step 7: Calculate γ and θ_h values and solve the single-objective mixed integer linear programming model. Stop if a decision maker is met by the current solution, otherwise create another compromise solution by changing γ and α values (and θ_h , if necessary) and shifting to step 3.

3.2. Heuristic approach

Inspired by local and structural search approaches for the efficient and effective search through the solution space, metaheuristic algorithms are offered as a new generation of approximation algorithms relying on heuristic methods. Unique search strategies are mainly used in metaheuristic

algorithms to prevent them from being trapped in local optimal solutions. Being popular, the existed techniques are extended to plentiful current ones to apply. No matter, what kind of techniques they are, simple local search approaches or complicated learning processes are always involved. Therefore, this is the main philosophy to consider numerous metaheuristic algorithms such as the ant colony optimization and evolutionary computation algorithms including the genetic algorithm, the simulated annealing algorithm, tabu search.

Meta-heuristics characteristics

Regarding high levels of complexity and computation, operations planning and cross docks scheduling can be taken into account as combinatorial optimization and NP-hard problems. The most difficult part of traditional approaches and conventional optimization tools is to find an optimal solution for a high dimensional problem within a reasonable computation time traditional approaches and conventional optimization tools. Solving such complicated problems cannot be expected by polynomial algorithms considering a reasonable computation time. Hence, accurate methods are not capable of finding solutions for problems in high dimensions. To provide an optimal solution within a reasonable time, heuristic and metaheuristic algorithms give us a hand. Genetic algorithm (GA) and ant colony optimization (ACO) are offered respectively as two meta-heuristics to solve the proposed model.

3.2.1. A GA for cross-docking problem

Local search procedures are the main bases for GA as the most favorable class of evolutionary algorithms. Functioning as follows, this algorithm considers several features, such as population, chromosomes, genes, reproduction parameters and generation to evolve its search procedures: firstly, an initial population consisting of sets of chromosomes and genes is created at random; secondly, genetic reproducing parameters such as crossover and mutation are applied to the primary population to achieve to the related descendant populations. We do the same in each iteration (generation) of the algorithm, till the best possible solution is obtained regarding a related stopping criterion. Being combined via crossover or mutation operations in the mating pool, the GA can provide us a wide variety of strategies and alternatives in order to create more chromosomes.

Here is the right place to offer the overall framework of our applied GA as follows: (1) we create random chromosomes (sequences) within initialization phase to create the population by merging them; (2) we set the value of an objective function as the fitness function to evaluate the generated solutions (chromosomes). Since we aim at minimizing the objective function, the utility function is defined as follows:

$$fitness\ function = \frac{1}{objective\ value} \quad (154)$$

(3) roulette wheel selection and the tournament selection are two parameters of selection operator that select a chromosome; (4) reproduction scheme including three major approaches, elitism, crossover and mutation; (5) The termination or stopping condition is defined as follows: The First Priority: The convergence of the algorithm on a specific solution. The Second Priority: No progress. In other words, if the genetic algorithm continues running on previous chromosomes and does not improve after x iterations, it should be terminated. The stopping condition is supposed to be $x= 30$ in this paper. The Third Priority: The number of iterations. Not meeting the above conditions, the number of iterations is set to a specific value. In the present paper, the termination condition included 300 generations.

Displaying the Solutions to the Proposed Genetic Algorithm

Despite the type and number of vehicles in each dock, research problem includes several constraints such as capacity constraints on cross docks and vehicles. Regarding the mentioned logics and constraints, chromosomes were shown more flexibly in this study to eliminate infeasible solutions. In this study, solutions were supposed to take the following steps:

1. Determining the way through which Demands are Allocated to Vehicles in the Loading Process:

This section is dedicated to find an answer for the above question. This part of the chromosome is a matrix made up of two dimensions indicating the number of demands and the number of vehicles. The order of allocation and sequence of routes by every vehicle in the loading process are presented in Table (1). As an example, the demand points 16, 10, 36, and 23 were allocated to vehicle 1 and are set to be taken in the same sequence.

Table (1). Displaying a Solution in the Genetic Algorithm

Determining the way through which Demands are Allocated to Vehicles in the Loading Process																			
q_0	9	40	5	7	16	21	35	37	26	33	8	10	25	34	29	17	19	30	31
q_1	2	3	9	3	1	8	3	9	10	5	3	1	5	4	4	6	7	1	7

- Determining the way through which Demands are Allocated to Vehicles in the Delivery process:

This section evaluates the way through which demands are allocated to vehicles. This part of the chromosome is a matrix of two dimensions which shows the number of demands and the number of vehicles. The order of allocation and sequence of routes taken by every vehicle in the delivery process are presented in Table (2). The applied allocation method is the same as that of the delivery phase. We corrected q_4 in this paper in several ways. We should deliver the same demand points, allocated to a warehouse by vehicles in the previous phase with the help of the same vehicles.

Table (2). Displaying a Solution in the Genetic Algorithm

Determining the way through which Demands are Allocated to Vehicles in the Delivery process																				
q_2	13	31	2	14	26	19	36	21	32	15	33	17	20	39	23	10	38	5	12	7
q_4	5	7	9	8	4	4	6	6	10	8	4	1	8	2	1	1	2	8	5	9

- Determining the Way through which Vehicles are Allocated to Cross-docks:

This section examines how to allocate vehicles to cross-docks. This part of the chromosome is a matrix of two dimensions which are representatives for the number of vehicles and the number of warehouses, respectively. The way through which chromosomes are displayed to allocate vehicles to cross-docks is presented in Table (3). For instance, vehicles 1, 3, 6, 8, and 9 were allocated to cross-docks 2.

Table (3). Displaying a Solution in the Genetic Algorithm

Determining the way through which vehicles are allocated to cross-docks										
q_3	1	2	3	4	5	6	7	8	9	10
w	2	1	2	1	1	2	1	2	2	1

- Determining the Way through which Vehicles Are Allocated to Receiving Doors:

Determination of a way through which vehicles can be allocated to receiving doors of cross-docks is the main purpose of this section. This part of the chromosome is a matrix of two dimensions defining the number of vehicles and the number of receiving doors of cross-docks. The way through which vehicles can be allocated to receiving doors of cross-docks are demonstrated in Table (4) like the following array. For instance, Vehicles 8, 6, 4, 2, and 1 were allocated to Receiving Door 2.

Table (4). Displaying a Solution in the Genetic Algorithm

Determining the Way through which Vehicles Are Allocated to Receiving Doors										
q_5	8	4	10	5	6	7	3	2	9	1
q_6	2	3	1	3	2	1	2	2	3	2

- Determining an appropriate method to allocate vehicles to Sending Doors:

This section mainly aims at finding an appropriate method to allocate vehicles to sending doors of cross-docks. This part of the chromosome is a matrix of two dimensions indicating the number of vehicles and the number of sending doors. Table (5) demonstrates an appropriate method to display chromosomes to allocate vehicles to sending doors of cross-docks like the following array. For instance, Vehicles 7 and 9 were allocated to Sending Door 3.

Table (5). Displaying a Solution in the Genetic Algorithm

Determining the Way through which Vehicles Are Allocated to Receiving Doors										
q_7	8	4	7	2	10	9	3	1	5	6
q_8	1	2	3	1	2	3	2	2	2	1

3.2.2. A SACO for cross-docking problem

Firstly developed by [38], Ant Colony Optimization (ACO) can be considered as another class of population-based meta-heuristics aiming at offering solution for combinatorial optimization problems. Inspired by the social behavior of ant colonies, this algorithm copies the social behavior of ant colonies through which each ant tries to communicate with others by laying pheromone trails. Since it detects the solution space and creates new solutions iteratively by copying the real behavior of ants, the pheromone trail plays a crucial role in ACO.

Simple-ACO (SACO) is an elementary version of ACO algorithm which assumes similarity between artificial ants' behavior and their natural counterpart. There is an experimental way to test this algorithm with the help of finding shortest paths in graphs as a simple example. Despite the fact that the shortest path problems can be solved with deterministic algorithms in polynomial time, it is an interesting problem for studying the behavior of ACO algorithms, since (i) the problem is solved by real ant colonies, (ii) due to its simplicity and the fact that algorithm behavior is not obscured by technicalities of the problem under consideration, and (iii) significant features are expected in solving this simple problem to be strengthened while attacking much more difficult combinatorial optimization problems.

It seems that finding the shortest path on a graph $G = (N,A)$ is the first step to be considered. S-ACO exploits a set of variables $T = \tau_{ij}(t)$ called artificial pheromone trails that are associated to the arcs (i,j) of the graph G . Pheromone trails are read and written by the ants. The amount (intensity) of each pheromone trail is proportional to the utility, as estimated by the ants, of using the corresponding arc to build good solutions.

The responsibility of each ant is to start from the source node and create a candidate solution for the shortest path problem considering a step-by-step decision policy in S-ACO. Local pheromone information storing at a node itself and/or on its outgoing arcs, is read (sensed) by the ant and used in a stochastic way to determine which node should move next: being located at a node i , an ant k uses the pheromone trails τ_{ij} to compute the probability p_{ij}^k of choosing j as the next node:

$$p_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha}{\sum_{j \in N_i^k} \tau_{ij}^\alpha} & \text{if } j \in N_i^k \\ 0 & \text{if } j \notin N_i^k \end{cases} \quad (155)$$

In which N_i^k is the feasible neighborhood of ant k which is located in node i . (to avoid division by zero at the beginning of the search process, an amount of pheromone $\tau_0 = 1$ is assigned to all the arcs of the graph G). The feasible neighborhood N_i^k of ant k located in node i contains all the nodes directly connected to node i , except for the predecessor of node i in S-ACO (that is, the last node ant k visited before moving to i). The ants are tempted to avoid returning to the same node they visited immediately before node i considering this way. Only in case N_i^k is empty (corresponding to a dead end in the graph), node i 's predecessor is included into N_i^k . It is worth mentioning that this decision policy can lead the ants to enter a loop.

Applying its decision policy, an ant repeatedly hops from node to node and stops moving when reaches the destination node. The time step at which ants reach the destination node may differ from ant to ant, due to differences among the ants' paths (no doubt, ants traveling on shorter paths will reach their destinations faster). The ants remove loops they might get entangled in so far while searching for the destination node when they get to it, and then take deterministic retracing backwardly to the source node step by step being exonerated from loops.

The ant adds pheromone to the edges it passes when it gets back to the source: the generic ant k deposits an amount $\Delta\tau^k$ of pheromone on each visited arc during its return trip (in the loop-free path). The pheromone value τ_{ij} will be updated if ant k at time t passes the arc (i,j) in particular using the following equation:

$$\tau_{ij}(t) \leftarrow \tau_{ij}(t) + \Delta\tau^k \quad (156)$$

We can also make $\Delta\tau^k$ a function of the generated solutions quality which is affected by the path length- the shorter the path the more the pheromone deposited by the ant as another complex method for updating the pheromone trails. In general, the amount of pheromone deposited by an ant is assumed to be a non-increasing function of the path length. In S-ACO, in particular, an ant deposits an amount of $\frac{1}{L^k}$, where L^k is the length of ant k 's path. We also expect pheromone trails to “evaporate” in order to avoid a quick convergence of all the ants towards a sub-optimal path. It should also be considered that the results reveal the hidden part of evaporation, even when real pheromone trails evaporate, evaporation fails to perform correctly in shortest path cases including real ants. We assume evaporation as an important tool for obtaining desired results with artificial ants regarding this fact that optimization problems faced artificial ants are much more complex than those of real ants. Hence, the existence of a mechanism for artificial ants is crucial to provide a “learning form” of the problem structure. Removing the memory of bad choices in the past, evaporation provides a platform for learning of new policies.

To stick to a practical approach, we calculate evaporation by decreasing pheromone trails at exponential speed, applying it to all pheromone trails at each iteration of the algorithm with the help of the following equation:

$$\tau_{ij} \leftarrow (1 - \rho)\tau_{ij} \quad \forall (i,j) \in A \quad (157)$$

Where $\rho \in (0,1]$ is a parameter.

It is worth mentioning that pheromone trail evaporation does not play any role while facing real ants. Thus, we test the performance of S-ACO when setting $\rho = 0$, that is in an experiment, when there is no evaporation. We conclude that the more complex the graph, the more important the role of pheromone evaporation to obtain the desired behavior of convergence on a shortest path. It should be considered that if the pheromone trails evaporate completely, that is, when $\rho = 1$, then the algorithm will be abated to a random search. We should revise the cross-docking problem to let it be admitted by the proposed ACO algorithm, there is a strong similarity between the trucks embedded in the travelling salesman problem (TSP) and these ants. In other words, each of the trucks stands for each of the ants. In order to clarify this algorithm, we are going to get through the main procedure.

We present different steps of the S-ACO algorithm in this section [39]:

Step 1: Initiating $\tau_{ij}(0)$ randomly.

Step 2: Repeating Step 3 and Step 4 for all of the n_k ants.

Step 3: Creating $x^k(t)$ route for the k^{th} ant in accordance with Equation (155).

Step 4: Calculating the fitness of the created route ($f(x^k(t))$).

Step 5: Using the process of pheromone evaporation to all of the (i, j) edges in the graph of possible routes in accordance with Equation (157).

Step 6: Repeating Step 7 for all of the n_k ants.

Step 7: Increasing the pheromone of each edge like (i, j) on the route created by the k^{th} ant in accordance with Equation (156) ($\Delta\tau^k = \frac{1}{f(x^k(t))}$).

Step 8: Repeating Steps 2-7 until the termination condition is convinced.

In the S-ACO algorithm, $x^k(t)$ is the solution found at t , and $f(x^k(t))$ shows the quality of solution.

As mentioned earlier in this paper, generating solution in the ant colony optimization is considered to be the same as other permutation problems such as TSP. nonetheless, regarding the method of displaying solutions, there are 9 strings. Colony optimization acts similar to genetic algorithm in applying methods for offering solutions.

Computational Results

This section is trying to analyze the results of solving the proposed model. Due to test the accuracy of the proposed model, several problems are solved in GAMS. This model mainly aims at minimizing costs. Models outputs and objective functions value are presented at the last part of this section. Regarding this fact that the majority of model parameters are fuzzy and it also includes parameters of right values and technological coefficients, and considers objective functions and main deterministic problem variables as constraints, a two-phase approach is applied to solve the proposed fuzzy model. We transform the initial fuzzy model into a deterministic equivalent auxiliary model in the first phase. Then we apply a fuzzy method to obtain the final preferred compromise solution, in the second step.

In order to generate triangular fuzzy numbers, we also estimate three sensitive points (the most probable value, the pessimistic value, and the optimistic value). To achieve this purpose, we generated the most probable value (c^m) of every parameter at first. In the next step, we used normal distribution to generate two random numbers (r_1, r_2) between 0.2 and 0.8 without changing the generality of problem. Then, the pessimistic (c^p) and optimistic (c^o) values were calculated using the fuzzy numbers [28]. Considering numerical solutions different dimensions, it included 4 cross docks, 3 doors (entrance and exit), 8 types of vehicles, and 40 types of customer demands. Parameters such as the speed of a vehicle (v), vehicles cost per a distance unit (uc_v), the cargo size for a request (q_r), loading locations and deliveries positions, cross docks position, other parameters constant values and the solution outputs are presented in Tables (6) – (13). The problem of different dimensions solved in GAMS is presented in Table (13) considering the following parameters. Input values of parameters helped us in solving these problems. Being solved in GAMS, The model resulted in answers to the objective functions that are analyzed in the fuzzy mode and presented in Tables (14) – (16) per different α parameters. Then, we employed metaheuristic algorithms such as genetic algorithm and ant colony optimization to solve the problem and analyzed their results using MATLAB outputs.

Table (6). The speed of a vehicle (v)

v	1	2	3	4	5	6	7	8
sp_v	70	70	70	70	70	70	70	70

Table (7). Vehicle transportation time from a receiving door ($d \in RD$) to a sending door ($\hat{d} \in SD$)

$tt_{d\hat{d}}$	RD1	RD2	RD3
RD1	2	4	8
RD2	4	2	5
RD3	7	6	2

Table (8). Vehicles cost per a distance unit (uc_v)

v	1	2	3	4	5	6	7	8
uc_v	200	230	210	270	200	230	210	270

Table (9). The cargo size for a request (q_r)

r	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
q_r	10	7	13	19	26	3	5	9	16	16	12	19	23	20	8	19	2	12	17	9
r	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
q_r	11	18	29	12	8	15	22	7	11	8	14	9	17	12	14	10	17	6	21	14

Table (10). Loading locations and deliveries positions

	X	Y	X	Y	X	Y	X	Y	
r	Loading position		Deliveries position		r	Loading position		Deliveries position	
1	41	49	20	20	17	5	30	30	42
2	35	17	31	52	18	20	40	40	3
3	55	45	24	12	19	15	60	60	5
4	55	20	35	40	20	45	65	65	56
5	15	30	41	37	21	45	20	20	68
6	25	30	53	52	22	45	10	10	69
7	20	50	45	30	23	55	5	5	48
8	10	43	40	25	24	44	22	22	50
9	55	60	11	14	25	28	25	25	39
10	30	60	65	7	26	40	47	22	39
11	20	42	60	12	27	48	23	31	33
12	50	35	13	52	28	26	29	50	20
13	30	25	63	65	29	18	22	18	43
14	15	10	47	47	30	45	38	50	29
15	30	5	40	60	31	53	43	18	15
16	10	20	20	55	32	40	19	27	42
33	29	51	60	41	37	16	24	71	8
34	20	36	39	22	38	47	85	17	83
35	50	25	45	42	39	21	66	5	74
36	67	19	37	85	40	74	31	30	7

Table (11). Cross docks position

w	X	Y
	The position of cross docks	
1	35	35
2	40	40
3	45	45
4	50	50
5	55	55

Table (12). Other parameters constant values

ft_r^P	0.5	lr_r	0.2	ft_w^P	0.5	lr_w	0.5
ft_r^D	0.5	ur_r	0.2	ft_w^D	0.5	ur_w	0.5

Table (13). Solution outputs

									Objective function			Time(s)		
	W	R	V	RD	SD	Q_w	Q_v	z_1	z_2	z_3	z_1	z_2	z_3	
1	2	8	2	2	2	150	100	166173.637	29304.471	166229.651	16	15	16	
2	2	9	3	2	2	150	100	318251.243	55180.808	318358.691	16	15	31	
3	2	10	3	2	2	150	100	368669.011	59195.042	368793.394	16	16	31	
4	2	11	3	2	2	150	100	378523.958	62444.011	378651.872	31	32	15	
5	2	12	3	2	2	150	100	434579.378	68074.483	434726.175	16	31	16	
6	2	8	2	3	3	150	100	166173.637	29304.471	166229.651	15	15	15	
7	2	9	3	3	3	150	100	318251.243	55180.808	318358.691	16	15	16	
8	2	10	3	3	3	150	100	368669.011	59195.042	368793.394	31	32	31	
9	2	11	3	3	3	150	100	378523.958	62444.011	378651.872	32	0.031	31	
10	2	12	3	3	3	150	100	434579.378	68074.483	434726.175	16	31	31	
11	2	14	3	3	3	150	100	521462.033	82596.812	521638.012	32	31	31	
12	2	16	4	3	3	150	100	897492.505	134105.579	897786.751	62	47	105.22	

There is obviously many differences between the fuzzy mode and the definite mode in terms of objective functions values. This difference proves the importance of considering uncertainty. Furthermore, the problem is not feasible for certain values of α . This problems solution is offered in Table (14).

Table (14). Objective Functions solutions for Different Parameters of α in the Fuzzy Mode

α	0.4	0.5	0.6	0.7	0.8	0.9	0.95	1
z_1	8406835.676	7354925.293	6371896.774	5458032.107	4613174.659	3837059.586	3474780.440	3129893.391
z_2	1215756.610	1063283.697	920818.388	788411.837	665978.677	553518.909	501029.047	451032.532
z_3	8409594.156	7357338.019	6373986.425	5459821.455	4614686.417	3838316.386	3475918.247	3130917.928

These tables prove our claim about the significant difference between the definite and fuzzy modes, and consequently, the importance of considering uncertainty in the model.

Due to provide solutions for the fuzzy multi-objective model, we applied the proposed approach. We consider different values of θ were considered; nonetheless, a preferable value is selected by the decision maker. In order to obtain the functions weights in the presence of different objective functions, we can apply other decision-making methods such as AHP. Tables (15) and (16) represent the sensitivity analysis (SA) performed on the problems parameters within different positions. In other words, the values of membership functions are presented in Table (15) for different parameters of α , and the final results of solving the model for α , β , and γ are presented in Table (16):

Table (15). The Values of the Membership Functions for Every Objective Function for Different Values of α

α	μ_1	μ_2	μ_3
0.8	0.601384482	0.714405423	0.600220716
0.9	0.230709369	0.252944375	0.229718001
0.95	0.155614454	0.124077384	0.154744266
1	0.081764138	0.028796304	0.080958103

Table (16). The Final Results of Solving the Model for α , β , and γ

α	γ	θ	Z_1	Z_2	Z_3	λ	α	γ	θ	Z_1	Z_2	Z_3	λ
0.8	0.9	(0.2*0.35*0.45)	4988200	719680	4989100	0.5732	0.4	0.4	(0.2*0.35*0.45)	4898300	692830	4898500	0.0905
		(0.8*0.15*0.05)				0.5972			(0.8*0.15*0.05)				0.1203
		(0.5*0.23*0.27)				0.5840			(0.5*0.23*0.27)				0.1039
		(0.37*0.33*0.3)				0.5822			(0.37*0.33*0.3)				0.1017
	(0.02*0.58*0.4)		0.5762		(0.02*0.58*0.4)		-						
	0.5	(0.2*0.35*0.45)	4848400	697160	4848700	0.1780	0.5	0.5	(0.2*0.35*0.45)	4898300	692830	4898500	0.0961
		(0.8*0.15*0.05)				0.2239			(0.8*0.15*0.05)				0.1209
		(0.5*0.23*0.27)				0.1987			(0.5*0.23*0.27)				0.1073
		(0.37*0.33*0.3)				0.1952			(0.37*0.33*0.3)				0.1054
	(0.02*0.58*0.4)		-		(0.02*0.58*0.4)		-						
	0.6	(0.2*0.35*0.45)	4848400	697160	4848700	0.1883	0.6	0.6	(0.2*0.35*0.45)	4898300	692830	4898500	0.1017
		(0.8*0.15*0.05)				0.2251			(0.8*0.15*0.05)				0.1215
(0.5*0.23*0.27)		0.2049				(0.2*0.35*0.45)			0.1106				
(0.37*0.33*0.3)		0.2021				(0.8*0.15*0.05)			0.1091				
(0.02*0.58*0.4)		-		(0.5*0.23*0.27)	0.95	(0.5*0.23*0.27)	0.8623						
0.9	0.7	(0.2*0.35*0.45)	4848400	697160	4848700	0.1987	0.7	0.7	(0.2*0.35*0.45)	4898300	692830	4898500	0.1073
		(0.8*0.15*0.05)				0.2262			(0.8*0.15*0.05)				0.1222
		(0.5*0.23*0.27)				0.2111			(0.5*0.23*0.27)				0.1140
		(0.37*0.33*0.3)				0.2090			(0.37*0.33*0.3)				0.1129
	(0.02*0.58*0.4)		0.2021		(0.02*0.58*0.4)		0.1091						
	0.8	(0.2*0.35*0.45)	4848400	697160	4848700	0.2090	0.8	0.8	(0.2*0.35*0.45)	4898300	692830	4898500	0.1129
		(0.8*0.15*0.05)				0.2274			(0.8*0.15*0.05)				0.1228
		(0.5*0.23*0.27)				0.2173			(0.5*0.23*0.27)				0.1173
		(0.37*0.33*0.3)				0.2159			(0.37*0.33*0.3)				0.1166
	(0.02*0.58*0.4)		0.2113		(0.02*0.58*0.4)		0.1141						
	0.9	(0.2*0.35*0.45)	4848400	697160	4848700	0.2193	0.9	0.9	(0.2*0.35*0.45)	4898300	692830	4898500	0.1184
		(0.8*0.15*0.05)				0.2285			(0.8*0.15*0.05)				0.1234
(0.5*0.23*0.27)		0.2235				(0.5*0.23*0.27)			0.1207				
(0.37*0.33*0.3)		0.2228				(0.37*0.33*0.3)			0.1203				
(0.02*0.58*0.4)		0.2205		(0.02*0.58*0.4)		0.1191							

These tables evidently prove that objective functions are highly sensitive to changes in γ for different parameters of α . When $\alpha= 0.9$, the sensitivity is higher. In other cases, changes of γ are relatively less effective. In order to make decisions for selecting a position, the decision maker considers all of the certain conditions. The degree of satisfaction depends on the importance of the objective function. The $Z3$ is the most important objective function and also $Z1$ and $Z2$ are actually relative measures from $Z3$. Therefore the compromise solution with highest satisfaction degree for $Z3$ is of particular interest and it makes sense to establish levels of satisfaction in this case.

The Results of Solving the Model by Using the Metaheuristic Algorithms (GA and Ant Colony Optimization):

We apply the proposed input parameters to generate sample problems for evaluating algorithms performances. We include 4 cross docks, 3 doors (entrance and exit), 8 types of vehicles, and 40 types of customer demands in our numerical solution.

In general, the required time for accessing optimal solutions or the solution space obtained by an accurate approach are regarded as the main criteria applied for determining dimensions of the problem. We will consider problem as a low-dimensional one provided that the accurate approach is able to find an optimal solution to the model in less than one hour or one and half an hour. In the contrary, we regard a problem as high-dimensional one provided that the accurate approach cannot even find an optimal solution during the same period. Considering it as an NP-hard problem, a time limitation was defined in executing GAMS to obtain an accurate solution. This interval lasted for one hour or 3600 seconds to clarify that if GAMS cannot solve the problem in the preset interval, the process will be stopped. Then, due to test the accuracy of the solution for the proposed model, we applied the genetic algorithm and ant colony optimization to solve several problems in MATLAB.

In the next step, we performed many tests to regulate the algorithms with the help of different values of parameters. Finally, we obtained the best results of genetic algorithm and ant colony optimization using the following set of values which are presented in Tables (17) and (18).

Table (17). Parameters values in the Genetic Algorithm

Population Size	$nPop=30$
Crossover Percentage	$pc=0.8$
Number of Off-springs (Parents)	$nc=2*round(pc*nPop/2)$
Mutation Percentage	$pm=0.3$
Number of Mutants	$nm=round(pm*nPop)$
Mutation Rate	$mu=0.02$
Selection Pressure	$beta=8$

Table (18). Values of Parameters in the Ant Colony Algorithm

Number of Ants (Population Size)	$nAnt=30$
Initial Phromone	$tau00=1$
Phromone Exponential Weight	$alpha=1$
Evaporation Rate	$rho=0.05$
	$Q=1$

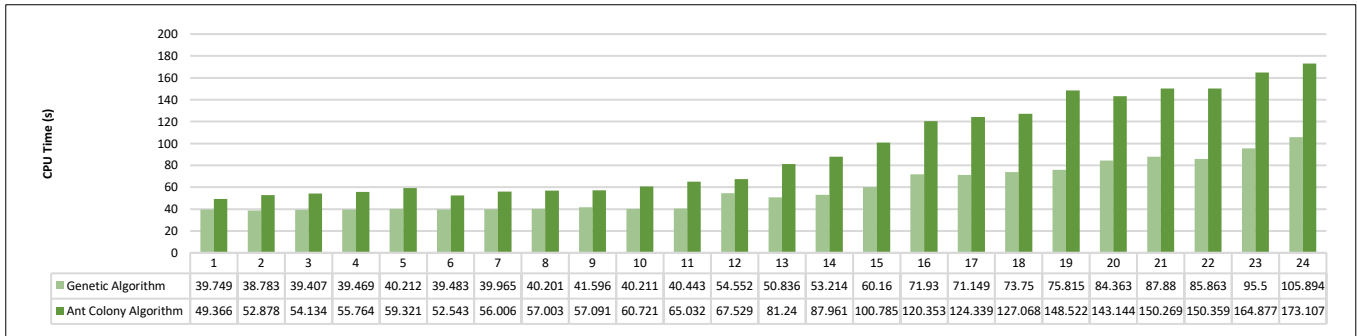
The features of the system on which the tests were performed are as follows: a computer with a processor operating at 2.30 GHz and 6400 GB of RAM. As mentioned earlier in this paper, the genetic algorithm and ant colony optimization led us to design the appropriate method in MATLAB. It was assumed that the test would determine and evaluate the performance of the proposed algorithms in different conditions. Validating the genetic algorithm and the ant colony optimization in real (high) dimensions problem resulted in the values in Table (19). After running on the 12 defined problems, the optimal results of each problem in two methods are given in Table (19).

Table (19). The Computational Results of the Proposed Algorithms in Sample Problems

	ACO optimum results								GA optimum results		
	<i>W</i>	<i>R</i>	<i>V</i>	<i>RD</i>	<i>SD</i>	<i>Q_w</i>	<i>Q_v</i>	<i>ACO</i>	<i>CPU Time (s)</i>	<i>GA</i>	<i>CPU Time (s)</i>
1	2	18	4	3	3	150	100	229927.8807	81.240	138893.5181	50.836
2	2	20	4	3	3	150	100	259149.7328	87.961	176261.5039	53.214
3	2	22	5	3	3	150	100	304650.2628	100.785	247566.2008	60.160
4	3	24	6	3	3	300	150	313717.9571	120.353	278415.6472	71.930
5	3	26	6	3	3	300	150	348449.0109	124.339	287649.5143	71.149
6	3	28	6	3	3	300	150	359000.0415	127.068	276982.581	73.750
7	3	30	6	3	3	300	150	370536.2703	148.522	277125.5252	75.815
8	3	32	7	3	3	300	150	402247.6574	143.144	324996.7977	84.363
9	3	34	7	3	3	300	150	443071.4481	150.269	350999.1235	87.880
10	3	36	7	3	3	500	300	456653.4925	150.359	357160.0109	85.863
11	3	38	8	3	3	500	300	516185.6917	164.877	407473.0801	95.500
12	4	40	8	3	3	500	300	578132.8091	173.107	466194.8065	105.894

As Table (19) verifies the algorithms reached nearly optimal solutions in reasonable intervals. Considering the value of the objective function and runtime as two main criteria to compare these meta-heuristics, genetic algorithm performed far better than ant colony optimization. The results are given in Table (20).

Table (20). The Runtimes of the GA and Ant Colony Optimization



Conclusion and Suggestions for Future research

Supply chain management has convinced many researchers within recent years aiming at facilitating industrial companies and organizations particularly in developed countries. A fruitful approach through supply chain management is the implementation of lean production and consequently, lean supply chain.

On the other hand, a successful supply chain can't be considered without designing appropriate cross docks; accordingly, logistic companies warmly welcome cross-docks in large-scale transportations. Cross docks set their target as applying the main policy for aggregating products within warehouses. Instead of being sent to customers, required demands from different suppliers will be aggregated in cross docks. In order to reduce the transportation costs, the products are better to be classified regarding customer demands and then, sent to destinations.

With respect to Dando and Cerda's model [26] on scheduling and planning cross-docking operations, we extended our model by adding several cross docks in this paper. To consider the uncertainty of fuzzy parameters, an interactive fuzzy approach was offered. There is no way to encounter the inaccurate and uncertain nature of parameters and model them except using a fuzzy distribution.

Mathematical programming techniques vividly fail to solve transportation decision making problems by fuzzy objective function coefficients. To overcome this deficiency, we provided a fuzzy-interactive multi-objective linear programming model for solving transportation decision problems by fuzzy objective function coefficients in this paper. The proposed method proved to be flexible and efficient, computationally.

As mentioned earlier, our proposed approach is a two-phase method that converts the initial fuzzy model to an equivalent auxiliary definite one in the first phase, and applies a fuzzy method to obtain the approved preferable solution, in the second step.

Using both definite and fuzzy methods for different parameters, a couple of numerical examples were conducted. Then, the final solutions were compared and analyzed. The results show high level of similarity between definite and fuzzy solutions while using certain parameters and high level of difference while using uncertain parameters to prove the importance of considering uncertainty.

No wonder, this problem is categorized as an NP-hard one regarding its time consumption and computational complexities. Accordingly, the genetic algorithm and ant colony optimization were used to solve the proposed model. Furthermore, the results of the genetic algorithm and those of the ant colony optimization were compared to each other and proved the algorithms efficiencies while being conducted on scheduling and planning cross-docking problems. It is also worth mentioning that the genetic algorithm performed better than ant colony optimization regarding time and solutions requirements.

Suggestions for Future research

Due to the extensive nature of cross dock problems, different methods have been offered to solve them. In order to change the previous models to more flexible versions, new assumptions can also be considered as follows:

- The relationship between suppliers and customers should be improved.
- Finding and applying other metaheuristic algorithms can also be left to the interested readers.
- Designing efficient and effective algorithms with the help of combining methodologies.

References:

- [1] Van Belle, J., Valckenaers, P., Cattrysse, D.: Cross-docking: State of the art. *Omega*. (2012)
- [2] Boysen, N., Flidner, M., Scholl, A.: Scheduling inbound and outbound trucks at cross docking terminals. *OR spectrum*. 32, 135-161 (2010)
- [3] TSUI L, Y., CHANG, C.H.: Microcomputer based decision support tool for assigning dock doors in freight yards. *Computers & Industrial Engineering*. 19, 309-312 (1990)
- [4] Bartholdi, J.J., Gue, K.R.: The best shape for a cross-dock. *Transportation Science*. 38, 235-244 (2004)
- [5] Yu, W., Egbelu, P. J.: Scheduling of inbound and outbound trucks in cross docking systems with temporary storage. *European Journal of Operational Research*. 184 (2008)
- [6] Li, Y., Lim, A., Rodrigues, B.: Crossdocking-JIT scheduling with time windows. *Journal of Operational Research Society*. 55, 1342-1351 (2004)
- [7] Wisittipanich, W., Hengmeechai, P.: Truck scheduling in multi-door cross docking terminal by modified particle swarm optimization. *Journal of Computers & Industrial Engineering*. 113, 793-802 (2017)
- [8] Serrano, C., Delorme, X., Dolgui, A.: Scheduling of truck arrivals, truck departures and shop-floor operation in a cross-dock platform, based on trucks loading plans. *International Journal of Production Economics*. 194, 102-112 (2017)
- [9] Molavi, D., Shahmardan, A., S.Sajadieh, M.: Truck scheduling in a cross docking systems with fixed due dates and shipment sorting. *Journal of Computers & Industrial Engineering*. 117, 29-40 (2018)
- [10] Amini, A., Tavakkoli-Moghaddam, R.: A bi-objective truck scheduling problem in a cross-docking center with probability of breakdown for trucks. *Journal of Computers & Industrial Engineering*. 96, 180-191 (2016)
- [11] Keshtzari, M., Naderi, B., Mehdizadeh, E.: An improved mathematical model and a hybrid metaheuristic for truck scheduling in cross-dock problems. *Journal of Computers & Industrial Engineering*. 91, 197-204 (2016)

- [12] Ahkamiraad, A., Wang, Y.: Capacitated and multiple cross-docked vehicle routing problem with pickup, delivery, and time windows. *Journal of Computers & Industrial Engineering*. 119, 76-84 (2018)
- [13] Bazgosha, A., Ranjbar, M., Jamili, N.: Scheduling of loading and unloading operations in a multi stations transshipment terminal with release date and inventory constraints. *Journal of Computers & Industrial Engineering*. 106, 20-31 (2017)
- [14] Hasani Goodarzi, A., Zegordi, S.H.: A location-routing problem for cross-docking networks: A biogeography-based optimization algorithm. *Journal of Computers & Industrial Engineering*. 102, 132-146 (2016)
- [15] Assadi, M.T., Bagheri, M.: Differential evolution and Population-based simulated annealing for truck scheduling problem in multiple door cross-docking systems. *Journal of Computers & Industrial Engineering*. 96, 49-161 (2016)
- [16] Maknoon, M.Y., Kone, O., Baptiste, P.: A sequential priority-based heuristic for scheduling material handling in a satellite cross-dock”, *Journal of Computers & Industrial Engineering*. 72, 43-49 (2014)
- [17] Azimi, P.: On-line cross docking: A general new concept at a container port. *Scientia Iranica*. 22, 2585-2594 (2015)
- [18] Motaghedi-Larijani, A., Aminnayeri, M.: Optimizing the number of outbound doors in the crossdock based on a new queuing system with the assumption of beta arrival time. *Scientia Iranica*. 25, 2282-2296 (2018)
- [19] BolooriArabani, A.R., FatemiGhomi, S.M., Zandieh, M.: Meta-heuristics implementation for scheduling of trucks in a cross-docking system with temporary storage. *Expert systems with Applications*. 38, 1964-1979 (2011)
- [20] Chen, P., Guo, Y., Lim, A., Rodrigues, B.: Multiple cross docks with inventory and time windows. *Computers & Operations Research*. 33, 43-63 (2006)
- [21] Lee, Y.H., Jung, J.W., Lee, K.M.: Vehicle routing scheduling for cross-docking in the supply chain. *Computers & Industrial Engineering*. 51, 247-256 (2006)
- [22] Wen, M., Larsen, J., Clausen, J., Cordeau, J.F., Laporte, G.: Vehicle routing with cross-docking. *Journal of the Operational Research Society*. 60, 1708-1718 (2009)
- [23] Miao, Z., Lim, A., Ma, H.: Truck dock assignment problem with operational time constraint within cross docks. *European journal of operational research*. 192, 105-115 (2009)
- [24] Gajovic, V., Paunović, M., Kocovic, J.: Modeling and simulation of logistic processes: risk assessment with a fuzzy logic technique. *Simulation Transactions of the Society for Modeling and Simulation International*. 94(6), 507-518 (2017)
- [25] Avila-Torres, P., Caballero, R., Litvinchev, I., Lopez-Irarragorri, F., Vasant, P.: The urban transport planning with uncertainty in demand and travel time: a comparison of two defuzzification methods. *Journal of Ambient Intelligence and Humanized Computing*. 9(3), 843-856 (2018)
- [26] Dondo, R., Cerdá, J.: A SWEEP-HEURISTIC BASED FORMULATION FOR THE VEHICLE ROUTING PROBLEM WITH CROSS DOCKING. *Computers & Chemical Engineering*. (2012)
- [27] Mousavi, S. M., Tavakkoli-Moghaddam, R.: A hybrid simulated annealing algorithm for location and routing scheduling problems with cross-docking in the supply chain. *Journal of Manufacturing Systems*. 33, 335-347 (2013)
- [28] Rajabi, M., Akbarpour Shirazi, M.: Truck scheduling in a cross-dock system with multiple doors and uncertainty in availability of trucks. *Journal of Applied Environmental and Biological Sciences*, 6, 101-109 (2016)
- [29] Liang, H., Wang, N.: Partially linear single-index measurement error models. *Statistica Sinica*. 15, 99-116 (2005)
- [30] Dubois, D., Fargier, H., Fortemps, P.: Fuzzy scheduling: Modeling flexible constraints vs. coping with incomplete knowledge. *European Journal of Operational Research*. 147, 231-252 (2003)
- [31] Baykasoğlu, A., Göçken, M., Unutmaz, D.: New approaches to due date assignment in job shops. *European Journal of Operational Research*. 187, 31-45 (2008)
- [32] Heilpern, S.: The expected value of a fuzzy number. *Fuzzy Sets and Systems*. 47, 81-86 (1992)

- [33] Jimenez, M., Arenas, M., Bilbao, A.: Linear programming with fuzzy parameters: an interactive method manufacturer. *Applied Mathematical Modelling*. *European Journal of Operational Research*. 36(6), 2762-2776 (2007)
- [34] Zimmermann, H.J.: Fuzzy programming and linear programming with several objective functions. *Fuzzy Sets and Systems*. 1, 45-55 (1978)
- [35] Inuiguchi, M., Ramík, J.: Possibilistic linear programming: a brief review of fuzzy mathematical programming and a comparison with stochastic programming in portfolio selection problem. 111, 3-28 (2000)
- [36] Parra, M.A., Terol, A.B., Uria, M.V.R.: A fuzzy goal programming approach to portfolio selection. *European Journal of Operational Research*. 133, 287–297 (2001)
- [37] Torabi, S.A., Hassini, E.: An Interactive Possibilistic Programming Approach for Multiple Objective Supply Chain Master Planning. *Fuzzy Sets and Systems*. 159, 193-214 (2008)
- [38] Dorigo, M., Caro, G. Di.: Ant colony optimization: a new meta-heuristic. *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99*. 2, 1470-1477 (1999)
- [39] Saniee Abadeh, M., Habibi, J., Soroush, E.: Induction of fuzzy classification systems via evolutionary ACO-based algorithms. *International Journal of Simulation Systems*. *Science and Technology*. 9(2008)